media5 corporation

Discover the Power of 5

# Reference Manual

# Mediatrix® 4102
# Mediatrix® 4102S

**SIP Version**

Product Version 5.0

Document Revision 13

July 30, 2012

RoHS COMPLIANT 2002/95/EC

**Media5 Corporation**
**4229 Garlock Street**
**Sherbrooke, Québec, Canada J1L 2C8**

**Mediatrix® 4102 Reference Manual**

© 2012, Media5 Corporation

All rights reserved. No part of this publication may be reproduced or used in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems – without the express written permission of the publisher.

Media5 Corporation reserves the right to revise this publication and make changes at any time and without the obligation to notify any person and/or entity of such revisions and/or changes.

**Trademarks**

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Adobe and Acrobat are registered trademarks of Adobe Systems Incorporated.

All other trademarks and registered trademarks are the property of their respective owners.

**Third-Party Software Copyright Information**

The Mediatrix® 4102 firmware aggregates some third-party software modules (open source and commercial) that are distributed to you in accordance with their respective licenses. Refer to the *Third Party Software Copyright Information* addendum available on the Mediatrix Download Portal, which lists the third-party software modules along with any copyright and license information.

# Contents

## Preface

**About this Manual ..... xix**

Document Objectives ..... xix
Intended Audience ..... xix
Related Documentation ..... xix
Document Structure ..... xx
Document Conventions ..... xxii
Warning Definition ..... xxii
Where to find Translated Warning Definition ..... xxii
Other Conventions ..... xxiii
SCN vs PSTN ..... xxiii
Standards Supported ..... xxiii
Obtaining Documentation ..... xxiii
Media5 Web Site ..... xxiii
Media5 Download Portal ..... xxiii
Documentation Feedback ..... xxiv
Unit Manager Network – Element Management System ..... xxiv
End User Technical Support ..... xxiv

# Installation and Web Page Configuration

## Chapter 1

**Installation ..... 3**

Requirements ..... 3
Safety Recommendations ..... 3
Package Contents ..... 3
Overview ..... 4
About the Mediatrix 4102 ..... 4
Placing a Call ..... 4
Management Choices ..... 5
Panels ..... 6
Front Indicators ..... 6
Rear Connectors ..... 7
Choosing a Suitable Installation Site ..... 7
Location ..... 8
Wall-Mounting ..... 8
Free Standing Unit ..... 8
Condensation ..... 9
Cleaning ..... 9
Hardware Connection ..... 9
Reserving an IP Address ..... 9
Before Proceeding ..... 9
Single Computer Installation ..... 10
Multiple Computer Installation ..... 11
Starting the Mediatrix 4102 for the First Time ..... 13

IP Address Discovery or Configuration ..... 13
Initial Provisioning Sequence ..... 14
Special Vocal Features ..... 16
LED Behaviour in Starting Mode ..... 16
LED Indicators ..... 16
Ready LED ..... 16
In Use LED ..... 16
ETH LED ..... 17
Power LED ..... 17
LED Patterns ..... 17
Booting LED Pattern Description ..... 18
NormalMode LED Pattern Description ..... 19
AdminMode LED Pattern Description ..... 19
Recovery Mode LED Patterns ..... 19
Reset / Default Switch ..... 20
At Run-Time ..... 20
At Start-Time ..... 20
Recovery Mode ..... 21
Factory Reset ..... 22
Software Restart ..... 23
Restart Behaviour ..... 23
Verifying the Installation ..... 24

## Chapter 2

**Web Interface – Introduction ..... 25**

Introduction ..... 25
End User vs. Administration Pages ..... 25
Using the Web Interface ..... 26
Web Interface Access Limitation ..... 26
Accessing the Web Interface ..... 26
System Status of the Mediatrix 4102 (End User Web Page) ..... 28
Network Parameters Status of the Mediatrix 4102 (End User Web Page) ..... 28
System Status of the Mediatrix 4102 (Administration Web Page) ..... 28
Menu Frame (End User Web Page) ..... 29
Menu Frame (Administration Web Page) ..... 30
Content Frame ..... 31
Submitting Changes ..... 31
Syslog Monitoring ..... 32
Configuring the Syslog Daemon Application ..... 33

## Chapter 3

**Web Interface ..... 35**

System Page ..... 35
WAN Page ..... 35
LAN Page ..... 37
STUN Page ..... 38
Configuration File Upload Page ..... 39
HTTP Server Password Page ..... 40
Default User Name and Password ..... 40
Issue: Factory Reset does not Reset the Default Password Value ..... 41
System Log Page ..... 42

# Chapter 4


## Web Interface – Management ..... 43

Admin Page ..... 43
HTTP Server Password – Administrator Web Page ..... 43
HTTP Server Password – End User Web Page ..... 45
System Management ..... 47
Group Port Management ..... 48
Interface Management ..... 48
Network Settings ..... 49
Ethernet Connection Speed ..... 49
Network Settings ..... 50
SNTP Settings ..... 51
Configuration File Download ..... 54
Configuration File Download Server ..... 54
Configuration File Server Settings ..... 55
Setting up the Configuration File Download ..... 56
Configuration Files Encryption ..... 59
Configuration Download Procedure ..... 60
Automatic Configuration Update ..... 60
Error Handling ..... 63
Firmware Download ..... 65
Before Downloading ..... 65
Firmware Servers Configuration ..... 66
Setting up the Firmware Download ..... 67
Firmware Download Procedure ..... 72
Automatic Firmware Update ..... 72
Spanning Tree Protocol (STP) ..... 75
Firmware Downgrade ..... 75
Emergency Firmware Procedure ..... 76


# Chapter 5


## Web Interface – SIP Parameters ..... 77

SIP Servers Configuration ..... 77
SIP Servers ..... 77
SIP Configuration ..... 78
SIP User Agent ..... 79
SIP Registration ..... 80
SIP Publication ..... 81
SIP Interop ..... 81
SIP Penalty Box ..... 81
SIP Transport Type ..... 83
Interop Parameters ..... 83
SIP Authentication ..... 85


# Chapter 6


## Web Interface – Telephony ..... 87

Digit Maps ..... 87
Syntax ..... 87
Special Characters ..... 88
How to Use a Digit Map ..... 88
General Parameters ..... 90
Allowed Digit Maps ..... 91

Blocked Digit Maps ..... 92
Voice & Fax Codecs ..... 93
G.711 PCMA and PCMU ..... 93
G.726 ..... 94
G.729 ..... 94
General Parameters ..... 95
G.711 Codec Parameters ..... 98
G.729 Codec Parameters ..... 99
G.726 Codecs Parameters ..... 99
Fax Parameters ..... 100
Call Forward ..... 104
On Busy ..... 104
On No Answer ..... 106
Unconditional ..... 107
Services ..... 109
Call Transfer ..... 109
Call Waiting ..... 110
Conference ..... 113
Call Hold ..... 115
Second Call ..... 116
Automatic Call ..... 116
Miscellaneous ..... 117
Country Selection ..... 117
Custom Tone Configuration ..... 118
Message Waiting Indicator ..... 125

Chapter 7

Web Interface – Advanced ..... 129

Quality of Service (QoS) ..... 129
802.1q Configuration ..... 129
DiffServ Configuration ..... 131
Emergency Page ..... 132
Emergency Call Configuration ..... 132
STUN Configuration ..... 133
SIP Outbound Proxy ..... 133
Restrictions on the Media5 STUN Implementation ..... 134
STUN Client Configuration ..... 134
SIP Custom NAT Traversal ..... 135

SNMP Configuration

Chapter 8

MIB Structure and SNMP ..... 139

SNMP Overview ..... 139
Definitions ..... 139
SNMP Versions ..... 140
SNMP Behaviour ..... 141
SNMPv3 Special Behaviour ..... 142
SNMP Configuration via a Configuration File ..... 143
MIB Structure ..... 147

Textual Conventions ..... 148
Objects, Conformance, and Events ..... 148
IP Addresses ..... 149
Persistence ..... 149
Changing a Parameter Value ..... 149
Tables ..... 150
Generic Variables ..... 150
Variables for Administrative Commands ..... 151
SNMP Access Limitation ..... 152
Current MIB Version ..... 152
Sending Configuration Data to the Mediatrix 4102 ..... 152
Configuration File ..... 152
Management Information Base – MIB ..... 152

# Chapter 9

**IP Address and Network Configuration ..... 155**

IP Addresses ..... 155
IP Addresses Formats in the DHCP Server ..... 155
Provisioning Source ..... 156
Services ..... 156
Configuration Source ..... 156
Local Host ..... 158
Static DNS ..... 160
Image ..... 161
Management Server ..... 162
Configuration File Fetching ..... 163
Syslog ..... 163
SIP Servers ..... 164
SNTP ..... 165
LAN Connector Static IP Address ..... 166
DHCP Configuration ..... 167
DHCP Options Waiting Time ..... 167
Bootp BROADCAST Flag in DHCP Requests ..... 167
Changing the Size of DHCP Requests ..... 168
DHCP Server Configuration ..... 168
Connection to the DHCP Behaviour ..... 168
Network Configuration ..... 169
Vendor and Site Specific DHCP Options ..... 169
Vendor Specific Options ..... 169
Site Specific Options ..... 170
Option Codes ..... 171
Entering IP Addresses ..... 171
Entering FQDNs ..... 172
Settings Example ..... 174
Error Handling ..... 175
DHCP Server Failures ..... 175
Vendor/Site Specific Option Missing ..... 175
DNS Failures ..... 175
Ethernet Connection Speed ..... 176
Speed and Duplex Detection Issues ..... 176

# Chapter 10

**SIP Servers ..... 177**

Registrar Server ..... 177

Configuration Source ..... 177
Proxy Server ..... 179
Configuration Source ..... 179
Outbound Proxy Server ..... 180
Configuration Source ..... 181
Loose Router Configuration ..... 182
Presence Compositor Server ..... 183
Configuration Source ..... 183

## Chapter 11

**DNS SRV Configuration ..... 185**

What is a DNS SRV? ..... 185
Priority vs Weight ..... 185
DNS SRV Call Flow ..... 186
Enabling DNS SRV on the Mediatrix 4102 ..... 186
DNS SRV Record Lock ..... 187
DNS SRV-Oriented Settings ..... 187

## Chapter 12

**Country-Specific Configuration ..... 189**

Caller ID Information ..... 189
Caller ID Generation ..... 189
ADSI ..... 190
Setting the Location (Country) ..... 190
Caller ID Selection ..... 191
Custom Tone Configuration ..... 192
Pattern Definition ..... 192
Customizing the Tones ..... 193
Custom Tone Example ..... 194

## Chapter 13

**Transparent Address Sharing ..... 199**

What is Transparent Address Sharing? ..... 199
Router Mode ..... 200
Cable vs DSL Modem ..... 201
Multicast and IGMP ..... 201
Configuration Steps ..... 201
PPPoE Service ..... 202
Enabling the PPPoE Service ..... 202
Setting a User Name and Password ..... 203
WAN Information Configuration Source ..... 204
Configuring TAS ..... 205
QoS Differentiated Services Fields ..... 206
LAN Interface ..... 206
MAC Address Spoofing ..... 207
WAN Upstream Bandwidth Control ..... 208
Enabling TAS ..... 209
Ports Settings ..... 210
UDP and TCP Ports ..... 210
T.38 Base Port Range ..... 210
RTP/RTCP Base Port Range ..... 211

Restarting the Mediatrix 4102 ..... 211
DHCP Server ..... 212
DHCP Server Compliance ..... 212
Supported DHCP Options ..... 212
DSL Modem Specific Information ..... 213
Establishing a Connection ..... 213
Error Handling ..... 214
Routing Mechanism ..... 215
Blocked Ports ..... 215
Using the Mediatrix 4102 with a Low Bandwidth Connection ..... 216
What is Considered a Low Bandwidth Connection? ..... 216
Configuration for a Low Bandwidth Connection ..... 216

Chapter 14

Configuration File Download ..... 219

Configuration File Download Server ..... 219
Configuring the TFTP Server ..... 219
Configuring the SNTP Server ..... 219
Configuring the HTTP Server ..... 219
Configuring the HTTPS Server ..... 220
Configuration File Server Settings ..... 221
Setting up the Configuration File Download ..... 222
Configuration Update Status ..... 224
Configuration Files Encryption ..... 225
Configuration Download via TFTP ..... 226
Configuration Download via HTTP/HTTPS ..... 227
Automatic Configuration Update ..... 228
Error Handling ..... 230
Management Server ..... 233
Management Server Configuration ..... 233
Downloading from the Management Server ..... 233
Error Handling ..... 234
Syslog Messages ..... 235
Configuration File Example ..... 236
Supported Characters ..... 237

Chapter 15

Software Download ..... 239

Before Downloading ..... 239
Configuring the TFTP Server ..... 239
Configuring the SNTP Server ..... 239
Configuring the HTTP Server ..... 239
Configuring the HTTPS Server ..... 240
Software Servers Configuration ..... 241
DHCP Configuration ..... 241
Static Configuration ..... 242
Download Procedure ..... 243
Extracting the Zip File ..... 243
Setting up the Image Path ..... 243
Software Download Status ..... 245
Download via TFTP ..... 247
Download via HTTP/HTTPS ..... 248
Automatic Software Update ..... 249
Spanning Tree Protocol (STP) ..... 251

Software Downgrade ....252
Emergency Software Procedure ....252
Using the Emergency Software .... 252

## Chapter 16

### Line Configuration .... 255

Lines Administrative State ....255
Temporary Administrative State .... 255
Permanent Administrative State .... 256
Unregistered Line Behaviour ....256
Flash Hook Detection ....256
Source Line Selection ....257
Examples of Source Line Selection Use .... 257
Loop Current ....258
Callee Hang-up Supervision ....259
Line Reversal ....260
Blanking of an Anonymous Caller ID ....260
Calling Number Transformation ....261
Regular Expressions .... 261

## Chapter 17

### Voice Transmissions .... 263

Codec Descriptions ....263
G.711 PCMA and PCMU .... 263
G.726 .... 264
G.729 .... 264
Preferred Codec ....265
Enabling Individual Codecs ....265
Packetization Time ....266
DTMF Transport Type ....268
DTMF Transport Using SIP INFO .... 269
DTMF Payload Type .... 270
DTMF – RFC 2833 Events .... 270
DTMF Transport over the SIP Protocol .... 271
DTMF Detection ....272
DTMF Frequencies .... 272
DTMF Detection Configuration .... 272
Adaptative Jitter Buffer ....274
About Changing Jitter Buffer Values .... 275
Voice Activity Detection ....275
G.711 and G.726 VAD .... 275
G.729 VAD .... 276
Echo Cancellation ....276
Signal Limiter .... 277
Comfort Noise ....277
User Gain ....278

## Chapter 18

### Fax Transmission .... 279

Introduction ....279

Fax Calling Tone Detection ..... 279
CED Fax Tone Detection ..... 280
Analog CED Detection Behaviour ..... 280
Clear Channel Fax ..... 281
Data Codec Selection Procedure ..... 283
T.38 Fax ..... 284
T.38 No-Signal ..... 285
T.38 INVITE Rejected with 606 ..... 286

# Chapter 19

## SIP Protocol Features ..... 287

User Agents ..... 287
Home Domain Override ..... 288
SIP User Agent Header ..... 288
Session Timers ..... 289
Session Timer Version ..... 289
Background Information ..... 290
Authentication ..... 290
Line-Specific Authentication ..... 290
Unit Authentication ..... 291
Authentication Request Protection ..... 291
SIP Trusted Sources ..... 292
NAT Traversal ..... 292
Mediatrix 4102 Configuration ..... 293
NAT System Configuration ..... 293
SIP Transport Type ..... 293
Transport Parameter ..... 294
UDP Source Port Behaviour ..... 294
SIP Penalty Box ..... 295
Penalty Box vs Transport Types ..... 295
Penalty Box Configuration ..... 295
Registration Parameters ..... 296
Refreshing Registration ..... 296
Registration Expiration ..... 296
Default Registration Expiration ..... 297
Publication Parameters ..... 297
Refreshing Publications ..... 297
Publications Expiration ..... 298
Default Publication Expiration ..... 298
Interop Parameters ..... 299
Call Transfer Capacity ..... 299
Transmission Timeout ..... 301
Max-Forwards Header ..... 301
Referred-By Field ..... 302
Direction Attributes in a Media Stream ..... 302
Allowing Multiple Active Media in Answer ..... 304
Local Ring Behaviour on Provisional Response ..... 304
SIP Credential ..... 305
Branch Parameter Settings ..... 305
Ringing Response Code ..... 306
URI-Parameters ..... 306
Unsupported INFO Request ..... 307
Outbound Proxy Usage ..... 307
International Code Mapping ..... 307
T.38 Negotiation Syntax ..... 308
Addressing Failed Registration Attempts ..... 308

SIP Domain in Request URI ..... 308
SIP From: URI Content ..... 309
Network Asserted Caller ID ..... 309
Payload Type Settings ..... 309
Controlling the Call Waiting Tone via SIP INFO ..... 310
Ignore Username Parameter ..... 311
Escaping the Pound (#) Character in SIP URI Username ..... 311
SIP OPTIONS Method Support ..... 311
Offer/Answer Model ..... 312
Allow Media Reactivation in Answer ..... 312
Allow Audio and Image Negotiation ..... 313
Codec Order in Answer ..... 313

## Chapter 20

**STUN Configuration ..... 315**

What is STUN? ..... 315
SIP Outbound Proxy ..... 315
Restrictions on the Media5 STUN Implementation ..... 315
STUN Client Configuration ..... 316

## Chapter 21

**SNTP Settings ..... 317**

Enabling the SNTP Client ..... 317
Configuration Source ..... 318
DHCP Configuration ..... 318
Static Configuration ..... 318
Defining a Custom Time Zone ..... 319
STD / DST ..... 319
OFFSET ..... 319
START / END ..... 319
Example ..... 320

## Chapter 22

**Digit Maps ..... 321**

What is a Digit Map? ..... 321
Syntax ..... 321
Special Characters ..... 322
How to Use a Digit Map ..... 322
Combining Several Expressions ..... 322
Using the # and * Characters ..... 323
Using the Timer ..... 323
Calls Outside the Country ..... 323
Example ..... 323
Validating a Digit Map ..... 323
Processing Digits When Pressed ..... 324
Setting up Digit Maps ..... 324
Refused Digit Maps ..... 325
Digit Maps Timeouts ..... 326
Digit Map Examples ..... 326
Digit Map Example 1 – Standard Calls ..... 326
Digit Map Example 2 – PBX Emulation ..... 328

## Chapter 23

**Telephony Features ..... 331**

Making Calls ..... 331
Complete Dialing Sequence ..... 331
Dialing a Telephone Number or Numerical Alias ..... 331
Emergency Call ..... 332

## Chapter 24

**Subscriber Services ..... 333**

Service Activation Processing ..... 333
Call Hold ..... 334
Enabling Call Hold ..... 335
Using Call Hold ..... 335
Second Call ..... 335
Enabling Second Call ..... 335
Using Second Call ..... 335
Call Forward ..... 336
Unconditional ..... 336
On Busy ..... 338
On No Answer ..... 340
Call Waiting ..... 342
Setting up Call Waiting ..... 342
Using Call Waiting ..... 343
Permanently Removing the Call Waiting Tone ..... 344
Call Transfer ..... 345
Blind Transfer ..... 345
Attended Transfer ..... 346
Conference Call ..... 347
Requirements ..... 347
Enabling the Conference Call Feature ..... 348
Managing a Conference Call ..... 349

## Chapter 25

**Telephony Attributes ..... 351**

Automatic Call ..... 351
Call Direction Restriction ..... 352
Hook Flash Processing ..... 352
IP Address Call Service ..... 353
Enabling IP Address Calls ..... 353
Dialing an IP Address ..... 353
PIN Dialing ..... 354
Remote Line Extension ..... 355
Delayed Hot Line ..... 356
Call Rejection ..... 357

## Chapter 26

**Message Waiting Indicator ..... 359**

What is Message Waiting Indicator (MWI)? ..... 359
Standard MWI Methods ..... 359

MWI Method #1 ..... 359
MWI Method #2 ..... 361
MWI Notify Service ..... 361
How does the Service Work? ..... 361
Configuring the IP Communication Server ..... 362
Configuring the Mediatrix 4102 ..... 362

## Chapter 27

**Management Server Configuration ..... 363**

Using the Management Server ..... 363
Configuration Source ..... 363

## Chapter 28

**Quality of Service (QoS) ..... 365**

Differentiated Services (DS) Field ..... 365
IEEE 802.1q ..... 366
Voice QoS vs RTCP Packets ..... 367
VLAN ..... 368
VLAN Substitution ..... 369
VLAN ID Filtering ..... 370
LAN and WAN with VLAN substitution ..... 370

## Chapter 29

**Syslog Daemon ..... 371**

Syslog Daemon Configuration ..... 371
Configuration Source ..... 372
Customizing Syslog Messages ..... 373
Configuring the Syslog Daemon Application ..... 373
Local Syslog ..... 374

## Chapter 30

**Statistics ..... 375**

RTP Statistics ..... 375
Statistics Buffers ..... 375
How are Statistics Collected? ..... 375
Statistics by Syslog ..... 376
Example ..... 377

## Chapter 31

**Maximum Transmission Unit (MTU) ..... 379**

What is MTU? ..... 379
Mediatrix 4102 MTU ..... 379
Possible Hardware Problem ..... 379

## Chapter 32

**Troubleshooting ..... 381**

General Operation Issues ..... 381
Calling Issues ..... 384
Fax Issues ..... 385
Tested Fax Models ..... 387
Issues Arising from Specific Combinations/Scenarios ..... 387
Configuration Issues ..... 388
Software Upgrade Issues ..... 388
SNMP Management Software Issues ..... 390

# Appendices

## Appendix A

**Standards Compliance and Safety Information ..... 395**

Standards Supported ..... 395
Disclaimers ..... 396
Federal Communications Commission (FCC) Part 15 ..... 396
CE Marking ..... 396
RoHS China ..... 397
Translated Warning Definition ..... 398
Safety Warnings ..... 399
Circuit Breaker (20A) Warning ..... 399
TN Power Warning ..... 399
Product Disposal Warning ..... 399
No. 26 AWG Warning ..... 399
WAN, LAN, Phone-Fax 1 and Phone-Fax 2 Connectors Warning ..... 399
LAN anf FXS Ports Connectors Warning ..... 400
Socket Outlet Warning ..... 400
Safety Recommendations ..... 400

## Appendix B

**Standard Hardware Information ..... 401**

Industry Standard Protocols ..... 401
Hardware Features ..... 402
Display ..... 402
Interfaces ..... 402
Power ..... 402
Casing / Installation ..... 402
Product Architecture Details ..... 402
Real Time Fax Router Technical Specifications ..... 403
Analog Line Interface (FXS) ..... 403
Audio Specifications ..... 404
DTMF Tone Detection ..... 404
DTMF Tone Generation ..... 404
MTBF Value ..... 404
Power Consumption ..... 405

Measurements at the DC input ..... 405
Operating Environment ..... 405
Dimensions and Weight ..... 405
Warranty ..... 405

# Appendix C

Cabling Considerations ..... 407

RJ-45 Cable ..... 407
Straight Through Cable ..... 407
Crossover Cable ..... 408
RJ-11 (Telephone) Cable ..... 409
Wiring Conventions ..... 409

# Appendix D

Country-Specific Parameters ..... 411

Definitions ..... 411
Conventions ..... 411
Distinctive Ring ..... 413
Australia ..... 414
Australia 1 ..... 414
Australia 2 ..... 415
Australia 3 ..... 416
Austria ..... 417
Austria 1 ..... 417
Austria 2 ..... 417
Brazil ..... 418
Chile ..... 419
Chile 1 ..... 419
Chile 2 ..... 420
China ..... 421
Czech Republic ..... 422
Denmark ..... 423
France ..... 424
Germany ..... 425
Germany 1 ..... 425
Germany 2 ..... 426
Germany 3 ..... 426
Hong Kong ..... 427
Indonesia ..... 428
Israel ..... 429
Italy ..... 430
Japan ..... 431
Malaysia ..... 432
Mexico ..... 433
Netherlands ..... 434
New Zealand ..... 435
North America ..... 436
North America 1 ..... 436
North America 2 ..... 437
Russia ..... 438
South Africa ..... 439
Spain ..... 440

Sweden ..... 441
Switzerland ..... 442
Thailand ..... 443
United Arab Emirates ..... 444
UK ..... 444

## Appendix E

**Glossary ..... 445**

## Appendix F

**List of Acronyms ..... 453**

## Appendix G

**List of MIB Parameters ..... 455**

PREFACE

# About this Manual

Thank you for purchasing the Mediatrix 4102 from Media5.

The Mediatrix 4102 offers two Ethernet connectors switches enabling to establish two connections between conventional analog telephones or Group 3 fax machines and either a WAN, a LAN or a personal computer. It can be used to provide connectivity to broadband access equipment for a Service Provider's IP Telephony offering to residential or SME markets. There are two Mediatrix 4102 models:

**Table 1:** Mediatrix 4102 Models

| Model | Interfaces | VoIP Call Capacity | Additional Memory |
|---|---|---|---|
| Mediatrix 4102 | 2 FXS ports | up to 2 | No |
| Mediatrix 4102S | 2 FXS ports | up to 2 | Yes |

To ensure maximum flexibility, the Mediatrix 4102 can:

- dynamically detect the most commonly used IP Telephony codecs and fax protocols, including T.38
- be auto-provisioned and remotely managed and upgraded

## Document Objectives

The *Mediatrix 4102 Reference Manual* provides technical information for the Mediatrix 4102.

Use the *Mediatrix 4102 Reference Manual* in conjunction with the appropriate publications listed in "Related Documentation" on page xix.

## Intended Audience

This manual provides all the technical information needed to install and manage the Mediatrix 4102. It is intended for network administrators and system managers who install and set up network equipment; consequently, it assumes a basic working knowledge of LANs.

From the perspective of the LAN administrator, a Mediatrix 4102 presents itself like another device to add to the LAN. It requires the same kind of TCP/IP addressing. The Mediatrix 4102 can also use a DHCP server on the LAN to automatically receive its IP configuration assignment.

## Related Documentation

In addition to this manual, the Mediatrix 4102 document set includes the following:

- *MIB Reference Manual*

  Lists and explains all parameters in the MIB structure.
- *Mediatrix 4102 Quick Start Guide*

  This printed booklet allows you to quickly setup and work with the Mediatrix 4102.

- *Third Party Software Copyright Information*

  This document lists the third-party software modules used in the Mediatrix 4102 along with any copyright and license information. This document is available at: http://www.media5corp.com/repository/common%20manuals/Third-Party_Software_Copyright_Information_Mediatrix.pdf.

Be sure to read any readme files, technical bulletins, or additional release notes for important information.

# Document Structure

The Mediatrix 4102 Reference Manual has three parts:

- "Installation and Web Page Configuration" on page 1. This part describes various installation of the Mediatrix 4102 and how to configure the unit via its web interface.
- "SNMP Configuration" on page 137. This part describes all SNMP-related parameters of the Mediatrix 4102.
- "Appendices" on page 393: This part contains supplemental information useful to the reader.

The Mediatrix 4102 Reference Manual contains the following information.

**Table 2:** Mediatrix 4102 Reference Manual Chapter/Appendices

| Title | Summary |
|---|---|
| **Installation and Web Page Configuration** | |
| "Chapter 1 - Installation" on page 3 | Describes the various installation scenarios of the Mediatrix 4102. Also presents the possible states and LED patterns of the Mediatrix 4102, as seen from an operator perspective. |
| "Chapter 2 - Web Interface – Introduction" on page 25 | Describes how to access the embedded web server of the Mediatrix 4102 to set parameters by using the HTTP protocol. |
| "Chapter 3 - Web Interface" on page 35 | Describes the web page parameters. |
| "Chapter 4 - Web Interface – Management" on page 43 | Describes the *Management* page of the web interface, which allows you to configure the configuration file download and firmware download parameters of the Mediatrix 4102. |
| "Chapter 5 - Web Interface – SIP Parameters" on page 77 | Describes the *SIP* page of the web interface, which allows you to configure various SIP-related parameters of the Mediatrix 4102. |
| "Chapter 6 - Web Interface – Telephony" on page 87 | Describes the *Telephony* page of the web interface, which allows you to configure the various telephony parameters of the Mediatrix 4102. |
| "Chapter 7 - Web Interface – Advanced" on page 129 | Describes the *Advanced* page of the web interface, which allows you to configure various system and network parameters of the Mediatrix 4102. |
| **SNMP Configuration** | |
| "Chapter 8 - MIB Structure and SNMP" on page 139 | Describes how the Mediatrix 4102 uses the SNMP protocol for its configuration. |
| "Chapter 9 - IP Address and Network Configuration" on page 155 | Describes how to set IP information in the Mediatrix 4102 and how to configure a DHCP server. |

**Table 2:** Mediatrix 4102 Reference Manual Chapter/Appendices (Continued)

| Title | Summary |
|---|---|
| "Chapter 10 - SIP Servers" on page 177 | Describes how to configure the Mediatrix 4102 to properly use the SIP servers. |
| "Chapter 11 - DNS SRV Configuration" on page 185 | Describes the Mediatrix 4102's behaviour with a DNS SRV. |
| "Chapter 12 - Country-Specific Configuration" on page 189 | Describes how to set the Mediatrix 4102 with the proper country settings. |
| "Chapter 13 - Transparent Address Sharing" on page 199 | Explains how to properly configure the Transparent Address Sharing service for a cable or DSL modem. |
| "Chapter 14 - Configuration File Download" on page 219 | Describes how to use the configuration file download feature to update the Mediatrix 4102 configuration. |
| "Chapter 15 - Software Download" on page 239 | Describes how to download a software version available on the designated software server into the Mediatrix 4102. |
| "Chapter 16 - Line Configuration" on page 255 | Describes the features available on the lines connected to the Mediatrix 4102. |
| "Chapter 17 - Voice Transmissions" on page 263 | Describes the various codecs the Mediatrix 4102 supports for transmitting audio signals. |
| "Chapter 18 - Fax Transmission" on page 279 | Describes how to perform fax transmissions in clear channel and T.38 with the Mediatrix 4102. |
| "Chapter 19 - SIP Protocol Features" on page 287 | Describes the SIP-specific feature to set up to properly use the SIP signalling programs and information defined in the Media5 SIP stack. |
| "Chapter 20 - STUN Configuration" on page 315 | Describes how to configure the STUN client of the Mediatrix 4102. |
| "Chapter 21 - SNTP Settings" on page 317 | Describes how to configure the Mediatrix 4102 to enable the notion of time (date, month, time) into it. |
| "Chapter 22 - Digit Maps" on page 321 | Describes how to use a Digit Map to compare the number users dialed to a string of arguments. |
| "Chapter 23 - Telephony Features" on page 331 | Explains how to perform basic calls with the Mediatrix 4102 and set the telephony variables of the unit to define the way it handles calls. |
| "Chapter 24 - Subscriber Services" on page 333 | Describes how to set and use the subscriber services available on the user's telephone. |
| "Chapter 25 - Telephony Attributes" on page 351 | Describes the telephony attributes available on the Mediatrix 4102. |
| "Chapter 26 - Message Waiting Indicator" on page 359 | Explains how to set the Mediatrix 4102 to use the Message Waiting Indicator service. |
| "Chapter 27 - Management Server Configuration" on page 363 | Describes how to configure the Mediatrix 4102 to connect to a module or software that is used to remotely set up Mediatrix units. |
| "Chapter 28 - Quality of Service (QoS)" on page 365 | Defines the QoS (Quality of Service) features available on the Mediatrix 4102. |

**Table 2:** Mediatrix 4102 Reference Manual Chapter/Appendices (Continued)

| Title | Summary |
|---|---|
| "Chapter 29 - Syslog Daemon" on page 371 | Describes how to configure and use the Syslog daemon. |
| "Chapter 30 - Statistics" on page 375 | Defines the statistics the Mediatrix 4102 can collect. |
| "Chapter 31 - Maximum Transmission Unit (MTU)" on page 379 | Describes the MTU (Maximum Transmission Unit) requirements of the Mediatrix 4102. |
| "Chapter 32 - Troubleshooting" on page 381 | Examines some of the problems you may experience when connecting the Mediatrix 4102 to the network and provides possible solutions. |
| **Appendices** | |
| "Appendix A - Standards Compliance and Safety Information" on page 395 | Lists the various standards compliance of the Mediatrix 4102. |
| "Appendix B - Standard Hardware Information" on page 401 | Lists the technical hardware information of the Mediatrix 4102. |
| "Appendix C - Cabling Considerations" on page 407 | Describes the pin-to-pin connections for cables used with the Mediatrix 4102. |
| "Appendix D - Country-Specific Parameters" on page 411 | Lists the various parameters specific to a country such as loss plan, tones and rings, etc. |

# Document Conventions

The following information provides an explanation of the symbols that appear on the Mediatrix 4102 and in the documentation for the product.

## Warning Definition

**Warning:** Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, you must be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents.

## Where to find Translated Warning Definition

For safety and warning information, see "Appendix A - Standards Compliance and Safety Information" on page 395.

This Appendix describes the international agency compliance and safety information for the Mediatrix 4102. It also includes a translation of the safety warning listed in the previous section.

### Other Conventions

The following are other conventions you will encounter in this manual.

**Caution:** Caution indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury and/or damage to the equipment or property.

**Note:** Note indicates important information about the current topic.

| | |
|---|---|
| **Standards Supported** | Indicates which RFC, Draft or other standard document is supported for a specific feature. |

This symbol indicates you can also set the current configuration by using the Unit Manager Network Graphical User Interface. The text will provide the location in the *Unit Manager Network Administration Manual* where to find information related to the specific configuration.

### SCN vs PSTN

In Media5' and other vendor's documentation, the terms SCN and PSTN are used. A SCN (Switched Circuit Network) is a general term to designate a communication network in which any user may be connected to any other user through the use of message, circuit, or packet switching and control devices. The Public Switched Telephone Network (PSTN) or a Private Branch eXchange (PBX) are examples of SCNs.

### Standards Supported

When available, this document lists the standards onto which features are based. These standards may be RFCs (Request for Comments), Internet-Drafts, or other standard documents.

The Mediatrix 4102's implementations are ***based*** on the standards, so it's possible that some behaviour differs from the official standards.

For more information on and a list of RFCs and Internet-Drafts, refer to the IETF web site at http://www.ietf.org.

## Obtaining Documentation

These sections explain how to obtain documentation from Media5.

### Media5 Web Site

Media5 offers the latest version of its products' documentation on its web site. You will thus be able to access and download the most current Media5 documentation. Follow this link: http://www.media5corp.com/en/documentation.

**Note:** This site does not contain any firmware versions.

### Media5 Download Portal

Media5 offers online documentation via a self register web-portal. You will thus be able to access and download the most current Media5 documentation. Follow this link to register: http://www.media5corp.com/en/support-portal.

**Note:** This site does not contain any firmware versions.

### Documentation Feedback

Media5 welcomes your evaluation of this manual and any suggestions you may have. These help us to improve the quality and usefulness of our publications.

Please send your comments to:

Media5 Corporation
Attention: Documentation Department
4229, Garlock Street
Sherbrooke, Quebec
Canada J1L 2C8
FAX: +1 (819) 829-5100

We appreciate your comments.

## Unit Manager Network – Element Management System

The Unit Manager Network is a user-friendly element management system designed to facilitate the deployment, configuration and provisioning of Mediatrix access devices and gateways.

The Unit Manager Network offers the following key features, enabling the simple and remote configuration and deployment of numerous Mediatrix units:

- Detection of the state of each Mediatrix unit (e.g. power on/off).
- Automatic update of the list with installation of new Mediatrix units.
- Real-time graphical presentation of actual configuration.
- Tracking of all configuration options of the Mediatrix units on the network.
- Control of configuration parameters of all Mediatrix units within the same network.
- Storage of backup configuration file of each Mediatrix unit.
- Display of firmware release for any Mediatrix unit.
- Field-upgrade of all Mediatrix units.
- Controlled Implementation of new software.
- Supports SNMP requests: GET, GET NEXT, GET TABLE, GET WALK, SET, TRAP.
- SNMP abstraction layer: configuration can be changed without SNMP MIB knowledge.

The demo version of the Unit Manager Network is available on the Media5 Download Portal at: https://support.mediatrix.com/DownloadPlus/Download.asp.

See the *Unit Manager Network Administration Manual* for more details on how to use it to configure any Mediatrix 4102 unit on the network.

## End User Technical Support

In order to maximize technical support resources, Media5 works through its partners to resolve technical support issues. All end users requiring technical support are encouraged to contact their vendor directly.

# Installation and Web Page Configuration

**Page Left Intentionally Blank**

CHAPTER 1

# Installation

This chapter describes the installation and initial provisioning of the Mediatrix 4102.

## Requirements

The Mediatrix 4102 requires the following items to work properly:

**Table 3:** Required Items

| Item | Description |
|---|---|
| DHCP Server (optional) | Supplies network parameters to the Mediatrix 4102. This applies to the DHCP connection type (usually with a cable modem installation). |
| DNS Server (optional) | Translates domain names into IP addresses. |
| SIP Server | Manages the active calls of the Mediatrix 4102. |
| Management Server (optional) | Module or software used to remotely manage and configure the Mediatrix 4102. Such software could be the Media5 Unit Manager Network. See "Unit Manager Network – Element Management System" on page xxiv for more details. |
| TFTP Server or HTTP Server | Necessary for software updates. |
| Syslog Daemon (optional) | Receives all status messages coming from the Mediatrix 4102. |

## Safety Recommendations

To ensure general safety, follow these guidelines:

- Do not open or disassemble the Mediatrix 4102.
- Do not get the Mediatrix 4102 wet or pour liquids into it.
- Do not perform any action that creates a potential hazard to people or makes the equipment unsafe.

## Package Contents

The Mediatrix 4102 package contains the following items:

- the Mediatrix 4102 unit
- a power cord for the country in which you are using the Mediatrix 4102
- a universal power supply
- a printed Flyer

You also need two 10/100 BaseT Ethernet RJ-45 cables.

# Overview

The Mediatrix 4102 is a standalone Internet telephony access device that connects to virtually any business telephone system supporting standard analog lines.

The Mediatrix 4102 offers two Ethernet connectors switches enabling to establish two connections between conventional analog telephones or Group 3 fax machines and either a WAN, a LAN or a personal computer.

The Mediatrix 4102 can be used to connect up to two analog phones or fax machines to a broadband access equipment for a Service Provider's IP Telephony offering to residential users.

This version of the Mediatrix 4102 uses the Session Initiation Protocol (SIP), which is a protocol for transporting call setup, routing, authentication, and other feature messages to endpoints within the IP domain.

## About the Mediatrix 4102

The Mediatrix 4102:

- Merges voice and data traffic onto a single unified network. Carrying telephone traffic over data networks uses less bandwidth (as compared to telephone trunks), resulting in a more cost-effective network solution.
- Easily integrates with existing telephone equipment. It converts any conventional analog telephone or fax machine into an Internet device.
- Bypasses long-distance toll charges for realized savings.
- Supports 10 Mb/s and 100 Mb/s Ethernet networks.
- Upgrades software easily for future enhancements.
- Uses the latest standards in Internet Telephony.
    - SIP protocol for call management
    - T.38 for fax relay
- Supports the following Codecs:
    - G.711 (µ-law, A-law)
    - G.726
    - G.729 A
    - G.729 A rev. B
    - T.38 (fax) over UDP or TCP
- Supports Quality of Service technologies.
    - Differentiated Services (DS) Field
    - IEEE 802.1q user priority tagging
- Offers an intuitive Web-based management interface to simplify operation and support.

## Placing a Call

You can place a call from a telephone or fax connected to a Mediatrix 4102. The unit automatically detects if the call originates from a voice or fax transmission and acts accordingly.

When placing a call, the Mediatrix 4102 collects the DTMF digits dialed and sends a message to the Registrar Server. The Registrar Server sends back a list of contacts where the dialed number could be located.

You can dial on a telephone/fax machine connected to the Mediatrix 4102 as you normally do.

## Management Choices

The Mediatrix 4102 offers various management options to configure the unit.

**Table 4:** Management Options

| Management Choice | Description |
|---|---|
| Web Interface | The Mediatrix 4102 web interface offers the following options:<br>• Password-protected access via basic HTTP authentication, as described in RFC 2617<br>• User-friendly GUI<br>Refer to "Chapter 2 - Web Interface – Introduction" on page 25 for more details. |
| SNMPv1/2/3 | The Mediatrix 4102 SNMP feature offers the following options:<br>• Password-protected access<br>• Remote management<br>• Simultaneous management<br>Refer to "Chapter 8 - MIB Structure and SNMP" on page 139 for more details. |
| Auto-Update | The Mediatrix 4102 auto-update options are as follows:<br>• Frequent polling<br>• Automatic software and configuration files downloads<br>• Configuration file encryption<br>Refer to "Chapter 15 - Software Download" on page 239 and "Chapter 14 - Configuration File Download" on page 219 for more details. |

# Panels

This section provides an overview of the front and rear panels of the Mediatrix 4102.

## Front Indicators

Figure 1 shows the visual indicators located on the front of the Mediatrix 4102.

**Figure 1:** Front Panel Indicators

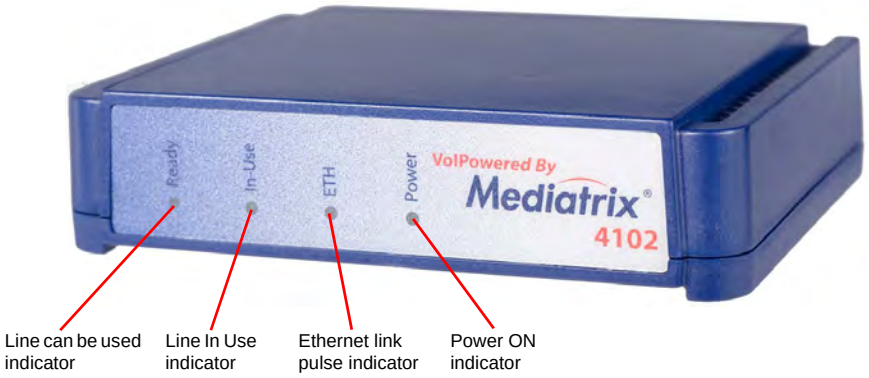



Table 5 describes the LEDs on the front panel of the Mediatrix 4102.

**Table 5:** Front Panel Indicators

| Indicator | Description |
|---|---|
| Ready | When lit, the Mediatrix 4102 is ready to initiate or receive a call. The unit does not have to be registered to a server. |
| In Use | When lit, at least one of the FXS lines is in use. |
| ETH | Provides the state of the network connected to the *WAN* and *LAN* connectors. |
| Power | When lit, power is applied to the Mediatrix 4102. |

See "LED Indicators" on page 16 for a detailed description of the LED patterns the Mediatrix 4102 may have and the states they represent.

### Rear Connectors

The Mediatrix 4102 has several connections that must be properly set. Figure 2 shows the back panel of the Mediatrix 4102.

**Figure 2:** Back Panel Connectors

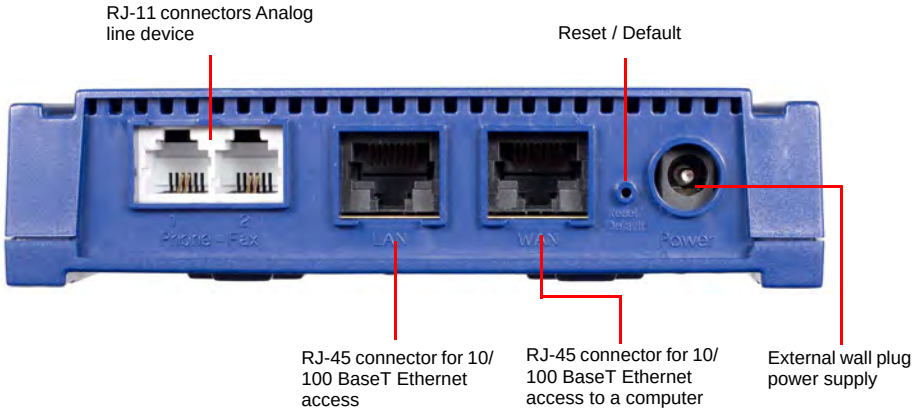



Table 6 describes the back panel connections.

**Table 6:** Back Connections of the Mediatrix 4102

| Connection | Description |
|---|---|
| WAN1 | A 10/100 BaseT Ethernet RJ-45 connector for access to a LAN, WAN or computer. |
| LAN | A 10/100 BaseT Ethernet RJ-45 connector that can be connected into the network card of a computer. |
| Phone-Fax 1 / Phone-Fax 2 | Two RJ-11 connectors to attach a conventional telephone or G3 fax machine. |
| Power connector | External 12 Vdc power supply. |
| Reset / Default switch | Resets configuration parameters of the Mediatrix 4102 to default (known) values. It can be used to reconfigure the unit.<br>**Warning**: Read Section "Reset / Default Switch" on page 20 before attempting to reset the unit. |

# Choosing a Suitable Installation Site

**Warning:** The analog lines of the Mediatrix 4102 are not intended for connection to a telecommunication network that uses outside cable.

**Warning:** To prevent fire or shock hazard do not expose the unit to rain or moisture.

The Mediatrix 4102 is suited for use in an office or residential environment where it can be wall-mounted or free standing.

## Location

Install the Mediatrix 4102 in a well-ventilated location where it will not be exposed to high temperature or humidity. Do not install the Mediatrix 4102 in a location exposed to direct sunlight or near stoves or radiators. Excessive heat could damage the internal components.

When deciding where to position the Mediatrix 4102, ensure that:

- The Mediatrix 4102 is accessible and cables can be easily connected.
- The cabling is away from the following:
  - Sources of electrical noise such as radios, transmitters, and broadband amplifiers.
  - Power lines and fluorescent lighting fixtures.
  - Water or moisture that could enter the casing of the Mediatrix 4102.
- The airflow is not restricted around the Mediatrix 4102 or through the vents on the top of the unit. The unit requires a minimum of 25 mm (1 in.) clearance.
- The operating temperature is between $0^{o}$C and $40^{o}$C.
- The humidity is not over 85% and is non-condensing.

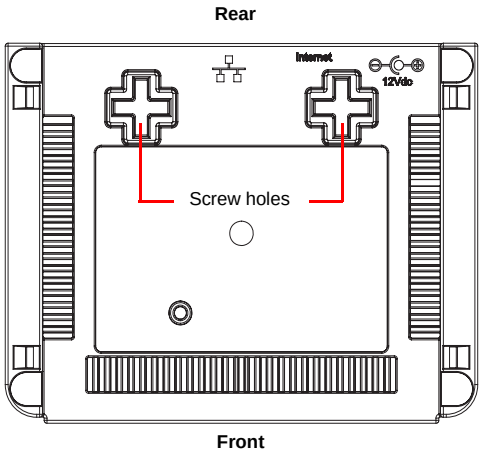
## Wall-Mounting

The Mediatrix 4102 has two screw holes on its bottom surface, allowing a single unit to be wall-mounted.

**To wall-mount the Mediatrix 4102:**

1. Disconnect all of the cables from the Mediatrix 4102 before mounting.
2. Ensure that the wall you are using is smooth, flat, dry and sturdy. Attach a piece of plywood, approximately 250 mm x 200 mm x 12 mm (10 inches x 8 inches x 0.5 inches) securely to the wall, if necessary.
3. Position the Mediatrix 4102 against the wall (or plywood) as illustrated in Figure 3.

**Figure 3:** Bottom View - Wall Mounting Screw Holes




You can position the Mediatrix 4102 any way you want.

4. Mark the position of the screw holes on the wall. Drill the two holes and install two screws.
5. Place the screw holes of the Mediatrix 4102 over the screws installed in the previous step.
6. Proceed to "Hardware Connection" on page 9.

## Free Standing Unit

When installing the Mediatrix 4102 on a desk or table, it should be located at least 20 cm from your monitor, computer casing or other peripherals, including speakers. Never put books or paper on the Mediatrix 4102.

### Condensation

When bringing the unit into a warm environment from the cold, condensation may result that might be harmful to the unit. If this occurs, allow the unit to acclimatize for an hour before powering it on.

### Cleaning

To clean the Mediatrix 4102, wipe with a soft dry cloth. Do not use volatile liquids such as benzine and thinner that are harmful to the unit casing.

For resistant markings, wet a cloth with a mild detergent, wring well and then wipe off. Use a dry cloth to dry the surface.

# Hardware Connection

**Warning:** Do not connect the Mediatrix 4102 directly to Telecommunication Systems.

The Mediatrix 4102 may be installed in various ways. This section describes two of these installations: in a single computer configuration without a router and multi-computer configuration with a router.

See "Appendix C - Cabling Considerations" on page 407 for more details on the cables the Mediatrix 4102 uses.

## Reserving an IP Address

**Note:** Perform this step only for a cable modem installation.

Before connecting the Mediatrix 4102 to the network, Media5 strongly recommends that you reserve an IP address in your DHCP server – if you are using one – for the unit you are about to connect. This way, you know the IP address associated with a particular unit.

DHCP servers generally allocate a range of IP addresses for use on a network and reserve IP addresses for specific devices using a unique identifier for each device. The Mediatrix 4102 unique identifier is its media access control (MAC) address. You can locate the MAC address as follows:

- It is printed on the label located on the bottom side of the unit.
- It is located in the *sysMgmtMIB* under the *sysMacAddress* variable.
- You can dial the following digits on a telephone connected to the Mediatrix 4102:

  `*#*1`

  The Mediatrix 4102 answers back with its MAC address. See "Special Vocal Features" on page 16 for more details.

## Before Proceeding

Most computers are configured by default to automatically obtain an IP address via DHCP. If the computer connected to the Mediatrix 4102 is set with a static IP address, you must change the setting. Please refer to your operating system's documentation to perform this task.

### 10/100 BaseT Ethernet RJ-45 Cable

When connecting an Ethernet cable to the Mediatrix 4102, use a standard telecommunication cord with a minimum of 26 AWG wire size.

You can either use a crossover or straight Ethernet cable to connect in the *WAN* or *LAN* connectors. These connectors perform automatic MDI / MDIX detection, meaning that they adapt to the type of cable connected to them.

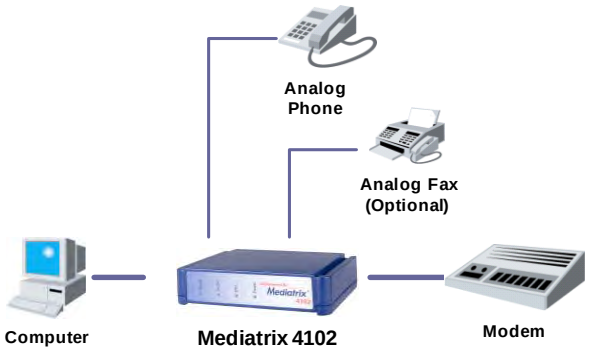
The auto MDI / MDIX feature works only when the connectors are configured in auto detect mode (see "Ethernet Connection Speed" on page 176 for more details).

Whenever you force the Mediatrix 4102 to use a specific Ethernet mode (for example 100Mb Full Duplex), the type of cable to use depends on the other peer. For example, a straight cable is required to connect the Mediatrix 4102 to a hub or a switch, while a crossover cable is required to connect the Mediatrix 4102 to a PC. See "Appendix C - Cabling Considerations" on page 407 for more details.

## Single Computer Installation

The following steps describe how to install the Mediatrix 4102 with a single computer. The installation may either be performed with a cable or DSL modem. The resulting layout could be something similar to Figure 4.

**Figure 4:** Single Computer Network Configuration

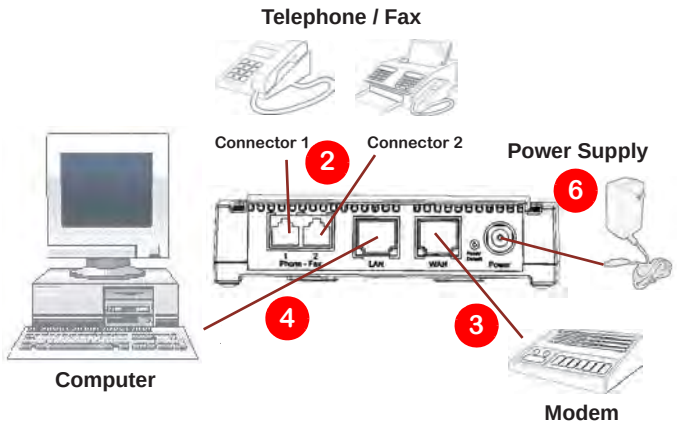



If your Internet connection requires PPPoE authentication, enter a PPPoE user name and password in the Mediatrix 4102 as described in "System Page" on page 35.

**Note:** Do not set the PC to use PPPoE because the Mediatrix 4102 will take care of the authentication.

The following figure illustrates the hardware connections.

**Figure 5:** Single Computer Installation




▶ **To install the Mediatrix 4102 with a single computer:**

1. Before you begin, be sure that all of your hardware is powered off, including the PC, modem, and Mediatrix 4102.
2. Connect analog telephones or fax machines into the *Phone/Fax* connectors.

3. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *WAN* connector of the Mediatrix 4102. Connect the other end to the cable or DSL modem.

   See "10/100 BaseT Ethernet RJ-45 Cable" on page 9 for more details on this cable.

4. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *LAN* connector of the Mediatrix 4102. Connect the other end to the network card of your computer.

   See "10/100 BaseT Ethernet RJ-45 Cable" on page 9 for more details on this cable.

5. Power on the cable or DSL modem. Depending on your modem, you may have to wait a few minutes before it properly establishes the Internet connection. Refer to your modem's documentation for more details.

6. Once the modem is ready, connect the power cord to the Mediatrix 4102 and then connect the other end to an electrical outlet.

**Warning:** The electrical outlet must be installed near the Mediatrix 4102 so that it is easily accessible.

This turns the Mediatrix 4102 on. You should not unplug it when not in use.

- If the *Power* LED is steady on, proceed with the next step.
- If the *Power* LED is blinking, wait until the *In Use* LED blinks before proceeding with the next step. This may take up to three minutes.

Most DSL users will need to enable PPPoE after they have installed the Mediatrix 4102. The *Ready* LED will remain off until this setting has been changed. For more informations, refer to "Chapter 2 - Web Interface – Introduction" on page 25.

7. Power on the PC.

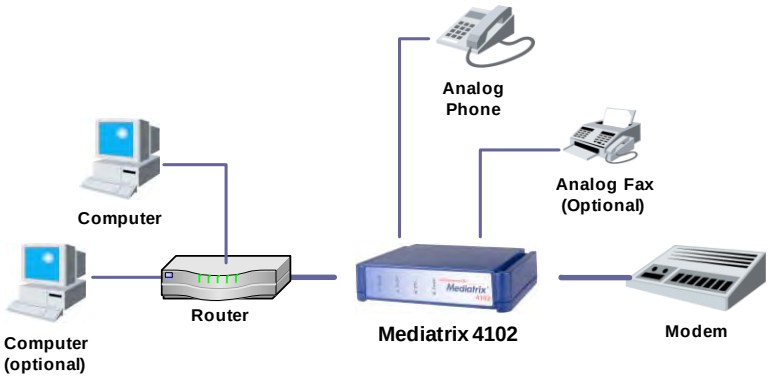
   Your computer does not have to be turned on for the telephone or fax services.

   Media5 suggests to access the unit's web interface to configure its basic uplinks parameters. See "Chapter 2 - Web Interface – Introduction" on page 25 for more details.

## Multiple Computer Installation

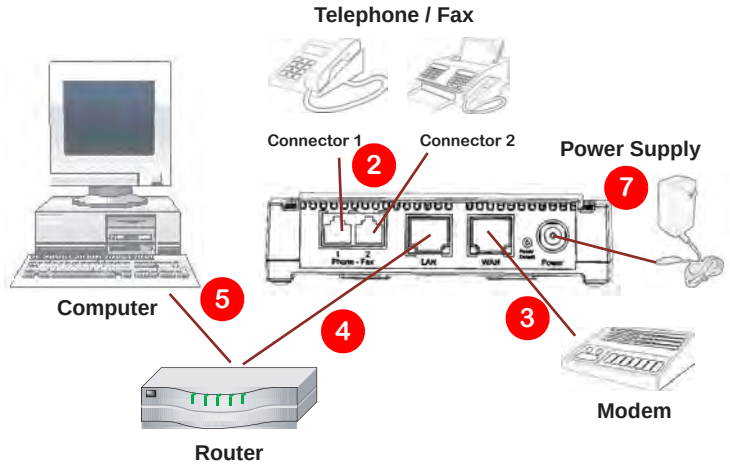
You can use a router with the Mediatrix 4102 to provide Internet connectivity to more than one PC or other device. The following steps describe how to install the Mediatrix 4102 with a router. The installation may either be performed with a cable or DSL modem. The resulting layout could be something similar to Figure 6.

**Figure 6:** Router Network Configuration




The following figure illustrates the hardware connections.

**Figure 7:** Router Installation




If your Internet connection requires PPPoE authentication, enter a PPPoE user name and password in the Mediatrix 4102 as described in "System Page" on page 35.

**Note:** Most home routers are configured by default to automatically obtain an IP address via DHCP. If the router connected to the Mediatrix 4102 is set with a static IP address, you must change the setting. Please refer to your router's documentation to perform this task.

**To install the Mediatrix 4102 with a router:**

1. Before you begin, be sure that all of your hardware is powered off, including the PC, router, modem, and Mediatrix 4102.
2. Connect analog telephones or fax machines into the *Phone/Fax* connectors.
3. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *WAN* connector of the Mediatrix 4102. Connect the other end to the cable or DSL modem.

   See "10/100 BaseT Ethernet RJ-45 Cable" on page 9 for more details on this cable.
4. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *LAN* connector of the Mediatrix 4102. Connect the other end to the WAN / Uplink connector of the router.

   See "10/100 BaseT Ethernet RJ-45 Cable" on page 9 for more details on this cable.
5. Connect a 10/100 BaseT Ethernet RJ-45 cable into the LAN connector of the router. Connect the other end to the network card of your PC.

   See "10/100 BaseT Ethernet RJ-45 Cable" on page 9 for more details on this cable.
6. Power on the cable or DSL modem. Depending on your modem, you may have to wait a few minutes before it properly establishes the Internet connection. Refer to your modem's documentation for more details.
7. Once the modem is ready, connect the power cord to the Mediatrix 4102 and then connect the other end to an electrical outlet.

**Warning:** The electrical outlet must be installed near the Mediatrix 4102 so that it is easily accessible.

This turns the Mediatrix 4102 on. You should not unplug it when not in use.

- If the *Power* LED is steady on, proceed with the next step.
- If the *Power* LED is blinking, wait until the *In Use* LED blinks before proceeding with the next step. This may take up to three minutes.

8. Power on the router. Depending on your router, you may have to wait a few minutes before it is ready. Refer to your router's documentation for more details.

   Most DSL users will need to enable PPPoE after they have installed the Mediatrix 4102. The *Ready* LED will remain off until this setting has been changed. For more informations, refer to "Chapter 2 - Web Interface – Introduction" on page 25.

9. Power on the PC.

   Your computer does not have to be turned on for the telephone or fax services.

   Media5 suggests to access the unit's web interface to configure its basic uplinks parameters. See "Chapter 2 - Web Interface – Introduction" on page 25 for more details.

# Starting the Mediatrix 4102 for the First Time

This step depends on the WAN connection type set in the web interface (see "Chapter 2 - Web Interface – Introduction" on page 25).

Most DSL users will need to enable PPPoE after they have installed the Mediatrix 4102. The *Ready* LED will remain off until this setting has been changed. For more informations, refer to "Chapter 2 - Web Interface – Introduction" on page 25.

If you are using a cable modem, the default MIB parameters are set so that the unit can be directly plugged into a network and provisioned with a DHCP server. Media5 strongly recommends to set your DHCP server before installing the unit on the network. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

If you are experiencing problems, or if you do not want to use a DHCP server, perform a Recovery Mode procedure, as explained in "Recovery Mode" on page 21.

## IP Address Discovery or Configuration

Once the physical connection is complete and the Mediatrix 4102 is powered up, the first thing to do is find out the IP address the Mediatrix 4102 is using. The Mediatrix 4102's IP address can be set either dynamically or statically. The default behaviour of the Mediatrix 4102 is to try to obtain a dynamic IP address through a DHCP server.

**Caution:** If you set a Mediatrix 4102 with a static *eth1-4* IP address in a subnet (for instance, 192.168.200.1) and the *eth5* interface receives a dynamic IP address in the same subnet (via a DHCP server or PPP peer), you will not be able to contact the unit via the WAN. You must be careful that a dynamic IP address does not overlap a static IP subnet that is already configured. Note that the current default value of the Mediatrix 4102 is 192.168.0.10.

### Dynamic IP Address Discovery

Before connecting the Mediatrix 4102 to the network, Media5 strongly suggests that you reserve an IP address in your DHCP server for the unit you are about to connect ("Reserving an IP Address" on page 9).

If you have not reserved an IP address, you can discover which IP address has been assigned to the Mediatrix 4102 by either:

- consulting your DHCP server's logs to find out details on the DHCP lease that was given to the Mediatrix 4102.
- using a network packet sniffer (e.g., Wireshark) to examine the DHCP messages exchanged between the Mediatrix 4102 and your DHCP server while the Mediatrix 4102 boots up.

**To start the Mediatrix 4102 with a dynamic IP address:**

1. If you need to discover the IP address of the Mediatrix 4102, install and start your network packet sniffer.

2. Power on the Mediatrix 4102 by connecting the other end of the power cord to an electrical outlet. The electrical outlet must be installed near the Mediatrix 4102 so that it is easily accessible.

**Note:** If the *Power* LED is always blinking and never turns on, this means that the Mediatrix 4102 cannot find a DHCP server. Check that you have a DHCP server properly configured on your network. If you do not have a DHCP server, go to the section "Default Static IP Address Configuration" on page 14.

### Default Static IP Address Configuration

If there is no DHCP server in your network, then the IP address has to be configured statically.

▶ **To start the Mediatrix 4102 with a static IP address:**

1. With a 10/100 Hub and two 10/100 BaseT Ethernet RJ-45 straight cables, connect both cables to the Hub; one of them is connected into the *WAN* connector of the Mediatrix 4102 and the other one links the computer to the Hub.
2. Reconfigure the IP address of your computer to *192.168.0.10* and the Subnet Mask to *255.255.255.0*. Restart the computer.
3. Power on the Mediatrix 4102 by connecting the other end of the power cord to an electrical outlet. The electrical outlet must be installed near the Mediatrix 4102 so that it is easily accessible.
4. Insert a small, unbent paper clip into the *Reset / Default* switch hole located at the  of the Mediatrix 4102. The *Power* LED will start blinking, and after a few seconds, all the LEDS will start blinking. Release the paper clip after all the LEDs start blinking and before they all stop blinking (between 5-10 seconds). Only the *Power* and *Ready* LEDs should go on blinking to inform you that the recovery mode procedure has been performed.

   After a recovery mode is performed, the Mediatrix 4102 uses the default IP address 192.168.0.1. Refer to "Recovery Mode" on page 21 for details on the recovery mode procedure.

   You must perform the recovery mode in a closed network and perform it on only one Mediatrix 4102 at a time, since the default IP address is the same on every unit.

## Initial Provisioning Sequence

When starting the Mediatrix 4102 for the first time, it needs to be configured before it can support calls. This process is known as *provisioning*. This sequence assumes that you have installed the Mediatrix 4102 hardware as per "Hardware Connection" on page 9.

The Mediatrix 4102 requests its configuration only on the first restart. You can change the configuration at will after the initial provisioning and the provisioning system can refresh the Mediatrix 4102 configuration. The provisioning system consists of the Management Server and a DHCP server. The Management Server includes a provisioning client, provisioning server, and SNMP proxy server.

### Provisioning Sequence in DHCP

The following describes the initial provisioning sequence of a Mediatrix 4102 that uses a DHCP connection type.

▶ **Initial provisioning sequence:**

1. When the Mediatrix 4102 starts, it broadcasts a message requesting DHCP services (if the unit is configured to start in DHCP mode).
2. The DHCP server responds with a set of IP addresses and network parameters, one of which is the Mediatrix 4102 IP address.

   The following are some of the network parameters assigned via DHCP:

   - Mediatrix 4102 IP address
   - Subnet Mask
   - Default Router IP address
   - Primary and Secondary DNS IP addresses

- Management Server IP address and port number (optional)
- Configuration file server IP address and port number (optional)
- SIP Servers IP address and port number

3. The Mediatrix 4102 may request its configuration in two ways:
   - by using the IP address of the Management Server to request its configuration.
   - by using a configuration file.

## Provisioning Sequence in PPPoE

If the WAN connection type is set to PPPoE (see "Chapter 2 - Web Interface – Introduction" on page 25), the Mediatrix 4102 establishes a PPP session with an access concentrator. This access concentrator sends the following information to the unit:

- WAN IP address
- Default Router IP address (0.0.0.0).

The access concentrator may also supply primary and secondary DNS servers. If this is the case, the new DNS servers supersede the servers defined locally.

Refer to "Chapter 13 - Transparent Address Sharing" on page 199 for more details.

This implies that you may have to enter static values for additional network parameters.

**To set static information:**

1. In the *ipAddressConfig* folder, locate the *localHostSelectConfigSource* variable (under the *ipAdressConfigLocalHost* group).
2. Set this variable to **static**.
3. Set a static value for the following network parameters:

**Table 7:** Network Parameters Static Variables

| Variable | Default Static Value |
|---|---|
| localHostStaticPrimaryDns[a] | "192.168.0.10" |
| localHostStaticSecondaryDns[a] | "192.168.0.10" |
| localHostStaticSubnetMask | "255.255.255.0" |

a. If you do not want to use a DNS, set the variable to **0**.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *IP Configuration*.

**Caution:** These variables are vital to the proper operation of the Mediatrix 4102. If a variable of this group is not set properly, the unit may not be able to start or be contacted after it has started.

**Note:** Media5 recommends not to set the *localHostStaticSubnetMask* variable to 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

4. Set other static information as required.

   See "Services" on page 156 for more details.

   The next step would be to configure the IP routing information of the Mediatrix 4102 as described in "Chapter 13 - Transparent Address Sharing" on page 199.

### Special Vocal Features

When entering special characters on your telephone pad, the Mediatrix 4102 talks back to you with relevant information.

▶ **To access special vocal features:**

1. Take one of the telephones connected to the Mediatrix 4102.
2. Dial one of the digits sequence on the keypad.

**Table 8:** Special Vocal Features

| Digits to Dial | Information Vocally Sent by the Mediatrix 4102 |
|---|---|
| *#*0 | Current IP address of the Mediatrix 4102 (static or DHCP). |
| *#*1 | MAC address of the Mediatrix 4102. |
| *#*2 | Current WAN IP address of the Mediatrix 4102. |

### LED Behaviour in Starting Mode

When the Mediatrix 4102 starts and it is not configured to use a DHCP server, it uses static IP addresses. If the static information is not valid, the *LAN* LED blinks at 1 Hz with 75% duty cycle. This lets you know that you must perform a Factory reset or Recovery mode operation. See "Reset / Default Switch" on page 20 for more details.

## LED Indicators

A LED can be ON, OFF, BLINKING or controlled by hardware (HW). The blinking behaviour is described in terms of rate (in Hertz – Hz) and duty cycle (in percentage). For instance, a LED that turns on every two seconds and stays on for one second would be described as: blink 0.5 Hz 50%. The hardware (HW) behaviour is not defined. It is usually the standard state for the *ETH* LED.

### Ready LED

The *Ready* LED provides an "at-a-glance" view of the Mediatrix 4102 operational status. It is an aid for installation and on-site support. This LED is:

- ON when all elements of the *ifAdminOpState* column are "enabled".
- OFF when all elements of the *ifAdminOpState* column are "disabled".
- Blinking when at least one element of the *ifAdminOp State* column is "enabled" and at least one element is "disabled".

Patterns and meanings of the *Ready* LED are described in Table 11 on page 19.

Refer to the *MIB Reference Manual* for more details on the *ifAdminOpState* variable.

### In Use LED

The *In Use* LED provides feedback of the activity on the line. If a line is ringing, off-hook, or displaying information (ADSI), then this LED is ON. The *In Use* LED is ON when at least one element in the *ifAdminUsageState* column is "busy". Patterns and meanings of the *In Use* LED are described in Table 11 on page 19.

Refer to the *MIB Reference Manual* for more details on the *ifAdminState* variable.

## ETH LED

The *ETH* LED provides the Link and Heartbeat status of the network connected to the Ethernet connector. If there is no link under HW control, the LED is OFF. When a link is established, but no activity is detected, the LED is ON; it turns off for very short periods of time when activity is detected and blinks rapidly when the Ethernet is loaded. Patterns and meanings of the *ETH* LED are described in Table 11 on page 19.

## Power LED

The *Power* LED indicates whether or not the Mediatrix 4102 is operational at its most basic level. It does not imply that the unit can be used, only that it is capable of being used. Healthy operation would be steady ON. Patterns and meanings of the *Power* LED are shown in Table 11 on page 19.

## LED Patterns

Table 9 describes the different states a Mediatrix unit can have and their associated LED patterns.

**Table 9:** States and LED Patterns

| State | Description | LEDs Pattern | | | |
|---|---|---|---|---|---|
| | | **Ready** | **In Use** | **ETH** | **Power** |
| Booting | Follows a hardware start or a reset. | See "Booting LED Pattern Description" on page 18 | | | |
| Normal Mode | "Normal" state of the unit where calls can be initiated. Each LED has a separate behaviour. | See "NormalMode LED Pattern Description" on page 19 | | | |
| AdminMode | Calls are not permitted and maintenance actions can be performed. | See "AdminMode LED Pattern Description" on page 19 | | | |
| Recovery Mode | The IP addresses for local host, image server, syslog server, etc., are temporarily set to known values. Calls are not allowed. | Blink 1 Hz 75% | Off | Misc.[a] | Blink 1 Hz 75% |
| Reset Pending | Triggered when the *Reset / Default* switch is pressed and held for at least 2 seconds. | Off | Off | Off | Blink 1 Hz 50% |
| Reboot Pending | Triggered when the *Reset / Default* switch is pressed in either the *ResetPending* or *RecoveryMode Pending* states. The unit prepares for a physical shutdown and restart. | Off | Off | Off | Off |
| Recovery Mode Pending | Triggered when the *Reset / Default* switch is held at start-time or for at least 7 seconds. | Blink 1 Hz 50% | Blink 1 Hz 50% | Blink 1 Hz 50% | Blink 1 Hz 50% |
| Default Settings Pending | Triggered when the *Reset / Default* switch is not released while in *RecoveryModePending* state.<br>At run time, if the *Reset / Default* switch is released within 5 seconds, the unit applies default settings, otherwise the action is cancelled and the unit goes back to the Operation Modes state or it resets.<br>At start time, the unit stays in this state until the *Reset / Default* switch is released. The unit then applies the default settings and restarts. | On | On | On | On |
| Image DownloadIn Progress | A software image is downloaded into the unit and written to persistent storage. | LEDs are blinking at 1 Hz 75%, *one at a time*, from left to right. | | | |

**Table 9:** States and LED Patterns (Continued)

| State | Description | LEDs Pattern | | | |
|---|---|---|---|---|---|
| | | Ready | In Use | ETH | Power |
| Image Download Error | Triggered after a failure of an image download operation. After 4 seconds, the unit restarts. | Blink 2 Hz 50% | Blink 2 Hz 50% | Blink 2 Hz 50% | Blink 2 Hz 50% |
| InitFailed | Triggered when bad initialization parameters are detected and the unit cannot start correctly.<br>**Note**: If the network configuration is dynamic, the unit stays in the *Booting* state and continues to query the DHCP until it receives valid values. If the configuration is static, the LED pattern indicates that the unit must be reset to default settings or put into recovery mode for maintenance and correction of network values. | Off | Off | Blink 4 Hz 50% | Off |
| DiagFailed | Triggered at start-time when the hardware or software diagnostic fails. This is a critical error and the unit may require RMA. | Off | Off | Off | Blink 4 Hz 50% |
| NetworkRescue | The unit tries to download and install a firmware given by the Network Rescue server. | Off | LEDs are blinking to show a LED displacing light from left to right and right to left. | | |

a. See the corresponding LED pattern in "NormalMode LED Pattern Description" on page 19.

## *Booting* LED Pattern Description

While in the *Booting* state, the LEDs of the Mediatrix 4102 behave independently; the following table indicates the behaviour for each LED.

**Table 10:** LED Patterns in Booting Mode

| LED | Pattern | Meaning |
|---|---|---|
| Ready | Steady Off | Not Ready. |
| In Use | Steady Off | Cannot be in use. |
| ETH (HW Ctrl) | Steady On | Ethernet connection detected. |
| | Steady Off | Ethernet connection not detected or hardware control not activated. |
| | Blinking (variable rate) | Ethernet activity detected. |
| Power | Steady On | Power is On. |
| | Blinking 1 Hz 75% | Waiting for a DHCP answer. |

## *NormalMode* LED Pattern Description

While in the *NormalMode* state, the LEDs of the Mediatrix 4102 behave independently; the following table indicates the behaviour for each LED.

**Table 11:** LED Patterns in Operation Mode

| LED | Pattern | Meaning |
|---|---|---|
| Ready | Steady On | All lines are enabled (operational state). |
| | Steady Off | All lines are disabled (operational state). |
| | Blink 0.25 Hz 75% | At least one line is enabled and at least one line is disabled (operational state). |
| In Use | Steady On | At least one line is busy (usage state). |
| | Steady Off | All lines are not busy (usage state) or the unit is not connected to the network. |
| ETH (HW Ctrl) | Steady On | Ethernet connection detected. |
| | Steady Off | Ethernet connection not detected. |
| | Blinking (variable rate) | Ethernet activity detected. |
| Power | Steady On | Power is On. |
| | Steady Off | Power is Off. |
| | Blinking 1 Hz 75% | Waiting for a DHCP answer. |

## *AdminMode* LED Pattern Description

While in the *AdminMode* state, the LEDs of the Mediatrix 4102 behave independently; the following table indicates the behaviour for each LED.

**Table 12:** LED Patterns in AdminMode

| LED | Pattern | Meaning |
|---|---|---|
| Ready | Blinking 1 Hz 75% | *Ready* and *Power* LEDs blink off phase at 180 degrees. |
| In Use | Steady Off | All analog lines are not available. |
| ETH (HW Ctrl) | Steady On | Ethernet connection detected. |
| | Steady Off | Ethernet connection not detected. |
| | Blinking (variable rate) | Ethernet activity detected. |
| Power | Blinking 1 Hz 75% | *Ready* and *Power* LEDs blink off phase at 180 degrees. |

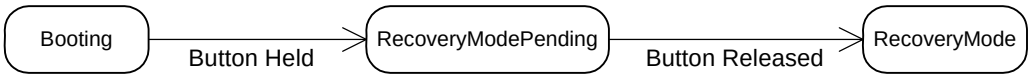
## Recovery Mode LED Patterns

There are two different sequences of LED patterns indicating that a recovery is in process.

### At Start-Time

When pressing the *Reset* / *Default* switch at start-time, the state sequence goes as follows:

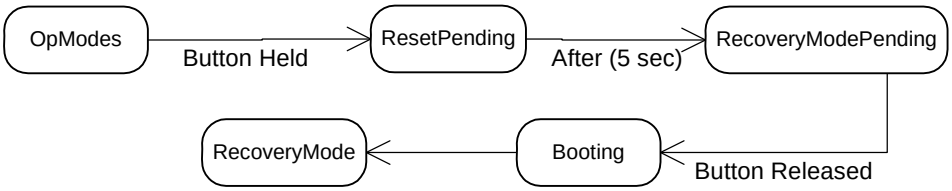
**Figure 8:** LED Pattern at Start-Time



### At Run-Time

When pressing the *Reset / Default* switch at run-time, the state sequence goes as follows:

**Figure 9:** LED Patterns at Run-Time




# Reset / Default Switch

The *Reset / Default* switch allows you to:

- Cancel an action that was started.
- Revert to known factory settings if the Mediatrix 4102 refuses to work properly for any reason or the connection to the network is lost.
- Reconfigure a unit.

## At Run-Time

The *Reset / Default* switch can be used at run-time – you can press the switch while the Mediatrix 4102 is running without powering the unit off. Table 13 describes the actions you can perform in this case.

**Table 13:** Reset / Default Switch Interaction

| Reset / Default Switch Pressed for: | Action | Comments | LEDs Pattern | | | |
|---|---|---|---|---|---|---|
| | | | Ready | In Use | ETH | Power |
| 2 to 5 seconds | Restarts the Mediatrix 4102 | No changes are made to the Mediatrix 4102 settings. | Off | Off | Off | Blink |
| 5 to 10 seconds | Restarts the Mediatrix 4102 in Recovery Mode | Sets the Mediatrix 4102 IP address to its default value in the MIB and restarts the unit. | Blink | Blink | Blink | Blink[a] |
| 10 to 15 seconds[b] | Restarts the Mediatrix 4102 in Factory Reset | Deletes the persistent MIB values, creates a new configuration file with the default factory values, and then restarts the unit. | On | On | On | On |

a. Synchronized blinking at 2 Hz (50% duty cycle).

b. You can disable the Factory reset procedure to avoid users deleting the existing configuration. See "Disabling the Factory Reset" on page 23 for more details.

## At Start-Time

The *Reset / Default* switch can be used at start-time – you power the unit off, and then depress the *Reset / Default* switch and power the unit back on. In this case, the following explains the reset behaviour:

- Pressing the *Reset / Default* switch at startup until all the LEDs start blinking restarts the Mediatrix 4102 in "Recovery Mode".

- Pressing the *Reset / Default* switch at startup until all the LEDs stop blinking and remain ON applies the “Factory Reset” procedure. This feature reverts the Mediatrix 4102 back to its default factory settings.

See “LED Indicators” on page 16 for a detailed description of the LED patterns related to the *Reset / Default* switch.

## Recovery Mode

The recovery mode restarts the Mediatrix 4102 in a known, static, and minimal state. It is used to recover from a basic configuration error that prevents you to reach the unit through the network. It may serve as a last resort before the Factory reset command. You must perform it in a closed network and on only one Mediatrix 4102 at a time, because the default IP address is the same on every unit.

The recovery mode is not intended to address configuration and/or software problems. For those types of problems, you must use the Factory reset.

**Note:** The procedure below assumes that you are performing it at run-time.

**To trigger the Recovery Mode:**

1. With a 10/100 Hub and two 10/100 BaseT Ethernet RJ-45 straight cables, connect both cables to the hub; one of them is connected into the *WAN* connector of the Mediatrix 4102 and the other one links the computer to the hub.

   Alternatively, you can connect a 10/100 BaseT Ethernet RJ-45 crossover cable into the *WAN* connector of the Mediatrix 4102 and connect the other end to your computer.

2. Reconfigure the IP address of your computer to *192.168.0.10* and enter the Subnet Mask of *255.255.255.0*. Restart the computer.

3. Insert a small, unbent paper clip into the *Reset / Default* switch hole located at the  of the Mediatrix 4102.

4. Hold the *Reset / Default* switch between 5 and 10 seconds – until the LEDs start blinking.

5. Release the paper clip.

   Only the *Power* and *Ready* LEDs should go on blinking to inform you that the recovery reset has been performed.

   In recovery mode, the provisioning source of the *localHostConfigSource* variable is set to **default**, meaning that the default factory setting is used.

   The following variables use their default values in the MIBs:

   - localHostAddress
   - localHostPrimaryDns
   - localHostSecondaryDns
   - localHostDefaultRouter
   - localHostSnmpPort
   - localHostSubnetMask
   - imagePrimaryHost
   - imagePrimaryPort
   - imageSecondaryHost
   - imageSecondaryPort
   - msHost
   - msTrapPort
   - syslogHost
   - syslogPort

   The following variables of the *mediatrixMgmt* group are all set to static:

   - imageConfigSource

- configFileFetchingConfigSource
- msConfigSource
- syslogConfigSource
- sntpConfigSource

All the persistent MIB values are kept.

In this mode, SIP is deactivated. Only SNMP or HTTP can be used to set the IP addresses listed above and the protocol-specific IP addresses (all IP addresses located under the *ipAddressConfig* folder in the MIB structure).

You can also download a software version, but you cannot download a configuration file.

6. When the Mediatrix 4102 has finished its provisioning sequence, perform the changes, and then turn it off, plug it on the network, and turn it on again.

   When restarting, the Mediatrix 4102 will not be in Recovery mode and will use the IP addresses configuration set forth in the MIBs.

   See "Changing a Parameter Value" on page 149 for more details.

**Note:** The recovery mode does not alter any persistent configuration data of the Mediatrix 4102.

## Factory Reset

The Factory reset reverts the Mediatrix 4102 back to its default factory settings. It deletes the persistent MIB values of the unit, including:

- The entire *mediatrixMIBs* configuration.
- The MIB-II setup.
- The software download configuration files.
- The SNMP configuration, including the SNMPv3 passwords and users.
- The PPPoE configuration, including the user names and passwords.

The Factory reset creates a new configuration file with the default factory values. It should be performed with the Mediatrix 4102 connected to a network with access to a DHCP server. If the unit cannot find a DHCP server, it sends requests indefinitely.

You can disable the Factory reset to avoid users deleting the existing configuration. See "Disabling the Factory Reset" on page 23 for more details.

**To trigger the Factory Reset:**

1. Power the Mediatrix 4102 off.
2. Insert a small, unbent paper clip into the *Reset / Default* switch hole located at the rear of the Mediatrix 4102. While depressing the *Reset / Default* switch, plug the power cord back in to power up the unit.

   Do not depress before all the LEDs stop blinking and are steadily ON.
3. Release the paper clip.

   The Mediatrix 4102 restarts.

   This procedure resets all variables in the MIB modules to their default value; defaults include the *localHostSelectConfigSource* variable set to **dhcp**.

   When the Mediatrix 4102 has finished its provisioning sequence, it is ready to be used with a DHCP-provided IP address and MIB parameters.

**Note:** The Factory reset alters any persistent configuration data of the Mediatrix 4102.

### Disabling the Factory Reset

You can disable the factory reset procedure, even if users depress the *Reset / Default* switch. Disabling the factory reset means that users will not be able to revert the Mediatrix 4102 back to its factory settings if there are configuration problems.

▶ **To change the factory reset behaviour:**

1. In the *sysAdminMIB*, set the *sysAdminDefaultSettingsEnable* variable to **disable**.

   In this case, users can only perform a Recovery Mode procedure. See "Reset / Default Switch" on page 20 for more details.

# Software Restart

You can initiate a software restart of the Mediatrix 4102 by using MIB parameters.

In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Restarting a Unit*.

▶ **To perform a software restart:**

1. In the *groupAdminMIB*, locate the *groupAdminMIBObjects* group.
2. Set the *groupSetAdmin* variable to the appropriate type of restart:
   - *Locked*: waits for the state of all lines to be locked, and then restarts. This is called a graceful restart.
   - *ForceLock*: restarts immediately. This is called an abrupt restart.
   - *Unlock*: the command is discarded.
3. Set the *groupReset* variable to **SoftReset**.

   The Mediatrix 4102 restarts.

## Restart Behaviour

This feature affects the behaviour of the Mediatrix 4102 when it restarts.

You can instruct the Mediatrix 4102 to check its TCP/IP stack before declaring the restart successful.

This could be useful when the unit is subjected to a broadcast storm (such as a TCP/IP flood) while it is restarting. In this case, and when the TCP/IP stack check is enabled, the unit enters into the rescue mode and cannot be contacted through SNMP. You thus need to restart the Mediatrix 4102 manually. However, when the TCP/IP stack check is disabled, a broadcast storm during a restart will cause the unit to continuously restart until the storm subsides.

▶ **To define the restart behaviour:**

1. In the *bootBehaviorMIB*, enable the *checkTcpIpStackForSuccessfulBoot* variable.

   When the variable is enabled, the TCP/IP stack must initialize properly to consider the restart a success. In a flood scenario, the unit may end up in the rescue mode.

   When the feature is disabled, even if the TCP/IP stack fails to initialize during a TCP/IP flood, the restart is considered successful and the unit does not enter into the rescue mode.

# Verifying the Installation

There are two ways to verify that the Mediatrix 4102 is properly connected to the IP network and is working:

- By contacting it with a SNMP Browser
- By pinging it

These two procedures assume that you know the IP address of the Mediatrix 4102 you want to verify. If the Mediatrix 4102 does not respond, do the following:

- Verify that the LAN cable is securely connected to the Mediatrix 4102 and to the network connector.
- Be sure that you did not connect a crossover network cable.
- Verify the state of the IP network to ensure it is not down (the  LED should be ON or blinking).

CHAPTER 2

# Web Interface – Introduction

The Mediatrix 4102 contains an embedded web server to set parameters by using the HTTP protocol. This web server may either be accessed via the LAN or WAN interface of the Mediatrix 4102, depending on the current access limitation. This access limitation may be modified in "Web Interface Access Limitation" on page 26.

| | |
|---|---|
| **Standards Supported** | RFC 2616 – Hypertext Transfer protocol - HTTP/1.1. |

## Introduction

The web interface may be used to:

- View the status of the Mediatrix 4102.
- Set the basic uplink parameters of the Mediatrix 4102.
- Peruse syslog messages the Mediatrix 4102 sends.
- Upload a configuration file to the Mediatrix 4102.
- Modify the password required to access the web interface.
- Set numerous parameters of the Mediatrix 4102 (in the Administration pages).

  All parameters located in the web interface may also be configured via SNMP. Each section of the web interface suggests a link to the corresponding SNMP section.

Before using the web-based configuration service, you must ensure that it is enabled.

### End User vs. Administration Pages

You can have access to two sets of web pages: Administration page and End User page. The Administration page offers a lot more parameters than the End User page. The access to each of these sites can be configured. For instance, you may grant a user access to the End User page, but not to the Administration page. See "Web Interface Access Limitation" on page 26 for more details on granting access to the web pages.

**To enable the web-based configuration service:**

1. In the *ipAddressConfig* folder, set the TCP port on which to listen for HTTP requests in the *httpServerPort* variable (under the *ipAddressConfigHttpEngine* group).

   This port number applies to the End User page.
2. Set the TCP port on which the Administration page listens for HTTP requests in the *httpServerAdminPort* variable.
3. In the *httpServerMIB*, enable the service by setting the *httpServerEnable* variable to **enable**.

# Using the Web Interface

Media5 recommends that you use the latest version of the Microsoft® Internet Explorer web browser to properly access the web interface.

## Web Interface Access Limitation

Access to the web interface can be limited to only one of the Mediatrix 4102's interface or all its interfaces. Furthermore, you can grant access to one or both of the Administration and End User web pages available. This can be modified only by using the proper MIB variable.

▶ **To limit the access to the web interface:**

1. In the *httpServerMIB*, configure the interface where the web pages can be accessed in the *httpServerAccess* variable.

   This port number applies to the End User page.

   You have the following choices:

   **Table 14:** Web Access Limitation Parameters

| Access | Description |
|---|---|
| lanOnly | You can access the web interface from the LAN side, which is usually associated with the *LAN* connector. |
| wanOnly | You can access the web interface from the WAN side, which is usually associated with the *WAN* connector. |
| all | You can access the web interface from both the LAN and WAN sides. |

**Note:** This variable is changed to **all** if Transparent Address Sharing is disabled. See "Configuring TAS" on page 205 for more details.

2. Configure the interface where the Administration web pages can be accessed in the *httpServerAdminAccess* variable.

   The choices are the same as for the End User web page.

3. Configure the realm, which identifies who requested the login information for the web-based configuration service, as follows:

   **Table 15:** Realm Parameters

| Web Page | Variable |
|---|---|
| End User | httpServerUserRealm |
| Administration | httpServerAdminRealm |

   The realm is usually presented by the user's browser when prompting for a user name and password. The default value is **default**.

4. Restart the Mediatrix 4102 so that the changes may take effect.

## Accessing the Web Interface

▶ **To use the web interface configuration:**

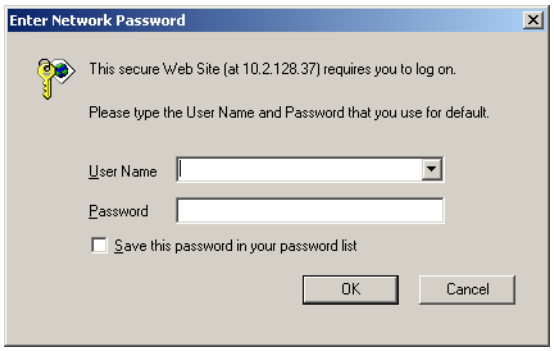
1. In your web browser's address field, type the default IP address of the Mediatrix 4102, which is **192.168.10.1** (if you have performed a recovery mode, this is **192.168.0.1** on the WAN port).

   If you want to access the Administration web page configured to be accessed on the WAN and port 8080, you must enter an URL similar to the following in your browser: **http://10.2.128.11:8080**.

It is also possible, with Administrator credentials, to access the End User web pages adding **/Res/** to the Administrator web page address: **http://10.2.128.11:8080/Res/**.

The following opens:

**Figure 10:** Login Window




The Mediatrix 4102 uses Digest Authentication to grant access to the web-based interface.

**2.** Enter the proper user name and password.

The user name and password must be valid. They are case sensitive hence they must be entered properly. Default factory values for the End User web page are:

- **User Name**: admin
- **Password**: 1234

Default factory values for the Administrator web page are:

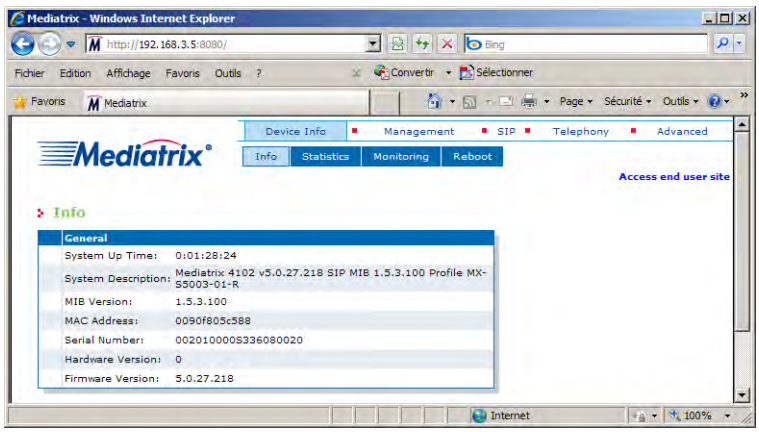
- **User Name**: root
- **Password**: 5678

Once you have accessed the web interface, you can change the End User page password as described in "HTTP Server Password Page" on page 40 and the Administration page password as described in "Admin Page" on page 43.

**3.** Click OK.

The *Overview* web page displays. It has two sub-pages: *System status* and *Network parameters*. It stays accessible for as long as the Internet browser used to access the Mediatrix 4102 web interface is opened.

**Figure 11:** Device Info Web Page



## System Status of the Mediatrix 4102 (End User Web Page)

The *Overview – System status* page displays the current system status of the Mediatrix 4102.

**Table 16:** System Status Page

| IP Information | Description |
|---|---|
| MAC Address | MAC address of the Mediatrix 4102. |
| Serial Number | Seriel number of the Mediatrix 4102. |
| Type | Product name. |
| Software version | Software version of the Mediatrix 4102. |
| Hardware version | Version of the analog circuit board of the Mediatrix 4102. |
| System description | A textual description of the Mediatrix 4102. It usually includes the full name and version identification of its hardware type, software operating-system, and networking software. |
| System uptime | The time (in hundredths of a second) since the network management portion of the system was last re-initialized. |
| Registration status port1<br><br>Registration status port2 | May have one of the following:<br>• ***Ready*** if the operational state of the current interface is enabled.<br>• ***Not ready*** if the component is operationally non-functional because of an internal condition that would not allow it to participate in a normal VoIP call. |

## Network Parameters Status of the Mediatrix 4102 (End User Web Page)

The *Overview – Network parameters* page displays the current network status of the Mediatrix 4102.

**Table 17:** Network Parameters Status Page

| IP Information | Description |
|---|---|
| WAN IP Address | Value of the *localHostWanAddress* MIB variable, which is the public IP address of the Mediatrix 4102. |
| WAN Network Mask | If the PPPoEservice is enabled, displays an empty string because there is no concept of network mask in PPP links. Otherwise, displays the value of the *localHostSubnetMask* MIB variable. |
| Default gateway | Value of the *localHostDefaultRouter* MIB variable. |
| Primary DNS | Value of the *localHostPrimaryDns* MIB variable. |
| Secondary DNS | Value of the *localHostSecondaryDns* MIB variable. |

## System Status of the Mediatrix 4102 (Administration Web Page)

The *Device Info – Info* page displays the current system status of the Mediatrix 4102.

**Table 18:** Info Page

| IP Information | Description |
|---|---|
| System Up Time | The time (in hundredths of a second) since the network management portion of the system was last re-initialized. |
| System Description | A textual description of the Mediatrix 4102. It usually includes the full name and version identification of its hardware type, software operating-system, and networking software. |

**Table 18:** Info Page (Continued)

| IP Information | Description |
|---|---|
| MIB Version | MIB version currently loaded in the Mediatrix 4102. |
| MAC Address | MAC address of the Mediatrix 4102. |
| Serial Number | Seriel number of the Mediatrix 4102. |
| Hardware version | Version of the analog circuit board of the Mediatrix 4102. |
| Firmware version | Software version of the Mediatrix 4102. |

## Menu Frame (End User Web Page)

The Menu frame is displayed at the top of the browser window. It contains management links that allow you to display web pages in the Content frame.

**Table 19:** Menu Frame Links

| Link | Description |
|---|---|
| Overview | ***System status***: Links to the *System status* sub-page of the *Overview* web page, which displays, in read-only format, the system parameters of the Mediatrix 4102. |
| | ***Network parameters***: Links to the *Network parameters* sub-page of the *Overview* web page, which displays, in read-only format, the IP addresses used by the Mediatrix 4102. |
| System | ***WAN***: Links to the *WAN* sub-page of the *System* web page, which allows you to set the uplink information used by the Mediatrix 4102. See "WAN Page" on page 35 for more details. |
| | ***LAN***: Links to the *LAN* sub-page of the *System* web page, which allows you to set the LAN interface of the Mediatrix 4102. See "LAN Page" on page 37 for more details. |
| | ***STUN***: Links to the *STUN* sub-page of the *System* page, which allows you to configure the STUN client of the Mediatrix 4102. See "STUN Page" on page 38 for more details. |
| Configuration File | Links to the *Configuration File upload* web page, which allows you to upload a configuration file from a computer to the Mediatrix 4102. See "Configuration File Upload Page" on page 39 for more details. |
| HTTP Server password | Links to the *HTTP server password* web page, which allows you to change the password. See "HTTP Server Password Page" on page 40 for more details. |
| System log | Links to the *System log* page, which displays, in read-only format, the syslog messages the Mediatrix 4102 sends. See "System Log Page" on page 42 for more details. |
| Reboot | Links to the *Reboot* page, which allows you to restart the Mediatrix 4102. |

## Menu Frame (Administration Web Page)

The Menu frame is displayed at the top of the browser window. It contains management links that allow you to display web pages in the Content frame.

**Table 20:** Menu Frame Links

| Link | Description |
|---|---|
| Device Info | ***Info***: Links to the *Info* sub-page of the *Device Info* web page, which displays, in read-only format, the system parameters of the Mediatrix 4102. |
| | ***Statistics***: Links to the *Statistics* sub-page of the *Device Info* web page, which displays, in read-only format, various communication statistics of the Mediatrix 4102. |
| | ***Monitoring***: Links to the *Monitoring* sub-page of the *Device Info* web page, which allows you to configure the syslog parameters of the Mediatrix 4102. See "Syslog Monitoring" on page 32 for more details. |
| Management | ***Admin***: Links to the *Admin* sub-page of the *Management* web page, which allows you to configure line administration parameters of the Mediatrix 4102. See "Admin Page" on page 43 for more details. |
| | ***Network Settings***: Links to the *Network Settings* sub-page of the *Management* web page, which allows you to configure network-related parameters of the Mediatrix 4102, such as IP address source, IP address, etc. See "Network Settings" on page 49 for more details. |
| | ***Configuration File***: Links to the *Configuration File* sub-page of the *Management* web page, which allows you to configure the various configuration file download parameters of the Mediatrix 4102. See "Configuration File Download" on page 54 for more details. |
| | ***Firmware Download***: Links to the *Firmware Download* sub-page of the *Management* web page, which allows you to configure the various firmware download parameters of the Mediatrix 4102. See "Firmware Download" on page 65 for more details. |
| SIP | ***Configuration***: Links to the *Configuration* sub-page of the *SIP* web page, which allows you to configure the SIP server and SIP user agent parameters of the Mediatrix 4102. See "SIP Servers Configuration" on page 77 for more details. |
| | ***Interop***: Links to the *Interop* sub-page of the *SIP* web page, which allows you to configure the various interoperability features of the Mediatrix 4102. See "SIP Interop" on page 81 for more details. |
| | ***Authentication***: Links to the *Authentication* sub-page of the *SIP* web page, which allows you to configure authentication parameters of the Mediatrix 4102. See "SIP Authentication" on page 85 for more details. |

**Table 20:** Menu Frame Links (Continued)

| Link | Description |
|---|---|
| Telephony | ***Digit Maps***: Links to the *Digit Maps* sub-page of the *Telephony* web page, which allows you to configure the various digit maps of the Mediatrix 4102. See "Digit Maps" on page 87 for more details. |
| | ***CODEC***: Links to the *CODEC* sub-page of the *Telephony* web page, which allows you to configure the voice and data codec-related parameters of the Mediatrix 4102. See "Voice & Fax Codecs" on page 93 for more details. |
| | ***Call Forward***: Links to the *Call Forward* sub-page of the *Telephony* web page, which allows you to configure the call forward on busy, on no answer, and unconditional parameters of the Mediatrix 4102. See "Call Forward" on page 104 for more details. |
| | ***Services***: Links to the *Services* sub-page of the *Telephony* web page, which allows you to configure the subscriber services of the Mediatrix 4102. See "Services" on page 109 for more details. |
| | ***Misc***: Links to the *Misc* sub-page of the *Telephony* web page, which allows you to configure advanced telephony attributes of the Mediatrix 4102. See "Miscellaneous" on page 117 for more details. |
| Advanced | ***QoS***: Links to the *QoS* sub-page of the *Advanced* web page, which allows you to configure the Quality of Service parameters of the Mediatrix 4102. See "Quality of Service (QoS)" on page 129 for more details. |
| | ***Emergency***: Links to the *Emergency* sub-page of the *Advanced* web page, which allows you to configure the Emergency Call parameters of the Mediatrix 4102. See "Emergency Page" on page 132 for more details. |
| | ***STUN***: Links to the *STUN* sub-page of the *Advanced* page, which allows you to configure the STUN client of the Mediatrix 4102. See "STUN Configuration" on page 133 for more details. |

### Content Frame

The Content frame is displayed in the lower part of the browser window. It contains the various web pages that allow you to manage the Mediatrix 4102.

# Submitting Changes

When you perform changes in the web interface and click the *Submit* button, the Mediatrix 4102 validates the changes. A message informs you that the Mediatrix 4102 must be restarted if one or more non-dynamic value was changed. If at least one modified value is invalid, a message informs you that some values are invalid. Most changes are not dynamic and require to restart the Mediatrix 4102.

The Reboot page then opens. You must click *Reboot*.

This restarts the Mediatrix 4102. If the unit is in use when you click *Reboot*, all calls are terminated.

# Syslog Monitoring

| **Standards Supported** | RFC 3164 – The BSD Syslog Protocol |
|---|---|

The *Monitoring* sub-page of the *Device Info* page allows you to set the Syslog daemon configuration of the Mediatrix 4102. You can also set these parameters and supplemental information via SNMP, as described in "Chapter 29 - Syslog Daemon" on page 371.

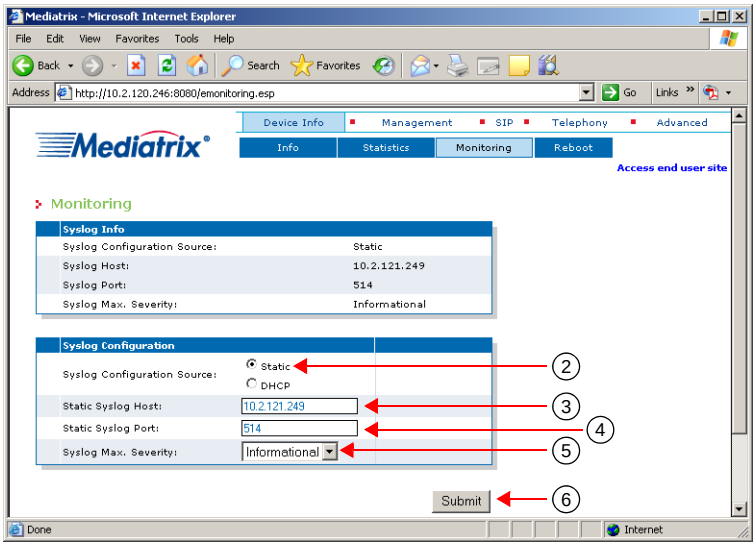
The Syslog daemon is a general purpose utility for monitoring applications and network devices with the TCP/IP protocol. With this software, you can monitor useful messages coming from the Mediatrix 4102 unit. If no Syslog daemon address is provided by a DHCP server or specified by the administrator, no messages are sent.

For instance, if you want to download a new software into the Mediatrix 4102, you can monitor each step of the software download phase. Furthermore, if the unit encounters an abnormal behaviour, you may see accurate messages that will help you troubleshoot the problem.

### ▶ To set the syslog parameters:

1. In the web interface, click the *Device Info* link, then the *Monitoring* sub-link.

**Figure 12:** Device Info – Monitoring Web Page




2. Select the configuration source of the syslog information in the *Syslog Configuration Source* choices.

**Table 21:** Syslog Configuration Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

3. If the Syslog configuration source is **Static**, enter the Syslog server static IP address or domain name in the *Static Syslog Host* field.

4. If the Syslog configuration source is **Static**, enter the Syslog server static IP port number in the *Static Syslog Port* field.

5. Set the syslog severity level in the *Syslog Max. Severity* choices.

   This indicates which syslog message is processed. Any syslog message with a severity value greater than the selected value is ignored by the agent.

   - Disabled
   - Critical
   - Error
   - Warning
   - Informational
   - Debug

   A higher level mask includes lower level masks, e.g., *Warning* includes *Error* and *Critical*.

6. Click *Submit* if you do not need to set other parameters.

**Note:** The current syslog information is displayed in the *Syslog Info* section.

The following are some of the messages the unit sends:

**Table 22:** Syslog Messages Examples

| Event | Level | Message |
|---|---|---|
| The configuration update with the specific configuration file has been successful (configuration file fetching) | Informational | `The specific configuration update succeeded.` |
| The configuration update with the specific configuration file experienced an error and has not been completed (configuration file fetching) | Error | `The specific configuration update failed.` |
| The software update has been successful | Informational | `The software update succeeded.` |
| The software update experienced an error and has not been completed | Error | `The software update failed.` |

## Configuring the Syslog Daemon Application

You shall configure the Syslog daemon to capture those messages. Refer to your Syslog daemon's documentation to learn how to properly configure it to capture messages.

CHAPTER 3

# Web Interface

This chapter describes the parameters you can set with the web interface.

## System Page

The *System* page allows you to set the following information:

- WAN uplink connection
- LAN interface information
- STUN parameters

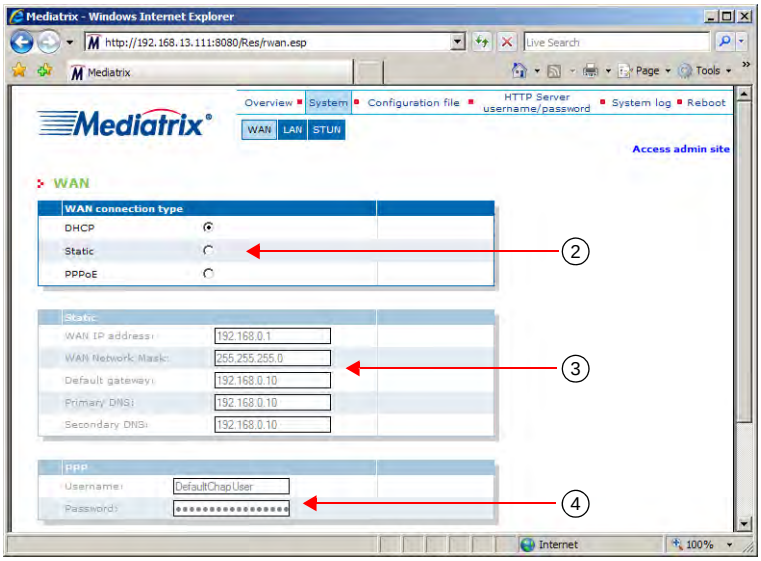
### WAN Page

The *WAN* page allows you to set the uplink information used by the Mediatrix 4102.

**To set WAN parameters:**

1. In the web interface, click the *System* link.
   This links to the *System – WAN* web page.

**Figure 13:** System – WAN Web Page



**2.** Set the *WAN connection type*.

You have the following choices:

**Table 23:** WAN Connection Type

| Connection Type | Description |
|---|---|
| DHCP | **Connection to use with a cable modem.**<br>This is the most common connection type with a cable modem. In this connection type, the PPPoE service (*pppoeEnable* variable) disabled and the configuration source (*localHostSelectConfigSource* variable) is set to "DHCP".<br>However, some locations may require to manually enter static IP information instead. If this is the case, select the **Static** connection type and proceed to Step 3. |
| Static | This connection type may be used for locations where cannot use the DHCP connection type. You are thus required to manually provide the IP information. See Step 3.<br>In this connection type, the PPPoE service (*pppoeEnable* variable) is disabled and the configuration source (*localHostSelectConfigSource* variable) is set to "static". |
| PPPoE | **Connection to use with a DSL modem.**<br>This is the most common connection type with a DSL modem. In this connection type, the PPPoE service (*pppoeEnable* variable) is enabled and the configuration source (*localHostSelectConfigSource* variable) is set to "static".<br>However, some DSL modems may require that you use the DHCP connection type instead. |

**3.** If the WAN connection type is **Static** (as set in Step 2), enter the following static IP information.

**Table 24:** IP Addresses Parameters

| Parameter | Definition |
|---|---|
| WAN IP Address | Public IP address of the Mediatrix 4102. This address is used for incoming signalling, media and management traffic. |
| WAN Network Mask | Subnet mask IP address used by the Mediatrix 4102. The subnet mask enables the network administrator to further divide the host part of the address into two or more subnets.<br>**Note:** Media5 recommends not to set a network mask of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts. |
| Default gateway | Default router IP address used by the Mediatrix 4102. A router is a device that connects any number of LANs. |
| Primary DNS | Primary Domain Name Server IP address used by the Mediatrix 4102. A DNS is an Internet service that translates domain names into IP addresses. |
| Secondary DNS | Secondary Domain Name Server IP address used by the Mediatrix 4102. |

**4.** If the WAN connection type is **PPPoE** (as set in Step 2), set the PPP user name and password.

When connecting to an access concentrator, it may request that the Mediatrix 4102 identifies itself with a specific user name and password.

There are no restrictions, you can use any combination of characters.

5. Click *Submit* if you do not need to set other parameters.

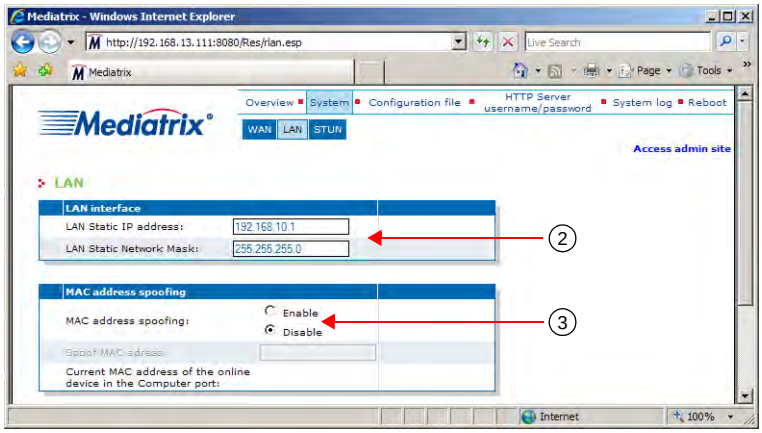
## LAN Page

The *LAN* page allows you to set the LAN interface of the Mediatrix 4102.

▶ **To set LAN parameters:**

1. In the *System* pages, click the *LAN* link.
   This links to the *System – LAN* web page.

**Figure 14:** LAN Web Page




2. Set the static IP address and network mask of the LAN interface.
   This is the IP information of the *Computer* connector, where you connect the PC or other IP equipment. The LAN interface is normally used to connect a PC that will have access to the WAN by sharing the Mediatrix 4102 WAN address. See "LAN Interface" on page 206 for more details.

**Note:** Media5 recommends not to set a network mask of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

3. If applicable, enable the MAC Address Spoofing feature.
   Spoofing the MAC address is useful in the case of ISPs that use the MAC address of the device connected to the *Computer* interface of the Mediatrix 4102 (e.g., a PC) to identify the connection.
   Enter the proper MAC address in the *Spoof MAC address* field. The current MAC address of the online device in the *Computer* connector is displayed below the field. A valid MAC address is a series of 12 alphanumeric characters without colons. See "MAC Address Spoofing" on page 207 for more details.
   The following MAC addresses are not allowed:
   - 000000000000
   - FFFFFFFFFFFF
   - 01xxxxxxxxxx, where x can be any digit or letter
4. Click *Submit* if you do not need to set other parameters.

## STUN Page

| | |
|---|---|
| **Standards Supported** | RFC 3489 – STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) |

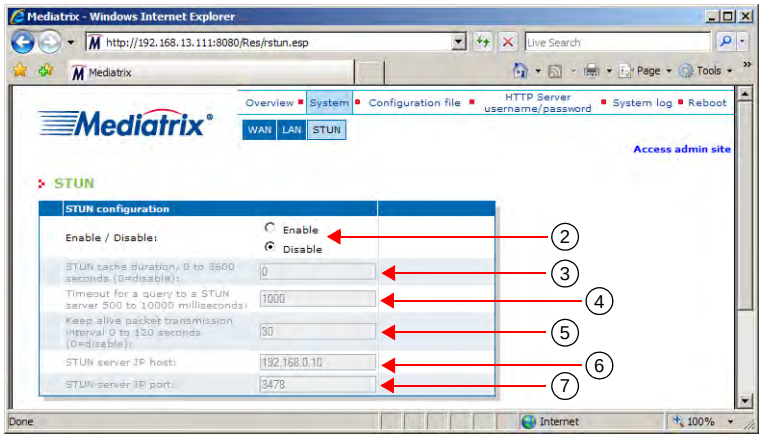
The *STUN* web page allows you to configure the STUN client of the Mediatrix 4102. See "Chapter 20 - STUN Configuration" on page 315 for more information.

### ▶ To set STUN parameters:

1. In the *System* pages, click the *STUN* link.

   This links to the *System – STUN* web page.

**Figure 15:** STUN Web Page




2. Enable the STUN client by selecting the *Enable* option.
3. Set the amount of time, in seconds, the Mediatrix 4102 should keep a STUN query result in its internal cache in the *STUN cache duration* field.

   Keeping a query in a cache helps to reduce the amount of unnecessary STUN queries when an RTP or T.38 socket is re-used within a short period of time. Available values range from 0 s to 3600 s.

   When set to **0**, the cache is disabled and the unit performs a STUN query each time a socket needs to be used.
4. Set the maximum amount of time, in milliseconds, the Mediatrix 4102 should wait for an answer to a STUN query sent to a STUN server in the *Timeout for a query to a STUN server* field.

   Available values range from 500 ms to 10000 ms.

   Caution is advised in setting long timeouts. In the advent of an unresponsive STUN server, the unit may end up waiting a long time before it determines that a call cannot be made due to the STUN server failure.
5. Define the interval, in seconds, at which the Mediatrix 4102 sends blank keepalive messages to keep a firewall hole opened in the *Keepalive packet transmission interval* field.

   Keepalive messages are used by both the signalling protocol socket and the RTP socket to keep those connections opened through a firewall. Available values range from 0 s to 120 s.

   When set to **0**, no keepalive packet is sent.

**Note:** Keepalive messages are not supported on the T.38 socket.

6. Set the static STUN server IP address or Fully Qualified Domain Name (FQDN) in the *STUN server IP host* field.
7. Set the static STUN server IP port number in the *STUN server IP port* field.

   The default value is **3478**.

**8.** Click *Submit*.

# Configuration File Upload Page

The *Configuration file upload* web page allows you to upload a configuration file from the computer connected to the *Computer* connector of the Mediatrix 4102 into the unit.

The configuration file is transferred by using the HTTP POST method.

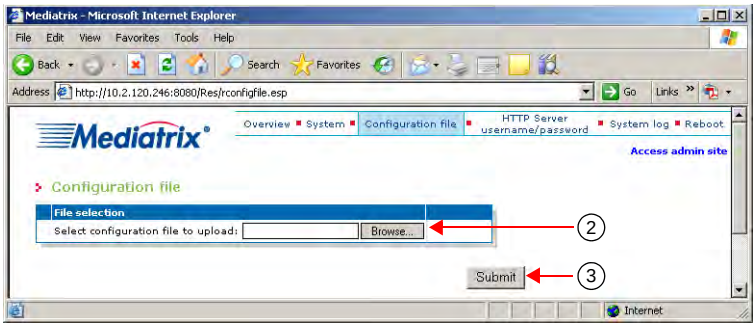
See "Chapter 14 - Configuration File Download" on page 219 for more information on configuration files.

### ▶ To upload a configuration file:

**1.** In the web interface, click the *Configuration file* link.

This links to the *Configuration file upload* web page.

**Figure 16:** Configuration file upload Web Page




**2.** Select the configuration file you want to upload to the Mediatrix 4102 in the *Select configuration file to upload* field.

You can click the *Browse* button to select a file. This button may not be available, depending on the web browser you are using.

**3.** Click the *Submit* button.

If a valid configuration file is successfully uploaded, then the Mediatrix 4102 automatically restarts to apply all the new settings. If the Mediatrix 4102 does not restart, this could mean the upload failed.

# HTTP Server Password Page

| | |
|---|---|
| **Standards Supported** | RFC 2617 – HTTP Authentication: Basic and Digest Access Authentication |

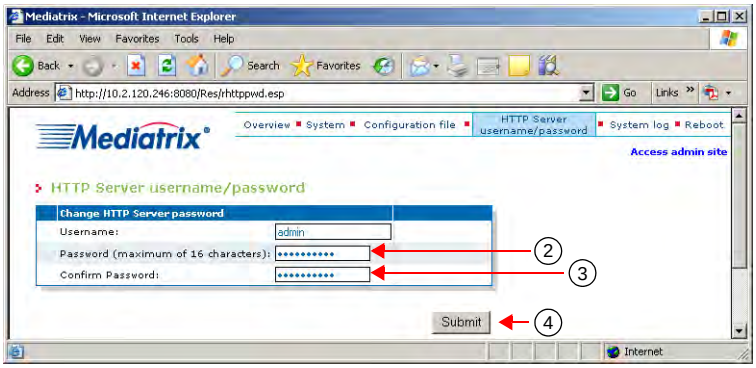
The *HTTP server password* page allows you to modify the default password to access the web interface. The Mediatrix 4102 supports basic HTTP authentication, as described in RFC 2617.

▶ **To change the password:**

1. In the web interface, click the *HTTP server password* link.

   The following opens:

**Figure 17:** HTTP server password Page




2. Enter the new password.

   The password is case sensitive. It can be a string of 0 to 16 characters. All characters are allowed. However, some special characters, such as accented characters (é, à, etc.), may not work.

3. Retype the password in the *Confirm Password* field.

4. Click *Submit*.

   The password resets back to the default value when:

   - Resetting the password by using the *httpServerResetToDefaultPwd* variable (see "Default User Name and Password" on page 40 for more details).
   - Performing a factory reset (see "Factory Reset" on page 22 for more details).

## Default User Name and Password

The default user name and password the web interface uses are stored in MIB variables you can modify.

▶ **To modify the default user name and password:**

1. In the *httpServerMIB*, set the default user name for the web interface access authentication in the *httpServerUsername* variable.

2. Set the default password for the web interface access authentication in the *httpServerDefaultPassword* variable.

   Both changes are immediate and take effect on all new web accesses.

▶ **To reset the web authentication password to the default value:**

1. In the *httpServerMIB*, set the *httpServerResetToDefaultPwd* variable to **reset**.

   The web password is reset to the default value specified by the *httpServerDefaultPassword* variable.

2. Restart the Mediatrix 4102 so that the change may take effect.

## Issue: Factory Reset does not Reset the Default Password Value

The following describes three cases in which the factory reset may not properly reset the HTTP server password. Each case defines the password you must use to access the web interface.

### Case #1

You do not modify the password via the Web page and you upgrade to a new software version with a new default password in the profile.

**Table 25:** Case 1 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the new profile. |
| Current Behaviour | The password to use is the default password in the ***previous*** version of the profile. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultPwd* variable as described in "Default User Name and Password" on page 40. |

### Case #2

You modify the password via the web interface and you upgrade to a new software version with a new default password in the profile.

**Table 26:** Case 2 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the password modified via the web interface in the older software version. |
| Current Behaviour | Same as the wanted behaviour. |
| Workaround | None required. |

### Case #3

You are performing a factory reset.

**Table 27:** Case 3 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the profile. |
| Current Behaviour | Same as the wanted behaviour. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultPwd* variable as described in "Default User Name and Password" on page 40. |

# System Log Page

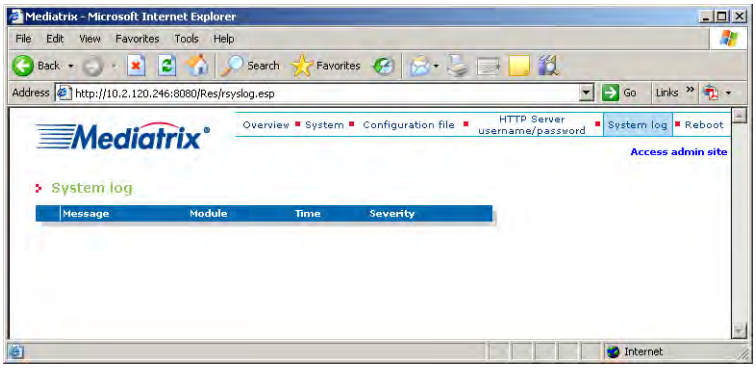
The *System log* page allows you to peruse the last *n* system log (syslog) messages sent by the Mediatrix 4102 since it last restarted.

▶ **To access the System log page:**

1. In the web interface, click the *System log* link.

   The following opens:

**Figure 18:** System log Page




Please refer to "Local Syslog" on page 374 for information on how to set the system log parameters.

CHAPTER 4

# Web Interface – Management

The *Management* section of the web interface allows you to configure general parameters of the Mediatrix 4102, as well as its configuration file download and firmware download parameters.

## Admin Page

The *Admin* sub-page of the *Management* page allows you to configure line administration parameters of the Mediatrix 4102 grouped in four categories:

- HTTP Server Username/Password
- System Management
- Group Port Management
- Interface Management

### HTTP Server Password – Administrator Web Page

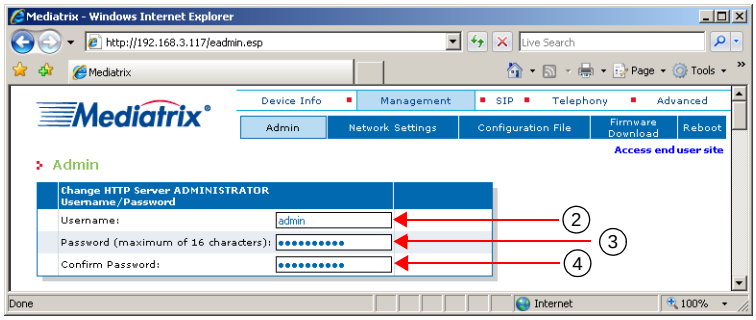
| | |
|---|---|
| **Standards Supported** | RFC 2617 – HTTP Authentication: Basic and Digest Access Authentication |

The *Change HTTP Server ADMINISTRATOR Username/Password* section allows you to modify the default password to access the web interface. The Mediatrix 4102 supports basic HTTP authentication, as described in RFC 2617.

**To change the username/password:**

1. In the web interface, click the *Management* link, then the *Admin* sub-link.

**Figure 19:** Management – Admin Web Page




2. Enter the new user name.
3. Enter the new password.

   The password cannot begin with “0” or exclusively be made up of several “0”.

   The password is case sensitive. It can be a string of 0 to 16 characters. All characters are allowed. However, some special characters, such as accented characters (é, à, etc.), may not work.
4. Retype the password in the *Confirm Password* field.

5. Click *Submit* if you do not need to set other parameters.

   The password resets back to the default value when:

   - Resetting the password by using the *httpServerResetToDefaultAdminPwd* variable (see "Default User Name and Password" on page 44 for more details).
   - Performing a factory reset (see "Factory Reset" on page 22 for more details).

## Default User Name and Password

The default user name and password the Administrator web interface uses are stored in MIB variables you can modify.

▶ **To modify the default user name and password:**

1. In the *httpServerMIB*, set the default user name for the web interface access authentication in the *httpServerAdminUsername* variable.
2. Set the default password for the web interface access authentication in the *httpServerDefaultAdminPassword* variable.

   Both changes are immediate and take effect on all new web accesses.

▶ **To reset the web authentication password to the default value:**

1. In the *httpServerMIB*, set the *httpServerResetToDefaultAdminPwd* variable to **reset**.

   The web password is reset to the default value specified by the *httpServerDefaultAdminPassword* variable.
2. Restart the Mediatrix 4102 so that the change may take effect.

## Issue: Factory Reset does not Reset the Default Password Value

The following describes three cases in which the factory reset may not properly reset the HTTP server password. Each case defines the password you must use to access the web interface.

### *Case #1*

You do not modify the password via the Web page and you upgrade to a new software version with a new default password in the profile.

**Table 28:** Case 1 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the new profile. |
| Current Behaviour | The password to use is the default password in the ***previous*** version of the profile. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultAdminPwd* variable as described in "Default User Name and Password" on page 44. |

### *Case #2*

You modify the password via the web interface and you upgrade to a new software version with a new default password in the profile.

**Table 29:** Case 2 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the password modified via the web interface in the older software version. |
| Current Behaviour | Same as the wanted behaviour. |

**Table 29:** Case 2 Issue (Continued)

| Item | Description |
|---|---|
| Workaround | None required. |

*Case #3*

You are performing a factory reset.

**Table 30:** Case 3 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the profile. |
| Current Behaviour | Same as the wanted behaviour. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultAdminPwd* variable as described in "Default User Name and Password" on page 44. |

## HTTP Server Password – End User Web Page

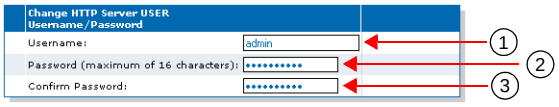
| **Standards Supported** | RFC 2617 – HTTP Authentication: Basic and Digest Access Authentication |
|---|---|

The *Change HTTP Server USER Username/Password* section allows you to modify the default password to access the End User web interface. The Mediatrix 4102 supports basic HTTP authentication, as described in RFC 2617.

▶ **To change the username/password:**

1. In the *HTTP Server USER Username/Password* section of the *Admin* page, enter the new user name.

**Figure 20:** User Username/Password – Admin Web Page




2. Enter the new password.

   The password is case sensitive. It can be a string of 0 to 16 characters. All characters are allowed. However, some special characters, such as accented characters (é, à, etc.), may not work.

3. Retype the password in the *Confirm Password* field.

4. Click *Submit* if you do not need to set other parameters.

   The password resets back to the default value when:

   - Resetting the password by using the *httpServerResetToDefaultPwd* variable (see "Default User Name and Password" on page 46 for more details).
   - Performing a factory reset (see "Factory Reset" on page 22 for more details).

## Default User Name and Password

The default user name and password the End User web interface uses are stored in MIB variables you can modify.

**To modify the default user name and password:**

1. In the *httpServerMIB*, set the default user name for the web interface access authentication in the *httpServerUsername* variable.
2. Set the default password for the web interface access authentication in the *httpServerDefaultPassword* variable.

   Both changes are immediate and take effect on all new web accesses.

**To reset the web authentication password to the default value:**

1. In the *httpServerMIB*, set the *httpServerResetToDefaultPwd* variable to **reset**.

   The web password is reset to the default value specified by the *httpServerDefaultPassword* variable. The change is immediate and takes effect on all new web accesses.

## Issue: Factory Reset does not Reset the Default Password Value

The following describes three cases in which the factory reset may not properly reset the HTTP server password. Each case defines the password you must use to access the web interface.

### *Case #1*

You do not modify the password via the Web page and you upgrade to a new software version with a new default password in the profile.

**Table 31:** Case 1 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the new profile. |
| Current Behaviour | The password to use is the default password in the ***previous*** version of the profile. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultPwd* variable as described in "Default User Name and Password" on page 46. |

### *Case #2*

You modify the password via the web interface and you upgrade to a new software version with a new default password in the profile.

**Table 32:** Case 2 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the password modified via the web interface in the older software version. |
| Current Behaviour | Same as the wanted behaviour. |
| Workaround | None required. |

*Case #3*

You are performing a factory reset.

**Table 33:** Case 3 Issue

| Item | Description |
|---|---|
| Wanted Behaviour | The password to use is the default password in the profile. |
| Current Behaviour | The password to use is the same as the one used in the older software version. |
| Workaround | Once using the new software version, use the *httpServerResetToDefaultPwd* variable as described in "Default User Name and Password" on page 46. |

## System Management

The following are the system management parameters you can set. These parameters apply to the whole Mediatrix 4102.

You can also set these parameters via SNMP, as described in "Chapter 8 - MIB Structure and SNMP" on page 139.

▶ **To set the system management parameters:**

1. In the *System Management* section of the *Admin* page, select the proper command to execute in the *System Command* drop-down menu.

**Figure 21:** Management – System Management Web Page

This command controls the various commands that can be performed by the unit.

**Table 34:** System Commands

| Command | Description |
|---|---|
| noOp | No action is taken. |
| checkRam | Launches the RAM check routine. |
| checkRom | Currently unused. |
| downloadSoftware | Launches a firmware update. See "Firmware Download" on page 65 for more details. |
| resetStats | Resets all cumulated call statistics. |
| setConfigSourcesStatic | Sets all configuration sources supported by the unit to "static". This command can be used when no DHCP server is present in the network to easily configure the unit to use static values. |
| updateConfiguration | Downloads configuration files. See "Configuration File Download" on page 54 for more details. |

2. Click *Submit* if you do not need to set other parameters.

## Group Port Management

You can set the administrative state of all the lines of the Mediatrix 4102.

▶ **To set the group port management parameters:**

1. In the *Group Port Management* section of the *Admin* page, select the proper command to execute in the *Group Port Command* field.

**Figure 22:** Group Port Management Section

This command locks/unlocks all the lines of the Mediatrix 4102. This state is kept until the unit restarts. It offers the following settings:

**Table 35:** Group Port Settings

| Setting | Description |
|---|---|
| noOp | No action is taken. |
| unlockAllPorts | Registers all the lines to the SIP server. |
| lockAllPorts | Cancels all the lines registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated. |

2. Click *Submit* if you do not need to set other parameters.

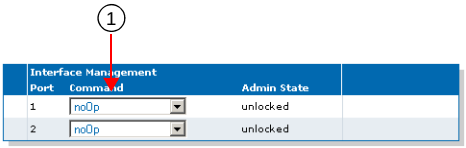
## Interface Management

You can set the administrative state of a line that will be kept until the Mediatrix 4102 restarts.

▶ **To set the interface management parameters:**

1. In the *Interface Management* section of the *Admin* page, select the proper command to execute in the *Command* column.

**Figure 23:** Interface Management Section




This command temporary locks/unlocks the selected line of the Mediatrix 4102. This state is kept until the unit restarts. It offers the following settings:

**Table 36:** Temporary Lock Settings

| Setting | Description |
|---|---|
| noOp | No action is taken. |
| unlock | Registers the line to the SIP server. |
| lock | Cancels the line registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated. |
| forcelock | Cancels the line registration to the SIP server. All active calls in progress are terminated immediately. No new calls may be initiated. |

2. Click *Submit* if you do not need to set other parameters.

# Network Settings

The *Network Settings* sub-page of the *Management* page allows you to configure network-related parameters of the Mediatrix 4102 grouped in three categories:

- Ethernet
- Network Settings
- SNTP

## Ethernet Connection Speed

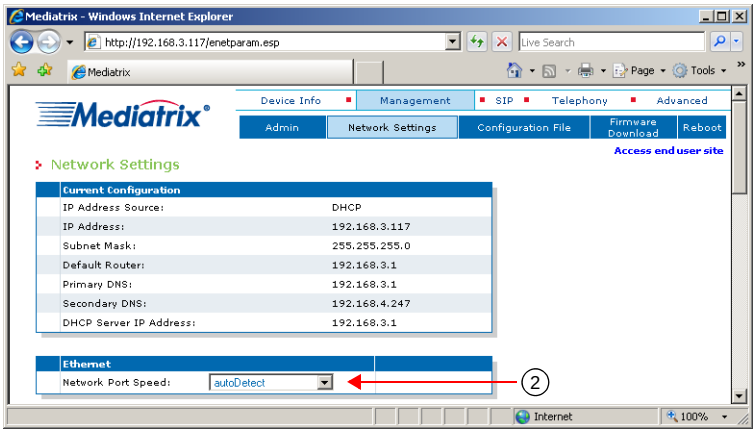
You can set the speed of the Mediatrix 4102's Ethernet connection.

You can also set this parameter via SNMP, as described in "Ethernet Connection Speed" on page 176.

**To set the Ethernet connection speed:**

1. In the web interface, click the *Management* link, then the *Network Settings* sub-link.

**Figure 24:** Management – Network Settings Web Page




2. Set the Ethernet connection speed of the *WAN* connector in the *Network Port Speed* field.

   The following values are available:

   - Auto detect
   - 10Mbs-HalfDuplex
   - 100Mbs-HalfDuplex
   - 10Mbs-FullDuplex
   - 100Mbs-FullDuplex

   A half-duplex connection refers to a transmission using two separate channels for transmission and reception, while a full-duplex connection refers to a transmission using the same channel for both transmission and reception.

   If unknown, select **Auto detect** so that the Mediatrix 4102 can automatically detect the network speed.

**Caution:** Whenever you force a connection speed / duplex mode, be sure that the other device and all other intermediary nodes used in the communication between the two devices have the same configuration. See "Speed and Duplex Detection Issues" on page 50 for more details.

3. Click *Submit* if you do not need to set other parameters.

### Speed and Duplex Detection Issues

There are two protocols for detecting the Ethernet link speed:

- An older protocol called parallel detection.
- A more recent protocol called auto-negotiation (IEEE 802.3u).

The auto-negotiation protocol allows to detect the connection speed and duplex mode. It exchanges capabilities and establishes the most efficient connection. When both endpoints support the auto-negotiation, there are no problems. However, when only one endpoint supports auto-negotiation, the parallel detection protocol is used. This protocol can only detect the connection speed; the duplex mode cannot be detected. In this case, the connection may not be established.

The Mediatrix 4102 has the possibility to force the desired Ethernet link speed and duplex mode by disabling the auto-negotiation and selecting the proper setting. When forcing a link speed at one end, be sure that the other end (a hub, switch, etc.) has the same configuration. To avoid any problems, the link speed and duplex mode of the other endpoint must be exactly the same.

## Network Settings

The *Network Settings* section allows you to set the IP information the Mediatrix 4102 needs to work properly. This section is vital to the proper operation of the Mediatrix 4102. If a field of this group is not properly set, the Mediatrix 4102 may not be able to restart and be contacted after it has restarted.

You can also set this parameter via SNMP, as described in "Chapter 9 - IP Address and Network Configuration" on page 155.

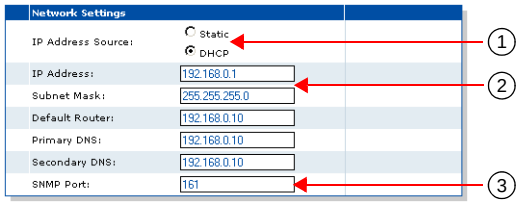
**To set network parameters:**

1. In the *Network Settings* section of the *Network Settings* page, select the configuration source of the network information in the *IP Address Source* choices.

**Table 37:** Network Settings Configuration Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

**Figure 25:** Network Settings Section




2. If the IP address source is **Static**, enter the following static IP information.

**Table 38:** IP Addresses Parameters

| Parameter | Definition |
|---|---|
| IP Address | Public IP address of the Mediatrix 4102. This address is used for incoming signalling, media and management traffic. |

**Table 38:** IP Addresses Parameters (Continued)

| Parameter | Definition |
|---|---|
| Subnet Mask | Subnet mask IP address used by the Mediatrix 4102.<br>**Note:** Media5 recommends not to set a subnet mask of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts. |
| Default Router | Default router IP address used by the Mediatrix 4102. |
| Primary DNS | Primary Domain Name Server IP address used by the Mediatrix 4102. |
| Secondary DNS | Secondary Domain Name Server IP address used by the Mediatrix 4102. |

3. Enter the default SNMP agent port in the *SNMP Port* field.
   This is the port number to use to reach the local host via the SNMP protocol.
4. Click *Submit* if you do not need to set other parameters.

**Note:** The current configuration file server information is displayed in the *Current Configuration* section.

## SNTP Settings

| **Standards Supported** | RFC 1769 – Simple Network Time Protocol (SNTP) |
|---|---|

The Simple Network Time Protocol (SNTP) enables the notion of time (date, month, time) into the Mediatrix 4102. It updates the internal clock of the unit, which is the client of a SNTP server. It is required when dealing with features such as the caller ID.

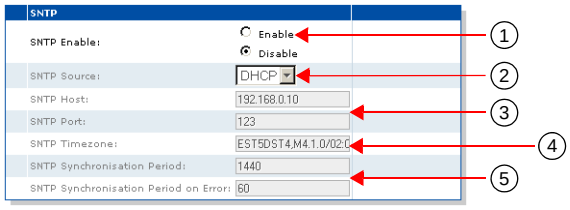
SNTP is used to synchronize a SNTP client with a SNTP or NTP server by using UDP as transport.

You can also set these parameters via SNMP, as described in "Chapter 21 - SNTP Settings" on page 317.

▶ **To set the SNTP client of the Mediatrix 4102:**

1. In the *SNTP* section of the *Network Settings* page, select **Enable** in the *SNTP Enable* choices.

**Figure 26:** SNTP Section




2. Select the configuration source of the SNTP information in the *SNTP Source* choices.

**Table 39:** Network Settings Configuration Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

3. If the SNTP source is **Static**, enter the following static IP information.

**Table 40:** SNTP Static Address

| Field | Description |
|---|---|
| SNTP Host | Static SNTP server IP address or domain name. |
| SNTP Port | Static SNTP server IP port number. |

4. Enter a valid string in the *SNTP Timezone* field.

   The format of the string is validated upon entry. Invalid entries are refused. The default value is:

   `EST5DST4,M4.1.0/02:00:00,M10.5.0/02:00:00`

   A POSIX string is a set of standard operating system interfaces based on the UNIX operating system. The format of the IEEE 1003.1 POSIX string is defined in the <bootp-dhcp-option-88.txt> Internet draft as:

   `STDOFFSET[DST[OFFSET],[START[/TIME],END[/TIME]]]`

   Refer to the following sub-sections for explanations on each part of the string.

5. Set the synchronization information:

**Table 41:** SNTP Synchronization Information

| Field | Description |
|---|---|
| SNTP Synchronisation Period | Time interval (in minutes) between requests made to the SNTP server. The result is used to synchronize the unit with the time server. |
| SNTP Synchronisation Period on Error | Time interval (in minutes) between retries after an unsuccessful attempt to reach the SNTP server. |

6. Click *Submit* if you do not need to set other parameters.

   The current configuration file server information is displayed in the *SNTP Info* section.

## STD / DST

Three or more characters for the standard (STD) or alternative daylight saving time (DST) time zone. Only STD is mandatory. If DST is not supplied, the daylight saving time does not apply. Lower and upper case letters are allowed. All characters are allowed except digits, leading colon (:), comma (,), minus (-), plus (+), and ASCII NUL.

## OFFSET

Difference between the GMT time and the local time. The offset has the format *h[h][:m[m][:s[s]]]*. If no offset is supplied for DST, the alternative time is assumed to be one hour ahead of standard time. One or more digits can be used; the value is always interpreted as a decimal number.

The hour value must be between 0 and 24. The minutes and seconds values, if present, must be between 0 and 59. If preceded by a minus sign (-), the time zone is east of the prime meridian, otherwise it is west, which can be indicated by the preceding plus sign (+). For example, New York time is GMT 5.

## START / END

Indicates when to change to and return from the daylight saving time. The *START* argument is the date when the change from the standard to the daylight save time occurs; *END* is the date for changing back. If *START* and *END* are not specified, the default is the US Daylight saving time start and end dates. The format for start and end must be **one** of the following:

- **n** where *n* is the number of days since the start of the year from 0 to 365. It must contain the leap year day if the current year is a leap year. With this format, you are responsible to determine all the leap year details.

- **Jn** where *n* is the Julian day number of the year from 1 to 365. Leap days are not counted. That is, in all years – including leap years – February 28 is day 59 and March 1 is day 60. It is impossible to refer to the occasional February 29 explicitly. The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is *02:00:00*.
- **Mx[x].y.z** where *x* is the month, *y* is a week count (in which the z day exists) and *z* is the day of the week starting at 0 (Sunday). As an example:

  `M10.4.0`

  is the fourth Sunday of October. It does not matter if the Sunday is in the 4th or 5th week.

  `M10.5.0`

  is the last Sunday of October (5 indicates the last z day). It does not matter if the Sunday is in the 4th or 5th week.

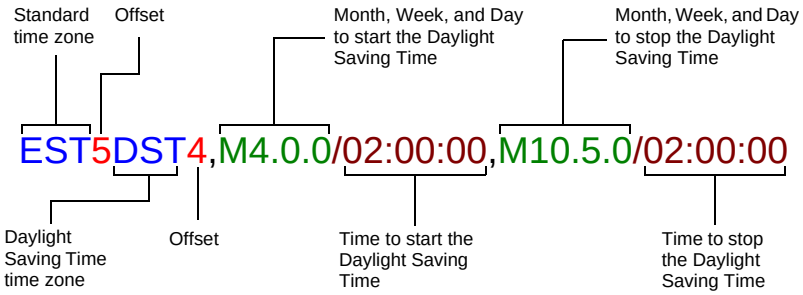
  `M10.1.6`

  is the first week with a Saturday (thus the first Saturday). It does not matter if the Saturday is in the first or second week.

  The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If TIME is not specified, the default is *02:00:00*.

## Example

The following is an example of a proper POSIX string:




The following are some valid POSIX strings:

**Table 42:** Valid POSIX Strings

| Time Zone | POSIX String |
|---|---|
| Pacific Time (Canada & US) | PST8PDT7,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Mountain Time (Canada & US) | MST7MDT6,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Central Time (Canada & US) | CST6CDT5,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Eastern Time Canada & US) | EST5EDT4,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Atlantic Time (Canada) | AST4ADT3,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| GMT Standard Time | GMT0DMT-1,M3.5.0/01:00:00,M10.5.0/02:00:00 |
| W. Europe Standard Time | WEST-1DWEST-2,M3.5.0/02:00:00,M10.5.0/03:00:00 |
| China Standard Time | CST-8 |
| Tokyo Standard Time | TST-9 |
| Central Australia Standard Time | CAUST-9:30DCAUST-10:30,M10.5.0/02:00:00,M3.5.0/02:00:00 |
| Australia Eastern Standard Time | AUSEST-10AUSDST-11,M10.5.0/02:00:00,M3.5.0/02:00:00 |
| UTC (Coordinated Universal Time) | UTC0 |

# Configuration File Download

The configuration file download feature allows to update the Mediatrix 4102 configuration by transferring a configuration file via TFTP or HTTP. The configuration file is transferred from the configuration file download server and the Mediatrix 4102 is the session initiator. The advantage of having the Mediatrix 4102 as the session initiator is to allow NAT traversal.

The *Configuration File* sub-page of the *Management* page allows you to set various configuration file download parameters grouped in three categories:

- General parameters
- Encryption
- Automatic Update

You can also set these parameters via SNMP, as described in "Chapter 14 - Configuration File Download" on page 219.

## Configuration File Download Server

The service allows to download a unique file for each Mediatrix 4102, and/or a file shared among many units. These configuration files may be encrypted or not.

You have the choice to perform the configuration file download by using the TFTP protocol or the HTTP protocol. You can also configure the Mediatrix 4102 to automatically update its configuration.

To download a configuration file, you may need to setup the following applications on your computer:

- TFTP server with proper root path
- SNTP server properly configured
- HTTP server with proper root path

### Configuring the TFTP Server

If you are to perform a configuration file download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the TFTP server host. It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

### Configuring the SNTP Server

If you are to use the automatic configuration file update feature (see "Automatic Configuration Update" on page 60 for more details), you need to have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "SNTP Settings" on page 51 for more details on how to configure the Mediatrix 4102 for a SNTP server.

### Configuring the HTTP Server

If you are to perform a configuration file download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

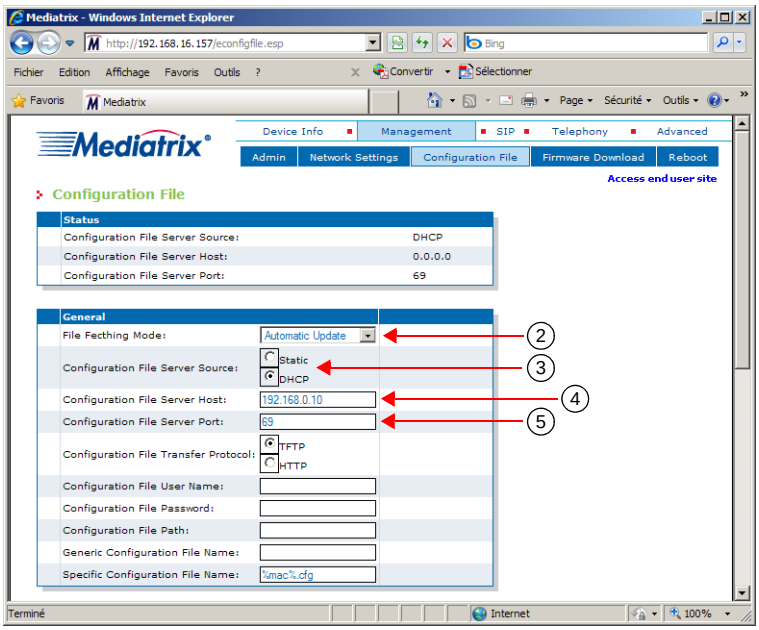
## Configuration File Server Settings

The Mediatrix 4102 must know the IP address and port number of its configuration file server. This server contains the configuration file the Mediatrix 4102 will download. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself.

### ▶ To set the configuration file server parameters:

1. In the web interface, click the *Management* link, then the *Configuration File* sub-link.

**Figure 27:** Management – Configuration File Web Page




2. Select how configuration files are downloaded in the *File Fetching Mode* drop-down menu.

**Table 43:** File Fetching Mode Parameters

| Parameter | Description |
|---|---|
| Disabled | Does not perform a configuration file download. |
| Use Management Server | Configuration files are sent by the management server upon request. The management server is the initiator of the TFTP session. |
| Automatic Update | Configuration files are automatically fetched by the unit. Refer to "Automatic Configuration Update" on page 60 for details. The unit is the initiator of the transfer sessions. This method facilitates the NAT traversal. |

3. Select the configuration source of the configuration download in the *Configuration File Server Source* choices.

**Table 44:** Configuration File Information Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |

**Table 44:** Configuration File Information Sources (Continued)

| Source | Description |
|---|---|
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

4. If the configuration file server configuration source is **Static**, enter the configuration file server static IP address or domain name in the *Configuration File Server Host* field.

   This is the current address of the PC that hosts the configuration files.

5. If the configuration file server configuration source is **Static**, enter the configuration file server static IP port number in the *Configuration File Server Port* field.

   The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP server to perform the configuration file download, you must change the port value to 80.

6. Proceed to "Setting up the Configuration File Download" on page 56.

## Setting up the Configuration File Download

When performing a configuration file download, you can download two different files:

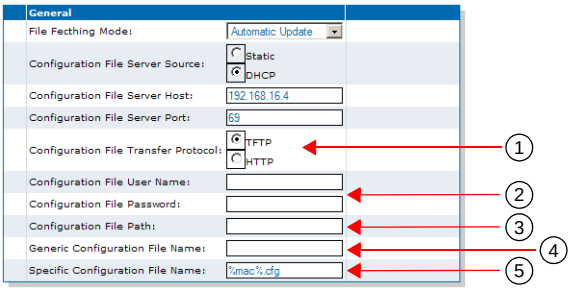
- A generic configuration file that should be used to update a large number of units with the same configuration.
- A specific configuration file that contains the configuration for a single unit, for instance the telephone numbers of its lines.

When both the generic and specific configuration files are downloaded, settings from the specific configuration file always override the settings from the generic configuration file. These files must be located in the same directory.

### ▶ To setup the configuration file download:

1. In the *General* section of the *Configuration File* page, set the transfer protocol to use in the *Configuration File Transfer Protocol* field.

**Figure 28:** General Section




You have the choice between **TFTP** and **HTTP**.

Your HTTP server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.

2. If your HTTP server requires authentication to download the configuration file, set the following:
   - The user name in the *Configuration File User Name* field.
   - The password in the *Configuration File Password* field.

   The Mediatrix 4102 supports basic and digest HTTP authentication, as described in RFC 2617.

**3.** Set the path, on the remote server, of the directory where the configuration files are located in the *Configuration File Path* field.

The path is case sensitive hence it must be entered properly.

The path is relative to the root path of the transfer server (*configFileFetchingHost*). Use the "/" character when defining the path to indicate sub-directories.

Let's consider the following example:

- The directory that contains the configuration file is called: **Config_File**.
- This directory is under **C:/Root/Download**.

**Table 45:** Path Configurations Example

| Root Path | Corresponding Path Name |
|---|---|
| c:/root/download | Config_File |
| c:/ | root/download/Config_File |
| c:/root | download/Config_File |

The following are some tips to help your download process:

- Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
- If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
- Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
- Cut and paste the path and/or name of the directory that contains the extracted files into the configuration file path of the Mediatrix 4102 (you may have to convert "\" into "/") to eliminate typographical errors.

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

**4.** Set the name of the generic configuration file to download in the *Generic Configuration File Name* field.

The file name is case sensitive hence it must be entered properly.

This file should be used to update a large number of units with the same configuration.

If you leave the field empty, the Mediatrix 4102 does not download the generic configuration file.

**5.** Set the name of the specific configuration file to download in the *Specific Configuration File Name* field.

The file name is case sensitive hence it must be entered properly.

This file should be used to update the configuration of a single unit.

This field may contain macros that are substituted by actual values when downloading the configuration file. Supported macros are:

- %mac%: the MAC address of the unit
- %product%: the product name of the unit
- %%: the character "%"

For instance:

- The "%mac%.xml" value for a Mediatrix 4102 with MAC address "0090F12345AB" will be "0090F12345AB.xml".
- The value "Hello%%Hi" will result in "Hello%Hi".
- The value "%%%mac%%%mac%.xml" will result in "%0090F12345AB%mac%.xml".

  From left to right: the first macro encountered is first substituted, the second macro encountered is then substituted, etc.

When the character “%” is not part of a macro, it is not replaced. The following are examples:

- The value “%mac.xml” stays “%mac.xml”
- The value “Hello%Hi” stays “Hello%Hi”
- The value “%moc%.xml” stays “%moc%.xml”

If the field is empty (after macro substitution), the Mediatrix 4102 does not download the specific configuration file.

**6.** Click *Submit* if you do not need to set other parameters.

## Configuration Update Status

If valid configuration files are successfully downloaded, then the Mediatrix 4102 automatically restarts to apply all the new settings. If the Mediatrix 4102 does not restart, this could mean the download failed or that the configuration in the file is the same as the configuration in the unit.

A lot of information is transmitted as system log (syslog) messages. The following are some of the syslog messages sent by the unit:

**Table 46:** Configuration File Download Syslog Messages

| Level | Message | Event |
|---|---|---|
| Informational | `The specific configuration update succeeded.` | The configuration update with the specific configuration file has been successful. |
| Error | `The specific configuration update failed.` | The configuration update with the specific configuration file experienced an error and has not been completed. |
| Informational | `The configuration file “XXX” was successfully fetched.` | A configuration file was successfully fetched. |
| Informational | `The unit configuration is not updated. The parameter values defined in the fetched configuration files are identical to the actual unit configuration.` | The parameter values defined in the fetched configuration files are identical to the actual unit configuration. |
| Informational | `The generic file \”%s\” parameter values are not applied. They are either identical to the unit configuration or overwritten by the specific file.` | The generic configuration file parameter values are either identical to the unit configuration or overwritten by the specific configuration file. |
| Warning | `None of the parameter values defined in the configuration file \”%s\” was successfully applied.` | No parameter value from a fetched configuration file was successfully applied (e.g., because of bad OIDs). |
| Informational | `Parameter values defined in the configuration file \”%s\” were successfully applied.` | A fetched configuration file was successfully applied. |
| Informational | `The unit is restarting to complete the configuration update.` | All necessary fetched configuration files were successfully applied. |

## Configuration Files Encryption

You can secure the exchange of configuration files between the server and the Mediatrix 4102. A privacy key allows the unit to decrypt a previously encrypted configuration file.

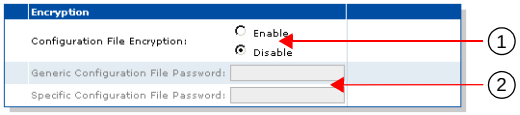
To encrypt a configuration file (generic or specific), you must use the MxCryptFile application. MxCryptFile is a command line tool that encrypts files before sending them to the Mediatrix 4102 unit. Contact your sales representative for more details.

The following describes how to decrypt a previously encrypted generic or specific configuration file. You must have one key for the generic configuration file and another key for the specific configuration file.

### ▶ To decrypt a configuration file:

1. In the *Encryption* section of the *Configuration File* page, select **Enable** in the *Configuration File Encryption* field.

**Figure 29:** Management – Encryption section




The Mediatrix 4102 will be able to decrypt the next encrypted generic or specific configuration file. If you select **Disable**, the configuration file is not decrypted by the unit and the configuration update fails.

2. Set the proper decryption password field with the password used to decrypt the configuration file.

**Table 47:** Decryption Passwords

| Configuration File | Field |
|---|---|
| Generic | Generic Configuration File Password |
| Specific | Specific Configuration File Password |

The password is encoded in hexadecimal notation. You can thus use characters in the range 0-9, A-F. All other characters are not supported.

Each character encodes 4 bits of the key. For instance, a 32-bit key requires 8 characters.

- If you enter too many bits, the key is truncated to the first 448 bits.
- If you do not enter enough bits, the key is padded with zeros.

For instance, a 32-bit key could look like the following: A36CB299.

This key must match the key used for the encryption of the relevant configuration file.

If the field is empty, the configuration file is not decrypted.

3. Click *Submit* if you do not need to set other parameters.

## Configuration Download Procedure

The following steps explain how to download configuration files from the web interface.

**Note:** The configuration download via TFTP can only traverse NATs of types “Full Cone” or “Restricted Cone”. If the NAT you are using is of type “Port Restricted Cone” or “Symmetric”, the file transfer will not work.

▶ **To download configuration files:**

1. Set the configuration file server host and port as defined in “Configuration File Server Settings” on page 55.

**Caution:** When downloading via HTTP, the configuration file server’s port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See “Configuration File Server Settings” on page 55 for more details.

2. Place the configuration files to download on the computer hosting the TFTP or HTTP server.

   These files must be in a directory under the TFTP root path.

3. If not already done, set the transport protocol and configuration file path as described in “Setting up the Configuration File Download” on page 56.

4. Initiate the configuration file download by setting the *System Command* drop-down menu of the System – Admin Web Page to **updateConfiguration**.

   The Mediatrix 4102 immediately downloads the configuration files. See “System Management” on page 47 for more details on the system commands.

5. Click *Submit*.

## Automatic Configuration Update

You can configure the Mediatrix 4102 to automatically update its configuration. This update can be done:

- Every time the Mediatrix 4102 restarts.
- At a specific time interval you can define.

### NAT Variations

NAT treatment of UDP varies among implementations. The four treatments are:

- Full Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host by sending a packet to the mapped external address.
- Restricted Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.
- Port Restricted Cone: Similar to a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.
- Symmetric: All requests from the same internal IP address and port, to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

For more details on NAT treatments, refer to RFC 3489.

## Automatic Update on Restart

The Mediatrix 4102 may download new configuration files each time it restarts.

▶ **To set the automatic update every time the Mediatrix 4102 restarts:**

1. Set the configuration file server host and port as defined in "Configuration File Server Settings" on page 55.

> **Caution:** When downloading via HTTP, the configuration file server's port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.
>
> If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 55 for more details.

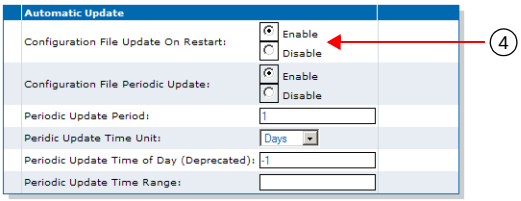
2. Place the configuration files to download on the computer hosting the HTTP or TFTP server.

   These files must be in a directory under the root path.

3. If not already done, set the transport protocol and configuration file path as described in "Setting up the Configuration File Download" on page 56.

4. In the *Automatic Update* section of the *Configuration File* page, select **Enable** in the *Configuration File Update On Restart* field.

**Figure 30:** Management – Automatic Update section




   The automatic configuration update will be performed each time the Mediatrix 4102 restarts.

   The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

5. Click *Submit* if you do not need to set other parameters.

## Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4102 to download new configuration files at a specific day and/or time.

▶ **To set the automatic update at a specific time interval:**

1. Set the configuration file server host and port as defined in "Configuration File Server Settings" on page 55.

> **Caution:** When downloading via HTTP, the configuration file server's port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.
>
> If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 55 for more details.

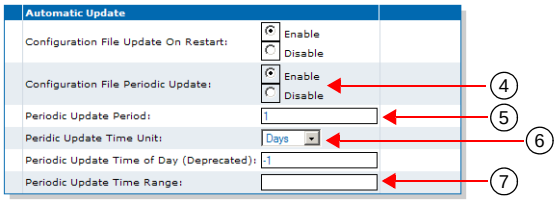
2. Place the configuration files to download on the computer hosting the HTTP or TFTP server.

   These files must be in a directory under the root path.

3. If not already done, set the transport protocol and configuration file path as described in "Setting up the Configuration File Download" on page 56.

4. In the *Automatic Update* section of the *Configuration File* page, select **Enable** in the *Configuration File Periodic Update* field.

**Figure 31:** Management – Automatic Update section




5. Set the waiting period between each configuration update in the *Periodic Update Period* field.

   The time unit for the period is specified in the *Periodic Update Time Unit* field (see Step 6). Available values are from 1 to 48.

6. Define the time base for automatic configuration updates in the *Periodic Update Time Unit* field.

   You have the following choices:

**Table 48:** Time Unit Parameters

| Parameter | Description |
|---|---|
| Minutes | Updates the unit's configuration every *x* minutes. You can specify the *x* value in the *Periodic Update Period* field (see Step 5). |
| Hours | Updates the unit's configuration every *x* hours. You can specify the *x* value in the *Periodic Update Period* field (see Step 5). |
| Days | Updates the unit's configuration every *x* days. You can specify the *x* value in the *Periodic Update Period* field (see Step 5).<br>You can also define the time of day when to perform the update in the *Periodic Update Time Range* field (see Step 7). |

7. If you have selected **days** in Step 6, set the time of the day when to initiate a configuration update in the *Periodic Update Time Range* field.

   The time of the day is based on the *SNTP Timezone* field of the *Management - Network Settings* page (see "SNTP Settings" on page 51 for more details).

   You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "SNTP Settings" on page 51 for more details on how to configure the Mediatrix 4102 for a SNTP server.

   If a time range is specified, the unit will download the configuration files at a random time within the interval specified.

   The format should be one of the following:

```
hh[:mm[:ss]]
hh[:mm[:ss]] - hh[:mm[:ss]]
```

   Where:

```
hh: Hours.
mm: Minutes.
ss: Seconds.
```

   The configuration files are downloaded at the first occurrence of this value and thereafter with a period defined by the *Periodic Update Period* field. Let's say for instance the automatic unit configuration update is set with the time of day at 14h00 and the update period at every 2 days.

   - If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.

- If the time range is set to '14:00 - 15:00' and the automatic unit configuration update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

**Note:** The *Periodic Update Time of Day* field is deprecated. It is recommended to use the *Periodic Update Time Range* field because it has precedence over this field.

**8.** Click *Submit* if you do not need to set other parameters.

## Error Handling

The following configuration file fetching service error sources are divided in three types depending on the transfer protocol: common errors (Table 35), TFTP errors (Table 36) and HTTP errors (Table 37). The error cause and the unit behaviour are also described.

**Table 49:** Configuration File Fetching Error Handling

| Error Type | Cause | Behaviour |
|---|---|---|
| **Common Error Handling** | | |
| Invalid file format | The file format is not valid. | Send a syslog **warning** message including the file location/name with the transfer server address:<br>`The fetched configuration file “XXX”, from server “XXX”, has an invalid format.`<br>No recorded settings applied. |
| Empty file | Committing an empty file. | Send a syslog **warning** message including the file location/name with the transfer server address:<br>`The fetched configuration file “XXX”, from server “XXX”, is empty.` |
| Invalid file content | The file contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126. | Send a syslog **warning** message including the file location/name, the transfer server address and the invalid character (ASCII code):<br>`The fetched configuration file “XXX”, from server “XXX”, has an invalid character “ASCII code XXX”.`<br>No recorded settings applied. |
| Invalid transfer server address | The server address is not valid. | Send a syslog **warning** message including the transfer server address:<br>`No configuration file is fetched because the server host “XXX” is invalid.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File size too big | Downloading a file with a size exceeding 512000 bytes. | Send a syslog **warning** message including the file location/name, the transfer server address, the file size and the maximum allowed size:<br>`The fetched configuration file “XXX”, from server “XXX”, has a size “XXX bytes” that exceeds the maximum allowed size “XXX bytes”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

**Table 49:** Configuration File Fetching Error Handling (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Invalid encryption | The configuration file cannot be decrypted. A badly encrypted file is detected if the header or the padding is invalid. | Send a syslog **warning** message including the file location/name and the transfer server address:<br>`The fetched configuration file \"%s\", from server \"%s\", can not be decrypted.` |
| **TFTP-Specific Error Handling** | | |
| File not found | Received error code 1 (file not found) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file "XXX" was not found on the TFTP server "XXX".`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Access violation | Received error code 2 (access violation) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file "XXX" was not fetched. There was a TFTP access violation with server "XXX".`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Connection timeout | No answer from the TFTP server. The time elapsed since the TFTP request was sent exceeds 32 seconds. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file "XXX" was not fetched. The TFTP connection with server "XXX" timed out.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Transfer error | Received a TFTP error (other than error code 1 and 2) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`Error in the TFTP transfer of the configuration file "XXX" from host "XXX" and port number XXX.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File size too big | Downloading a file with a size exceeding 512000 bytes. | Abort the transfer by sending error code 3 (disk full or allocation exceeded) to the TFTP client. |
| **HTTP-Specific Error Handling** | | |
| Access unauthorized | Received a 401 Unauthorized from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The access to configuration file "XXX" is unauthorized on HTTP server "XXX".`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File not found | Received a 404 Not Found from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file "XXX" was not found on the HTTP server "XXX".`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

**Table 49:** Configuration File Fetching Error Handling (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Session timeout | No answer from the HTTP server. The time elapsed since the HTTP request was sent exceeds 15 seconds. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file “XXX” was not fetched. The HTTP session with server “XXX” timed out.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Session closed by peer | The HTTP server closed the session. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file “XXX” HTTP transfer session was closed by peer: host “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Transfer error | Received an HTTP error (other than 401 and 404) from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address and port:<br>`Error in the HTTP transfer of the configuration file “XXX” from host “XXX” and port number XXX.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

# Firmware Download

This chapter describes how to download from the web interface a firmware version available on the designated firmware server into the Mediatrix 4102.

You have the choice to perform the firmware download by using the TFTP or protocol. You can also configure the Mediatrix 4102 to automatically update its firmware version.

The *Firmware Update* sub-page of the *Management* page allows you to set various firmware download parameters grouped in two categories:

- General parameters
- Automatic Update

You can also set these parameters via SNMP, as described in “Chapter 15 - Software Download” on page 239.

## Before Downloading

To download a firmware, you may need to setup the following applications on your computer:

- TFTP server with proper root path
- MIB browser (with the current Mediatrix 4102 MIB tree)

  You can use the MIB browser built in the Media5’s Unit Manager Network. See “Unit Manager Network – Element Management System” on page xxiv for more details.
- Firmware upgrade zip file
- SNTP server properly configured
- HTTP server with proper root path
- Syslog daemon (optional)

### Configuring the TFTP Server

If you are to perform a firmware download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the firmware file server. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4102 in the same subnet.

It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

### Configuring the SNTP Server

If you are to use the automatic firmware update feature (see "Automatic Configuration Update" on page 60 for more details), you must have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

### Configuring the HTTP Server

If you are to perform a firmware download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4102 in the same subnet.

It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

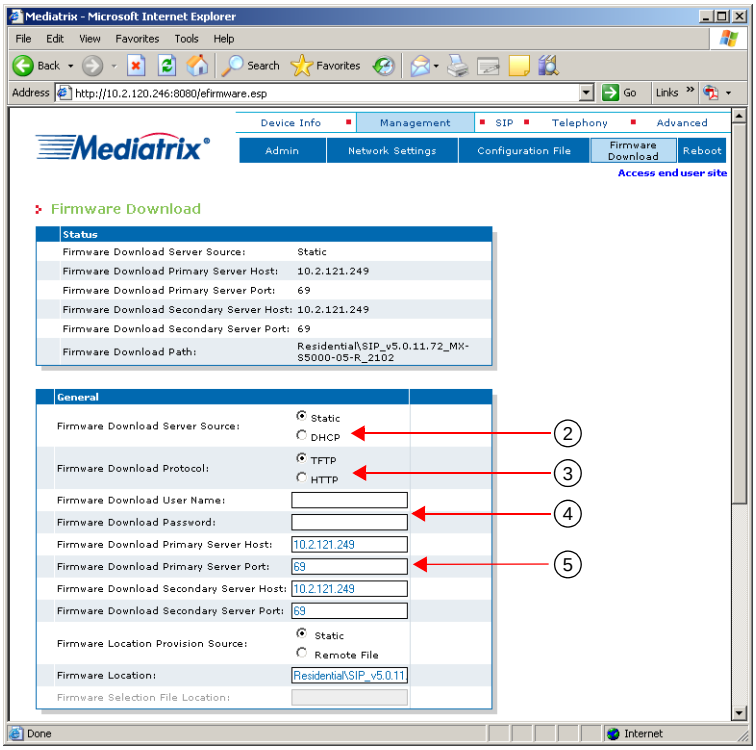
## Firmware Servers Configuration

The Mediatrix 4102 must know the IP address and port number of its Primary and Secondary firmware servers. These servers contain the files required for the firmware update. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself.

▶ **To set the firmware download server parameters:**

1. In the web interface, click the *Management* link, then the *Firmware Download* sub-link.

**Figure 32:** Management – Firmware Download Web Page



2. Select the configuration source of the firmware file information in the *Firmware Download Server Source* choices.

**Table 50:** Configuration File Information Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

3. Set the transfer protocol to use in the *Firmware Download Protocol* field.

   You have the choice between **tftp** and **http**.

   Your HTTP server may activate some caching mechanism for the firmware download. This mechanism caches the initial firmware download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the firmware download and perform it again immediately. The result will still return the original download and not the new one.

4. If your HTTP server requires authentication when downloading the firmware, set the following fields:
   - The user name in the *Firmware Download User Name* field.
   - The password in the *Firmware Download Password* field.

   The Mediatrix 4102 supports basic and digest HTTP authentication, as described in RFC 2617.

5. If the firmware download server configuration source is **Static**:
   - enter the firmware download primary server static IP address or domain name in the *Firmware Download Primary Server Host* field.
   - enter the firmware download primary server static IP port number in the *Firmware Download Primary Server Port* field.
   - enter the firmware download secondary server static IP address or domain name in the *Firmware Download Secondary Server Host* field.
   - enter the firmware download secondary server static IP port number in the *Firmware Download Secondary Server Port* field.

   The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP server to perform the firmware download, you must change the port value to 80.

6. Proceed to "Setting up the Firmware Download" on page 67.

## Setting up the Firmware Download

Configuration settings are not lost when upgrading the firmware to a newer version. However, configuration settings may be lost if you upload an older firmware to the device. See "Firmware Downgrade" on page 75 for more details.

### Extracting the Zip File

The zip file contains the firmware information required for the download.

Extract the contents of the zip file on the PC designated as the firmware download server. Be sure to use the defined folder name. This creates a directory that contains the files required for the Mediatrix 4102 to properly update its firmware.

The directory name must be the same as the name defined in the *Firmware Location* field or *Firmware Selection File Location* field. See "Setting up the Configuration File Download" on page 56 for more details.

Media5 suggests that a folder, named identically to the firmware build, be available and used for the files related to that build only. Each folder should include only one delivery to ensure accuracy.

This directory must be located under the root path as defined in the TFTP/HTTP server or the firmware download will not proceed.

## Setting up the Firmware Download Path

When performing a firmware download, you must configure the path, on the remote firmware download server, of the directory where you extracted the files required for the download. This applies to both the manual or automatic download procedure, using the HTTP or TFTP protocol.

The directory must be located under the root path, as defined in the TFTP or HTTP server, or the firmware download will not proceed. See "Before Downloading" on page 65 for more details.

The Mediatrix 4102 first downloads a file called "setup.inf". This file contains the list of all the other files to download, depending on the product. The "setup.inf" file and all the other files must be in the same directory. If any of the files is missing, the procedure will not work properly.

▶ **To setup the firmware download path:**

1. In the *General* section, select where to get the image location in the *Firmware Location Provision Source* field.

   You have the following choices:

**Table 51:** Image Location Parameters

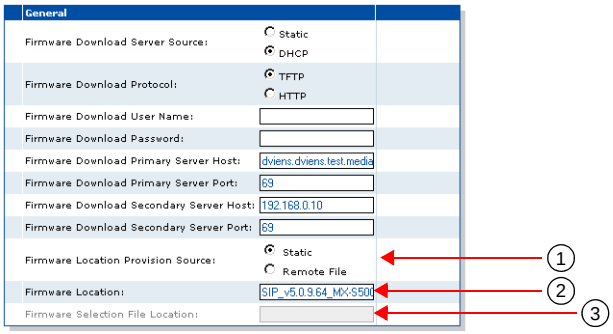
| Parameter | Description |
|---|---|
| static | Uses the directory specified in the *Firmware Location* field (see Step 2). |
| remoteFile | The image location is defined in a file called "mediatrix4102targetimage.inf". The location of this file is defined in the *imageSelectionFileLocation* variable.<br>This is useful if you are using automatic updates with multiple units (see Step 3). |

2. If you set the *Firmware Location Provision Source* parameter to **Static** (see Step 1), configure the firmware download path in the *Firmware Location* field.

   This is the location of the "setup.inf" file that contains the list of the files to download into the Mediatrix 4102. The "setup.inf" file and all the other files must be in the same directory. In other words, this is the path from the root TFTP/HTTP folder down to the files extracted from the zip file.

   Note that the path must contain a maximum of 63 characters.

**Figure 33:** Management – Firmware Download Web Page



3. If you set the *Firmware Location Provision Source* parameter to **remoteFile** (see Step 1):

   *a.* Create a text file and write the path and/or name of the directory that contains the files required for download. Save this file as “mediatrix4102targetimage.inf” under the server root path.

**Note:** If you leave the file empty, the Mediatrix 4102 will look for the firmware download information in the root directory of the image server.

   *b.* Configure the path of the “mediatrix4102targetimage.inf” file in the *Firmware Selection File Location* field.

   Note that the selection file name is in lower case. Some web servers are case sensitive. The path must contain a maximum of 63 characters.

   This is useful if you are using automatic updates with multiple units. If you want the units to download a new version, you only have to change the path once in the “mediatrix4102targetimage.inf” file. If you were to use the *Firmware Location* field, you would have to change the path in every unit.

4. Click *Submit* if you do not need to set other parameters.

**Note:** The current firmware server information is displayed in the *Status* section.

*Example*

Let’s consider the following example:

- The directory that contains the files required for download is called: **SIP_v5.0.1.1_MX-S5001-01**.
- This directory is under **C:/Root/Download**.

**Table 52:** Path Configurations Example

| Root Path | Corresponding Path Name |
|---|---|
| c:/root/download | SIP_v5.0.1.1_MX-S5001-01 |
| c:/ | root/download/SIP_v5.0.1.1_MX-S5001-01 |
| c:/root | download/SIP_v5.0.1.1_MX-S5001-01 |

The following are some tips to help your download process:

- If available, use the *Browse* button (or equivalent) of the TFTP/HTTP server to select the directory, eliminating typographical errors.
- Use the “/” character when defining the path to indicate sub-directories. For instance, *root/download*.

  If you are using the TFTP protocol to download the firmware, note that some TFTP servers on Windows do not recognize the “/” character and produce an error. In this case, use the “\” character.
- Use basic directory names, without spaces or special characters such as “~”, “@”, etc., which may cause problems.
- Cut and paste the path and/or name of the directory that contains the extracted files into the image path of the Mediatrix 4102 (you may have to convert “\” into “/”) to eliminate typographical errors.

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

## Firmware Download Status

You can validate the status of the firmware download in various ways.

### *Syslog Messages*

If you are using a Syslog daemon, you will receive messages that inform you of the firmware update status. The following are the syslog messages the Mediatrix 4102 sends:

**Table 53:** Firmware Update Syslog Messages

| Level | Message | Event |
|---|---|---|
| **General Messages** | | |
| Informational | `The software update succeeded.` | The firmware update has been successful. |
| Error | `The software update failed.` | The firmware update experienced an error and has not been completed. |
| Error | `The software update failed (xxx).` | An error occurs when updating the software, internal error code provided. |
| Warning | `Primary image server not specified, cannot download file: xxx` | This error occurs when an image download is initiated and no domain name or address is specified for the primary image server. |
| Warning | `Secondary image server not specified, cannot download file: xxx.` | When a request involving the primary server fails, the secondary server is tried.<br>This error occurs when there is no address or domain name specified for the secondary image server. |
| Error | `Cannot resolve address of image server: xxx.` | A DNS request failed to resolve the domain name of the image server (primary or secondary). |
| Error | `Target image at location: xxx from host: xxx is invalid or corrupted.` | For periodic and automatic updates, the target image to download is first compared with the installed image. This error occurs when this comparison failed because of corruption in the target image files. |
| Informational | `Image download transfer initiated.` | When manual, periodic or “at restart” image download is initiated. |
| Warning | `The file: xxx from host: xxx exceeds the size limit.` | The selection file or “setup.inf” file received exceeds 10000 bytes. |
| Informational | `Target image at location: xxx from host xxx is identical to currently installed image. Transfer aborted.` | For periodic and automatic updates, the target image to download is first compared with the installed image. This message occurs when this comparison determined that the target image is identical to the installed image. |
| Error | `Image does not support hardware (error %d)` | The software download failed because the software image is not compatible with the hardware. |
| **HTTP-Specific Messages** | | |
| Warning | `HTTP image transfer of file: xxx from host: xxx was closed by peer.` | The HTTP transfer was closed by the peer. |

**Table 53:** Firmware Update Syslog Messages (Continued)

| Level | Message | Event |
|---|---|---|
| Warning | `HTTP image transfer of file: xxx from host: xxx was closed due to unsupported or malformed response from the host.` | In the HTTP response, one of the following error occurred:<br>• The protocol version is not 1.0 or 1.1.<br>• Some field or line is not properly formatted.<br>• The trailing <crlf> is not present at the end of the header.<br>• Unsupported kind of response. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because of a malformed or incompatible request.` | When receiving HTTP response #400 or #403. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because of a server error.` | When receiving HTTP response #500 or #501. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because service is unavailable.` | When receiving HTTP response #503. |
| **TFTP-Specific Messages** | | |
| Warning | `Image transfer of file: xxx from host: xxx and port: xxx was closed due to unexpected error` | Unexpected error, either internal or on a TFTP or HTTP connection. |
| Warning | `Image transfer of file: xxx from host: xxx port: xxx was closed after timeout` | When not receiving TFTP packets for 32 seconds or not receiving a HTTP packet for 15 seconds. |
| Warning | `Image transfer. File: xxx not found on host: xxx` | When receiving TFTP error “NOT FOUND” or HTTP response #404. |
| Warning | `Image transfer. Access to file: xxx on host: xxx is unauthorized` | When receiving TFTP error “ACCESS” or HTTP response #401. |

If the local syslog messages are enabled (see “Local Syslog” on page 374 for more details), you can view these messages on the End User web interface.

### *LED States*

When the Mediatrix 4102 initiates a firmware download, the LEDs located on the front panel indicate the status of the process.

**Table 54:** LED States in Firmware Download

| Event | LED State |
|---|---|
| Image downloading and writing | Each LED blinks alternately at 1 Hz with 1/4 ON duty cycle.<br>**Warning**: Do not turn the Mediatrix 4102 off while in this state. |
| Image download failed | All LEDs blink at the same time at 2 Hz with 50% ON duty cycle for 4 seconds. |

See “LED Indicators” on page 16 for a detailed description of the LED patterns related to the firmware download process.

## Firmware Download Procedure

The following steps explain how to download a firmware from the web interface.

▶ **To download a firmware version:**

1. If not already done, setup the Image server used to download the firmware (see "Before Downloading" on page 65).
2. Place the firmware to download on the computer hosting the TFTP or HTTP server.
   The file must be in a directory under the TFTP root path.
3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.
4. If not already done, configure the Image path as described in "Setting up the Configuration File Download" on page 56.
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in "Firmware Servers Configuration" on page 66.

**Caution:** When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 55 for more details.

6. Initiate the firmware download by setting the *System Command* drop-down menu of the *Management – Admin* Web Page to **downloadSoftware**.
7. Click *Submit*.
   This starts the download process. See "System Management" on page 47 for more details on the system commands.

**Caution:** Never shutdown the Mediatrix 4102 manually while in the download process, because the image may be partially written and the Mediatrix 4102 is unable to restart.

The firmware download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

If Transparent Address Sharing is enabled during the firmware download, the PC connected to the Mediatrix 4102 may experience momentary loss of Internet connectivity.

## Automatic Firmware Update

You can configure the Mediatrix 4102 to automatically update its firmware. This update can be done:

- Every time the Mediatrix 4102 restarts.
- At a specific time interval you can define.

### Automatic Update on Restart

The Mediatrix 4102 may download a new firmware each time it restarts.

▶ **To set the automatic update every time the Mediatrix 4102 restarts:**

1. If not already done, setup the Image server used to download the firmware (see "Before Downloading" on page 65).
2. Place the firmware to download on the computer hosting the TFTP or HTTP server.
   The file must be in a directory under the TFTP root path.
3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.

4. If not already done, configure the Image path as described in "Setting up the Configuration File Download" on page 56.

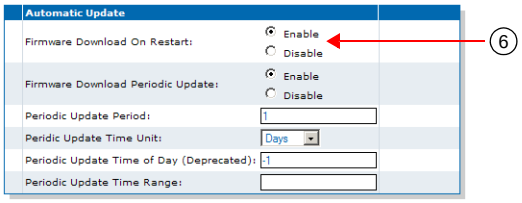
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in "Firmware Servers Configuration" on page 66.

**Caution:** When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 55 for more details.

6. In the *Automatic Update* section of the *Firmware Download* page, select **Enable** in the *Firmware Download On Restart* field.

**Figure 34:** Management – Automatic Update section




The automatic firmware download will be performed each time the Mediatrix 4102 restarts.

7. Click *Submit* if you do not need to set other parameters.

## Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4102 to download a new firmware at a specific day and/or time.

▶ **To set the automatic update at a specific time interval:**

1. If not already done, setup the Image server used to download the firmware (see "Before Downloading" on page 65).

2. Place the firmware to download on the computer hosting the TFTP or HTTP server.

   The file must be in a directory under the TFTP root path.

3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.

4. If not already done, configure the Image path as described in "Setting up the Configuration File Download" on page 56.

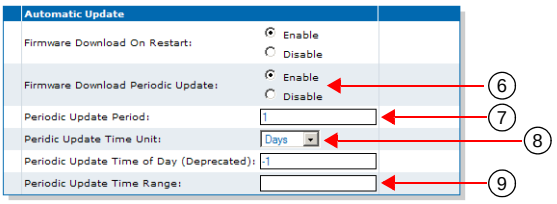
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in "Firmware Servers Configuration" on page 66.

**Caution:** When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 55 for more details.

6. In the *Automatic Update* section of the *Firmware Download* page, select **Enable** in the *Firmware Download Periodic Update* field.

**Figure 35:** Management – Automatic Update section




7. Set the waiting period between each firmware update in the *Periodic Update Period* field.

   The time unit for the period is specified in the *Periodic Update Time Unit* field (see Step 6). Available values are from 1 to 48.

8. Define the time base for automatic firmware updates in the *Periodic Update Time Unit* field.

   You have the following choices:

**Table 55:** Time Unit Parameters

| Parameter | Description |
|---|---|
| Minutes | Updates the unit's firmware every *x* minutes. You can specify the *x* value in the *Periodic Update Period* field (see Step 5). |
| Hours | Updates the unit's firmware every *x* hours. You can specify the *x* value in the *Periodic Update Period* field (see Step 5). |
| Days | Updates the unit's firmware every *x* days. You can specify the *x* value in the *Periodic Update Period* field (see Step 5).<br>You can also define the time of day when to perform the update in the *Periodic Update Time Range* field (see Step 7). |

9. If you have selected **Days** in Step 6, set the time of the day when to initiate a firmware update in the *Periodic Update Time Range* field.

   The time of the day is based on the *SNTP Timezone* field of the *Management - Network Settings* page (see "SNTP Settings" on page 51 for more details).

   You must have a time server SNTP that is accessible and properly configured, or the automatic firmware update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "SNTP Settings" on page 51 for more details on how to configure the Mediatrix 4102 for a SNTP server.

   If a time range is specified, the unit will initiate the image software download at a random time within the interval specified.

   The format should be one of the following:

   ```
   hh[:mm[:ss]]
   hh[:mm[:ss]] - hh[:mm[:ss]]
   ```

   Where:

   ```
   hh: Hours.
   mm: Minutes.
   ss: Seconds.
   ```

   The image software download is initiated at the first occurrence of this value and thereafter with a period defined by the *Periodic Update Period* field. Let's say for instance the automatic update is set with the time of day at 14h00 and the update period at every 2 days.

   - If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.

- If the time range is set to '14:00 - 15:00' and the automatic update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

**Note:** The *Periodic Update Time of Day* field is deprecated. It is recommended to use the *Periodic Update Time Range* field because it has precedence over this field.

10. Click *Submit* if you do not need to set other parameters.

## Spanning Tree Protocol (STP)

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. When a firmware download occurs, the *LAN* and *WAN* connectors of the Mediatrix 4102 may switch off. This shutdown may trigger these network switches to shutdown the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent firmware download.

To prevent this, the Mediatrix 4102 supports the STP. However, this management has a potential time cost. It may appear from time to time that firmware downloads take more time. This is normal.

The following is an example where the STP management impacts the download duration.

- The firmware download procedure does not use any DHCP and DNS services.
- The primary image server is down (or not properly configured).
- The secondary image server is up and running well.

In this case, the Mediatrix 4102 tries to contact the primary image server. As it is not available, the Mediatrix 4102 retries for two minutes. It contacts the secondary server after that period and starts the firmware download.

**Note:** When using the Mediatrix 4102, Media5 recommends to disable the Spanning Tree Protocol on the network to which the unit is connected.

## Firmware Downgrade

It is possible to downgrade a Mediatrix 4102 from the current version (for instance, v5.0rx.x) to an older version (for instance, v4.4rx.x).

**Note:** If you perform a default reset on the Mediatrix 4102, you must download the current version into the unit before performing the firmware downgrade procedure.

**To perform a firmware downgrade:**

1. Create, in a common folder under the TFTP root path, the current (for instance, v5.0) and older (for instance, v4.4) applications folders.
2. Re-update the Mediatrix 4102 with the current application.

   The Mediatrix 4102 runs the current firmware version (v5.0rx.x).
3. Perform the firmware downgrade to the older application (v4.4rx.x) as described in "Firmware Download Procedure" on page 72.

## Emergency Firmware Procedure

If the firmware download is suddenly interrupted, it may not be complete. Without any protection against this situation, the Mediatrix 4102 is not functional.

A transfer may be interrupted for the following reasons:

- An electrical shortage.
- The user of the Mediatrix 4102 can accidentally power off the unit.

Depending on the moment when the firmware download has been interrupted, the emergency firmware procedure (also called rescue application) can automatically start a new firmware download to repair the firmware if it has been corrupted by the interruption. However, there is a small but critical time frame during which unrecoverable errors could happen. This is why it is very important that the unit is not turned off during firmware downloads.

### Using the Emergency Firmware

When the emergency firmware procedure starts, the following steps apply:

1. The Mediatrix 4102 tries to initiate the firmware download with the primary firmware server.
2. If the firmware download fails with the primary firmware server, the Mediatrix 4102 tries to initiate the firmware download with the secondary firmware server.
3. If the primary and the secondary servers cannot be reached, the Mediatrix 4102 tries two default servers: 192.168.0.10 and then 192.168.0.2.

   If, for some reason, it is impossible to rescue the unit by using the primary and secondary servers, setting up a server at one of these addresses within the correct subnet will provide an ultimate way to rescue the unit. However, if these addresses cannot be reached from the unit's subnet, the default gateway must provide appropriate routing to them.
4. If the firmware download also fails with the two default servers, the Mediatrix 4102 idles for one minute.
5. After this one minute, the Mediatrix 4102 tries to initiate the firmware download again.
6. If the firmware download fails again with the primary, secondary, and default firmware servers, the Mediatrix 4102 idles for two minutes before attempting to initiate the firmware download.
7. If the emergency firmware download still fails, the Mediatrix 4102 tries to initiate the firmware download again by doubling the delay between each attempt up to a maximum of 16 minutes:
   - first attempt: 1 minute delay
   - second attempt: 2 minutes delay
   - third attempt: 4 minutes delay
   - fourth attempt: 8 minutes delay
   - fifth attempt: 16 minutes delay
   - sixth attempt: 16 minutes delay
   - etc.

   This procedure continues until the firmware download completes successfully. The firmware download can fail if the firmware server cannot be reached or if the firmware directory is not found on the firmware server.

CHAPTER 5

# Web Interface – SIP Parameters

The *SIP* page allows you to configure the various SIP-related parameters of the Mediatrix 4102:

- General SIP configuration parameters
- SIP Interop parameters
- Authentication parameters

## SIP Servers Configuration

| | |
|---|---|
| **Standards Supported** | • RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication<br>• RFC 3863 – Presence Information Data Format (PIDF) |

The *Configuration* sub-page of the *SIP* page allows you to configure the SIP server and SIP user agent parameters of the Mediatrix 4102.

### SIP Servers

The Mediatrix 4102 uses the following types of servers:

- Registrar server
- Proxy server
- Outbound Proxy server
- Presence Compositor server

#### Registrar Server

The registrar server accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.

#### Proxy Server

The proxy server is an intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is passed on to another entity that can further process the request. Proxies are also useful for enforcing policy and for firewall traversal. A proxy interprets, and, if necessary, rewrites parts of a request message before forwarding it.

#### Outbound Proxy Server

An outbound proxy is an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. The outbound proxy receives all outbound traffic and forwards it. Incoming traffic may or may not go through the outbound proxy. The outbound proxy's address is never used in the SIP packets, it is only used as a physical network destination for the packets.

When the outbound proxy is enabled, the proxy is still used to create the *To* and *From* headers, but the packets are physically sent to the outbound proxy.

The outbound proxy is enabled if the IP address is valid (i.e., not 0.0.0.0).

### Presence Compositor Server

A User Agent Server (UAS) that processes PUBLISH requests and is responsible for compositing event state into a complete, composite event state of a resource for a presentity.

## SIP Configuration

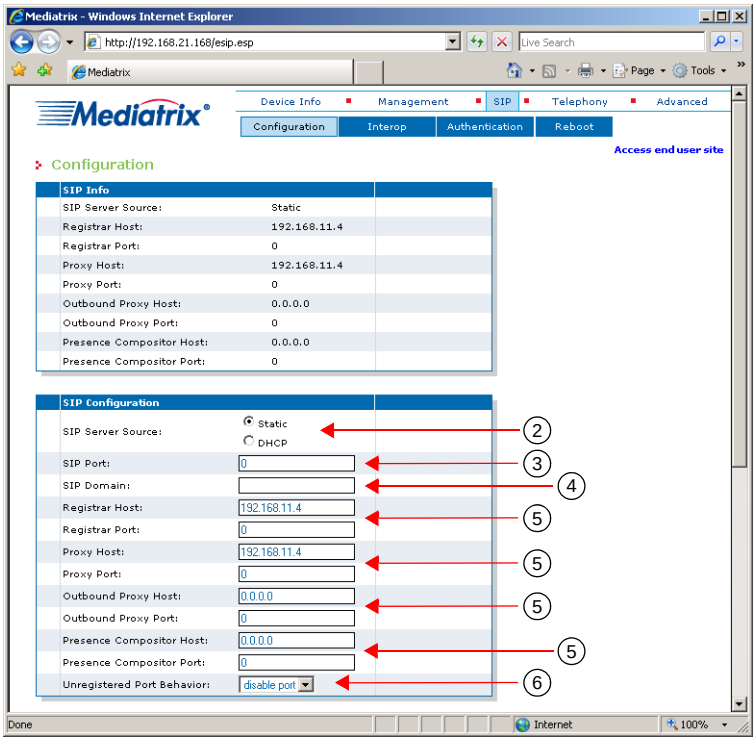
The Mediatrix 4102 must know the IP address and port number of the SIP servers. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself.

You can also set these parameters via SNMP, as described in "Chapter 10 - SIP Servers" on page 177 and "Chapter 19 - SIP Protocol Features" on page 287.

**To set the SIP servers configuration:**

1. In the web interface, click the *SIP* link, then the *Configuration* sub-link.

**Figure 36:** SIP – Configuration Web Page




2. Select the configuration source of the SIP servers information in the *SIP Server Source* choices.

**Table 56:** SIP Servers Configuration Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details. |

3. Set the user agent port number in the *SIP Port* field.

   If this field is set to 0, the default SIP port is used.

4. Define whether or not to override the default proxy home domain used by entering a domain in the *SIP Domain* field.

   This value replaces the home domain proxy host as defined in the *Proxy Host* field. It is used by the address of record in the *To* and *From* headers.

5. If the SIP server configuration source is **Static**:
   - enter the SIP registrar server static IP address or domain name in the *Registrar Host* field.
   - enter the SIP registrar server static IP port number in the *Registrar Port* field.
   - enter the SIP Proxy server static IP address or domain name in the *Proxy Host* field.
   - enter the SIP Proxy server static IP port number in the *Proxy Port* field.
   - enter the SIP outbound proxy server static IP address or domain name in the *Outbound Proxy Host* field.

     Setting the address to **0.0.0.0** disables the outbound proxy.
   - enter the SIP outbound proxy server static IP port number in the *Outbound Proxy Port* field.

**Note:** If the port number corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.

   - enter the SIP Presence Compositor server static IP address or domain name in the *Presence Compositor Host* field.
   - enter the SIP Presence Compositor server static IP port number in the *Presence Compositor Port* field.

6. Specify whether a line should remain enabled or not when not registered in the *Unregistered Port Behavior* field.

   This is useful if you want your users to be able to make calls even if the line is not registered with the SIP server. The following values are available:

**Table 57:** Unregistered Line Behaviour

| Value | Description |
|---|---|
| disable port | When the line is not registered, it is disabled. The user cannot make or receive calls. Picking up the handset yields a fast busy tone, and incoming INVITEs receive a “403 Forbidden” response. |
| enable port | When the line is not registered, it is still enabled. The user can receive and initiate outgoing calls. Note that because the line is not registered to a registrar, its public address is not available to the outside world; the line will most likely be unreachable except through direct IP calling. |

7. Click *Submit* if you do not need to set other parameters.

   The current SIP server information is displayed in the *SIP Info* section.

## SIP User Agent

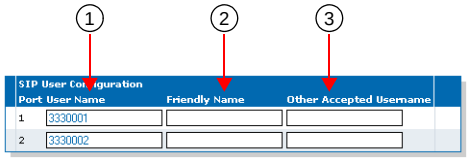
A user agent is a logical entity that can act as both client and server for the duration of a dialog. Each line (also known as endpoint) of the Mediatrix 4102 is a user agent.

You can set information for each user agent such as its telephone number and friendly name. This information is used to dynamically create the *To*, *From* and *Contact* headers used in the request the user agent sends. These headers make up the caller ID information that is displayed on telephones/faxes equipped with a proper LCD display.

▶ **To set user agent information:**

1. In the second *SIP Configuration* section of the *SIP* page, enter a user name for each port in the *User Name* column.

**Figure 37:** SIP – User Agent section




The user name uniquely identifies this endpoint in the domain, such as a telephone number. It is used to create the *Contact* and *From* headers. The *From* header carries the permanent location (IP address, home domain) where the endpoint is located. The *Contact* header carries the current location (IP address) where the endpoint can be reached. Contact headers are used in two ways:

- First, contacts are registered to the registrar. This enables callers to be redirected to the endpoint's current location.
- Second, a contact header is sent along with any request the user agent sends (e.g., INVITE), and is used by the target user agent as a return address for later requests to this endpoint.

You cannot set this field to an empty value. Furthermore, it is reset to 333000X during a factory reset, the X digit being the port number.

2. Enter another name for each line in the *Display Name* column.

   This is a friendly name for the user agent. It contains a descriptive version of the URI and is intended to be displayed to a user interface.

3. Enter a second accepted user name for each line in the *Other Accepted Username* field.

   This is a user name that the endpoint recognizes as its own, but does not register in contacts sent to the registrar. The endpoint only registers the user name set in the *User Name* column.

   You can use this column to add a variation on the user name. For instance, let's say that the user name is a telephone number, 555-1111. A variation could be to prefix the local area or country code, such as 819-555-1111.

   To include more than one user name, separate them with a "," character, such as: user1, user2, 5552222, 18195552222.

4. Click *Submit* if you do not need to set other parameters.

## SIP Registration

You can refresh the registration, i.e., commit the changes you have done to the registration. When refreshing the registration, all enabled endpoints unregister themselves from the previous registrar and send a new registration to the current registrar with the current parameters.

You can also set this parameter via SNMP, as described in "Registration Parameters" on page 296.

▶ **To refresh the registrations:**

1. In the *SIP Registration* section of the *SIP* page, set the registration command in the *SIP Registration Command* menu.

**Figure 38:** SIP – SIP Registration section

The following values are available:

- noOp: No operation.
- refresh: Refresh registrations.

## SIP Publication

You can refresh the publication, i.e., commit the changes you have done to the publication. When refreshing the publications, all enabled endpoints unpublish themselves from the previous Presence Compositor and send a new publication to the current Presence Compositor with the current parameters.

You can also set this parameter via SNMP, as described in "Publication Parameters" on page 297.

▶ **To refresh the publications:**

1. In the *SIP Publication* section of the *SIP* page, set the publication command in the *SIP Publication Command* menu.

**Figure 39:** SIP – SIP Publication section

The following values are available:

- noOp: No operation.
- refresh: Refresh publications.

# SIP Interop

The *Interop* sub-page of the *SIP* page allows you to configure the SIP penalty box, SIP transport parameters, and specific interop parameters of the Mediatrix 4102.

## SIP Penalty Box

The penalty box feature is used to "quarantine" a given host which address times out. During that time, the address is considered as "non-responding" for all requests.

This feature is most useful when using multiple servers and some of them are down. It ensures that users wait a minimal period of time before trying a secondary host.

You can also set these parameters via SNMP, as described in "SIP Penalty Box" on page 295.

### Penalty Box vs Transport Types

Media5 recommends to use this feature with care when supporting multiple transports (see "SIP Transport Type" on page 83 for more details) or you may experience unwanted behaviours.

When the Mediatrix 4102 must send a packet, it retrieves the destination from the packet. If the destination address does not specify a transport to use and does not have a DNS SRV entry that configures which transport to use, then the Mediatrix 4102 tries all transports it supports, starting with UDP. If this fails, it tries with TCP. The unit begins with UDP because all SIP implementations must support this transport, while the mandatory support of TCP was only introduced in RFC 3261.

**Note:** An important fact is that it is not the destination itself that is placed in the penalty box, but the combination of address, port and transport. When a host is in the penalty box, it is never used to try to connect to a remote host unless it is the last choice for the Mediatrix 4102 and there are no more options to try after this host.

Let's say for instance that the Mediatrix 4102 supports both the UDP and TCP transports. It tries to reach endpoint "B" for which the destination address does not specify a transport and there is no DNS SRV entry to specify which transports to use in which order. It turns out that this endpoint "B" is also down. In this case, the Mediatrix 4102 first tries to contact endpoint "B" via UDP. After a timeout period, UDP is placed in the penalty box and the unit then tries to contact endpoint "B" via TCP. This fails as well and TCP is also placed in the penalty box.

Now, let's assume endpoint "B" comes back to life and the Mediatrix 4102 tries again to contact it before UDP and TCP are released from the penalty box. First, the unit tries UDP, but it is currently in the penalty box and there is another transport left to try. The Mediatrix 4102 skips over UDP and tries the next target, which is TCP. Again, TCP is still in the penalty box, but this time, it is the last target the Mediatrix 4102 can try, so penalty box or not, TCP is used all the same to try to contact endpoint "B".

There is a problem if endpoint "B" only supports UDP (RFC 2543-based implementation). Endpoint "B" is up, but the Mediatrix 4102 still cannot contact it: with UDP and TCP in the penalty box, the unit only tries to contact endpoint "B" via its last choice, which is TCP.

The same scenario would not have any problem if the penalty box feature was disabled. Another option is to disable TCP in the Mediatrix 4102, which makes UDP the only possible choice for the unit and forces to use UDP even if it is in the penalty box.

You must fully understand the above problem before configuring this feature. Mixing endpoints that do not support the same set of transports with this feature enabled can lead to the above problems, so it is suggested to either properly configure SRV records for the hosts that can be reached or be sure that all hosts on the network support the same transport set before enabling this feature.

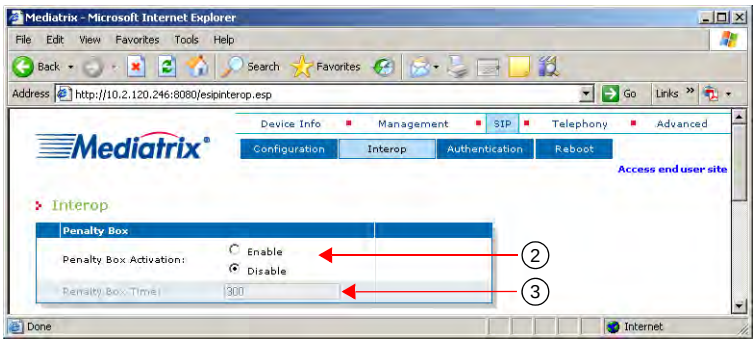
## Penalty Box Configuration

The following steps describe how to configure the penalty box feature.

### ▶ To set the SIP penalty box parameters:

1. In the web interface, click the *SIP* link, then the *Interop* sub-link.

**Figure 40:** SIP – Interop Web Page




2. In the *Penalty Box* section, enable the SIP penalty box feature by selecting **Enable** in the *Penalty Box Activation* choices.

   The penalty box is always "active". This means that even if the feature is disabled, IP addresses are marked as invalid, but they are still tried. This has the advantage that when the feature is enabled, IP addresses that were already marked as invalid are instantly put into the penalty box.

3. Set the amount of time, in seconds, that a host spends in the penalty box in the *Penalty Box Time* field.

   Changing the value does not affect IP addresses that are already in the penalty box. It only affects new entries in the penalty box.

4. Click *Submit* if you do not need to set other parameters.

## SIP Transport Type

| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol |
|---|---|

You can globally set the transport type for all the lines of the Mediatrix 4102 to either UDP (User Datagram Protocol) or TCP (Transmission Control Protocol). The Mediatrix 4102 will include its supported transports in its registrations.

Please note that RFC 3261 states the implementations must be able to handle messages up to the maximum datagram packet size. For UDP, this size is 65,535 bytes, including IP and UDP headers. However, the maximum datagram packet size the Mediatrix 4102 supports for a SIP request or response is 5120 bytes excluding the IP and UDP headers. This should be enough, as a packet is rarely bigger than 2500 bytes.

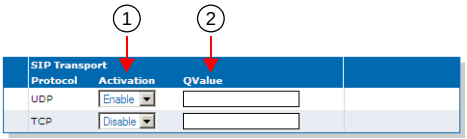
You can also set these parameters via SNMP, as described in "SIP Transport Type" on page 293.

▶ **To set the SIP transport type parameters:**

1. In the *SIP Transport* section of the *Interop* page, enable the transport types to use in the proper *Activation* drop-down menu.

   You can enable or disable the **UDP** and **TCP** transports.

**Figure 41:** SIP Transport Section




2. Set the priority order of the UDP and TCP transports in the proper *Q Value* field.

   A qvalue parameter is added to each contact. The qvalue gives each transport a weight, indicating the degree of preference for that transport. A higher value means higher preference.

   The format of the qvalue string must follow the RFC 3261 ABNF (a floating point value between 0.000 and 1.000). If you specify an empty string, no qvalue is set in the contacts.

3. Click *Submit* if you do not need to set other parameters.

## Interop Parameters

The interop parameters allow the Mediatrix 4102 to properly work, communicate, or connect with specific IP devices.

▶ **To set interop parameters:**

1. In the *Interop* section of the *Interop* page, set the *Escape Pound (#) in SIP URI Username* drop-down menu with the proper behaviour.

   This allows you to define whether or not the pound character (#) must be escaped in the username part of a SIP URI.

**Figure 42:** Interop Section

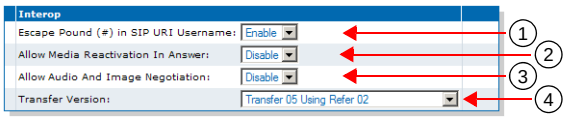



**Table 58:** Escaping Pound Character Parameter

| Parameter | Description |
|---|---|
| Enable | The Pound character (#) is escaped in the username part of a SIP URI. |

**Table 58:** Escaping Pound Character Parameter (Continued)

| Parameter | Description |
|---|---|
| Disable | The Pound character (#) is not escaped in the username part of a SIP URI.<br>Note that RFC 3261 specifies that the pound character (#) needs to be escaped in the username part of a SIP URI. |

2. Set the *Allow Media Reactivation in Answer* drop-down menu with the behaviour of the Mediatrix 4102 when receiving a SDP answer activating a media that had been previously deactivated in the offer

**Table 59:** Media Reactivation Parameters

| Parameter | Description |
|---|---|
| Enable | A media reactivated in an incoming answer is ignored. This behaviour goes against the SDP Offer/Answer model described by IETF RFC 3264. |
| Disable | A media reactivated in an incoming answer ends the current media negotiation and the call. This behaviour follows the SDP Offer/Answer model described by IETF RFC 3264. |

3. Set the *Allow Audio and Image Negotiation* drop-down menu with the behaviour of the Mediatrix 4102 when offering media or answering to a media offer with audio and image negotiation

**Table 60:** Audio and Image Negotiation Parameters

| Parameter | Description |
|---|---|
| Enable | The unit offers audio and image media simultaneously in outgoing SDP offers and transits to T.38 mode upon reception of a T.38 packet. Also, when the unit answers positively to a SDP offer with audio and image, it transits to T.38 mode upon reception of a T.38 packet. |
| Disable | Outgoing offers never include image and audio simultaneously. Incoming offers with audio and image media with a non-zero port are considered as offering only audio. |

4. Set the *Transfer Version* drop-down with the proper transfer version.

**Table 61:** Call Transfer Versions Supported

| Version | Description |
|---|---|
| Transfer 02 | The Mediatrix 4102 executes transfers by using the methods described in the now expired *draft-ietf-sip-cc-transfer-02.txt*. Its use is deprecated and you should use this setting for backward compatibility issues only. |
| Transfer 05 Using Refer 02 | The Mediatrix 4102 executes transfers by using the methods described in the more recent *draft-ietf-sip-cc-transfer-05.txt*. This draft version contains several enhancements over the previous ones. Among others, it is possible to use the *Replaces* header to provide a more seamless attended transfer to the user. This method also uses *draft-ietf-sip-refer-02.txt*. Use this setting if you do not need to interop with transfer02-enabled parties. See "Replaces Configuration Setting" on page 300 for more details. |
| Sipping Transfer 01 Using Refer RFC 3515 | The Mediatrix 4102 executes transfers by using the methods described in *draft-ietf-sipping-cc-transfer-01.txt*. This draft version is more recent than Transfer 02 and Transfer 05 Using Refer 02. This method also uses the *RFC 3515 - The Session Initiation Protocol (SIP)* Refer Method. |

5. Click *Submit* if you do not need to set other parameters.

# SIP Authentication

The *Authentication* sub-page of the *SIP* page allows you to configure the unit and user agent authentication parameters of the Mediatrix 4102.

| **Standards Supported** | Basic and Digest authentication as per RFC 3261 |
|---|---|

Authentication information allows you to add some level of security to the Mediatrix 4102 lines by setting user names and passwords. You can add two types of authentication information:

- user agent specific authentication

  You can define up to five user names and five passwords for each user agent of the Mediatrix 4102. A user agent can thus register with five different realms.

- unit authentication

  You can define up to five user names and five passwords for the Mediatrix 4102. These user names and passwords apply to all lines of the unit.

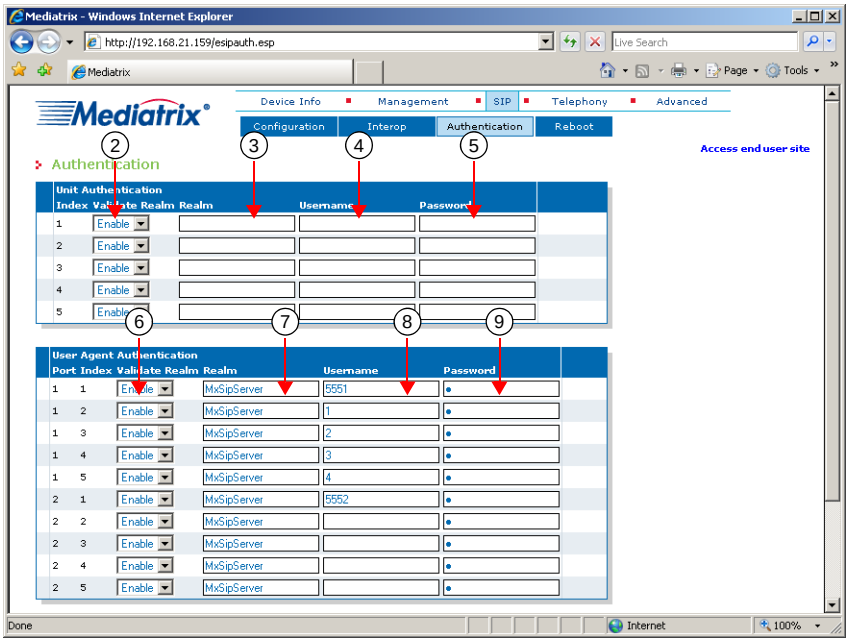
When a realm requests authentication, the user agent specific authentication is tried first, and then the unit authentication if required.

You can also set these parameters via SNMP, as described in "Authentication" on page 290.

### To set the SIP security parameters:

1. In the web interface, click the *SIP* link, then the *Authentication* sub-link.

**Figure 43:** SIP – Authentication Web Page



2. In the *Unit Authentication* section, select whether or not the current unit credentials are valid for any realm in the corresponding *Validate Realm* drop-down menu.

**Table 62:** Realm Authentication Parameters

| Parameter | Description |
|---|---|
| Disable | The current unit credentials are valid for any realm. The corresponding *Realm* field is read-only and cannot be modified. |
| Enable | The unit credentials are used only for a specific realm set in the corresponding *Realm* field. |

3. Enter a realm for each authentication row in the *Realm* column.

   When authentication information is required from users, the realm identifies who requested it.

4. Enter a string that uniquely identifies this endpoint in the realm in the *Username* column.

5. Enter a user password in the *Password* column.

6. In the *User Agent Authentication* section, select whether or not the current user agent credentials are valid for any realm in the corresponding *Validate Realm* drop-down menu.

**Table 63:** Realm Authentication Parameters

| Parameter | Description |
|---|---|
| Disable | The current user agent credentials are valid for any realm. The corresponding *Realm* field is read-only and cannot be modified. |
| Enable | The user agent credentials are used only for a specific realm set in the corresponding *Realm* field. |

7. Enter up to five realms for each user agent in the *Realm* column.

   When authentication information is required from users, the realm identifies who requested it.

8. Enter a string that uniquely identifies this user agent in the realm in the *Username* column.

9. Enter a user password in the *Password* column.

10. Click *Submit* if you do not need to set other parameters.

CHAPTER 6

# Web Interface – Telephony

The *Telephony* page allows you to configure the various telephony parameters of the Mediatrix 4102.

## Digit Maps

| | |
|---|---|
| **Standards Supported** | RFC 2705 – Media Gateway Control Protocol (MGCP) Version 1.0, section 3.4 (Formal syntax description of the protocol). |

A digit map allows you to compare the number users just dialed to a string of arguments. If they match, users can make the call. If not, users cannot make the call and get an error signal. It is thus essential to define very precisely a digit map before actually implementing it, or your users may encounter calling problems.

Because the Mediatrix 4102 cannot predict how many digits it needs to accumulate before transmission, you could use the digit map, for instance, to determine exactly when there are enough digits entered from the user to place a call.

### Syntax

The permitted digit map syntax is taken from the core MGCP specification, RFC 2705, section 3.4:

```
DigitMap = DigitString  / '(' DigitStringList ')'
DigitStringList = DigitString 0*( '|' DigitString )
DigitString = 1*(DigitStringElement)
DigitStringElement = DigitPosition ['.']
DigitPosition = DigitMapLetter / DigitMapRange
DigitMapLetter = DIGIT / '#' / '*' / 'A' / 'B' / 'C' / 'D' / 'T'
DigitMapRange =  'x' / '[' 1*DigitLetter ']'
DigitLetter ::= *((DIGIT '-' DIGIT ) / DigitMapLetter)
```

Where “x” means “any digit” and “.” means “any number of”.

For instance, using the telephone on your desk, you can dial the following numbers:

**Table 64:** Number Examples

| Number | Description |
|---|---|
| 0 | Local operator |
| 00 | Long distance operator |
| xxxx | Local extension number |
| 8xxxxxxx | Local number |
| #xxxxxxx | Shortcut to local number at other corporate sites |
| 91xxxxxxxxxx | Long distance numbers |
| 9011 + up to 15 digits | International number |

The solution to this problem is to load the Mediatrix 4102 with a digit map that corresponds to the dial plan.

A Mediatrix 4102 that detects digits or timers applies the current dial string to the digit map, attempting a match to each regular expression in the digit map in lexical order.

- If the result is under-qualified (partially matches at least one entry in the digit map), waits for more digits.
- If the result matches, dials the number.
- If the result is over-qualified (i.e., no further digits could possibly produce a match), sends a fast busy signal.

## Special Characters

Digit maps use specific characters and digits in a particular syntax. Those characters are:

**Table 65:** Digit Map Characters

| Character | Use |
|---|---|
| Digits (0, 1, 2... 9) | Indicates specific digits in a telephone number expression. |
| T | The Timer indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the SIP Server can make the call. |
| x | Matches any digit, excluding “#” and “*”. |
| \| | Indicates a choice of matching expressions (OR). |
| . | Matches an arbitrary number of occurrences of the preceding digit, including 0. |
| [ | Indicates the start of a range of characters. |
| ] | Indicates the end of a range of characters. |

## How to Use a Digit Map

Let’s say you are in an office and you want to call a co-worker’s 3-digits extension. You could build a digit map that says “after the user has entered 3 digits, make the call”. The digit map could look as follows:

```
xxx
```

You could refine this digit map by including a range of digits. For instance, you know that all extensions in your company either begin with 2, 3, or 4. The corresponding digit map could look as follows:

```
[2-4]xx
```

If the number you dial begins with anything other than 2, 3, or 4, the call is not placed and you get a busy signal.

### Combining Several Expressions

You can combine two or more expressions in the same digit map by using the “|” operator, which is equal to OR.

Let’s say you want to specify a choice: the digit map is to check if the number is internal (extension), or external (a local call). Assuming that you must first dial “9” to make an external call, you could define a digit map as follows:

```
([2-4]xx|9[2-9]xxxxxx)
```

The digit map checks if:

- the number begins with 2, 3, or 4 **and**
- the number has 3 digits

If not, it checks if:

- the number begins with 9 **and**
- the second digit is any digit between 2 and 9 **and**
- the number has 7 digits

**Note:** Enclose the digit map in parenthesis when using the “|” option.

## Using the # and * Characters

It may sometimes be required that users dial the "#" or "*" to make calls. This can be easily incorporated in a digit map:

```
xxxxxxx#
xxxxxxx*
```

The "#" or "*" character could indicate users must dial the "#" or "*" character at the end of their number to indicate it is complete. You can specify to remove the "#" or "*" found at the end of a dialed number. See "General Parameters" on page 90.

## Using the Timer

You can configure the Timer. See "General Parameters" on page 90 for more details. It indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the Mediatrix 4102 can make the call. A digit map for this could be:

```
[2-9]xxxxxxT
```

**Note:** When making the actual call and dialing the number, the Mediatrix 4102 automatically removes the "T" found at the end of a dialed number, if there is one (after a match). This character is for indication purposes only.

## Calls Outside the Country

If your users are making calls outside their country, it may sometimes be hard to determine exactly the number of digits they must enter. You could devise a digit map that takes this problem into account:

```
001x.T
```

In this example, the digit map looks for a number that begins with 001, and then any number of digits after that (x.).

## Example

Table 64 on page 87 outlined various call types one could make. All these possibilities could be covered in one digit map:

```
(0T|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|91xxxxxxxxxx|9011x.T)
```

## Validating a Digit Map

The Mediatrix 4102 validates the digit map as you are entering it and it forbids any invalid value.

## General Parameters

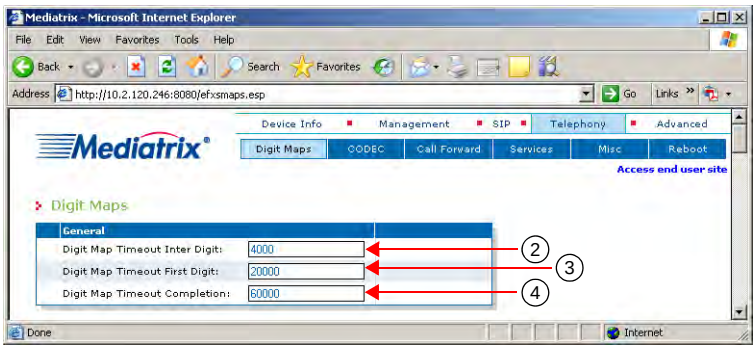
The following are the general digit maps parameters you can set.

You can also set these parameters via SNMP, as described in "Chapter 22 - Digit Maps" on page 321.

▶ **To set the general digit map parameters:**

1. In the web interface, click the *Telephony* link, then the *Digit Maps* sub-link.

**Figure 44:** Telephony – Digit Maps Web Page




2. In the *General* section, define the value of the "T" digit in the *Digit Map Timeout Inter Digit* field.

   The "T" digit expresses a time lapse between the detection of two DTMFs. This value is expressed in milliseconds (ms). Values range from 500 ms to 10000 ms.

3. Define the time between the start of the dial tone and the receiver off-hook tone, if no DTMF is detected, in the *Digit Map Timeout First Digit* field.

   This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms.

4. Define the total time the user has to dial the DTMF sequence in the *Digit Map Timeout Completion* field.

   The timer starts when the dial tone is played. When the timer expires, the receiver off-hook tone is played. This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms.

5. Click *Submit* if you do not need to set other parameters.

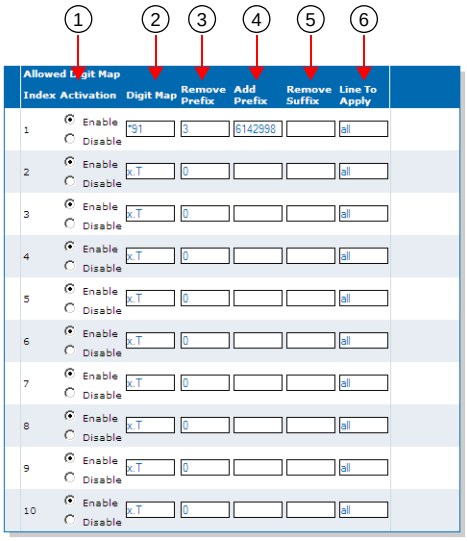
## Allowed Digit Maps

You can create/edit ten digit maps for the Mediatrix 4102. Digit map rules are checked sequentially. If a telephone number potentially matches two of the rules, the first rule encountered is applied.

▶ **To set up digit maps:**

1. In the *Allowed Digit Map* section – *Activation* column, enable one or more digit maps by selecting the corresponding **Enable** choice.

**Figure 45:** Allowed Digit Map Section




2. Define the digit map string that is considered valid when dialed in the *Digit Map* column.

   The string must use the syntax described in "Digit Maps" on page 87. A digit map string may have a maximum of 64 characters.

3. Define the amount of digits to remove from the beginning of the dialed number, after dialing, but before initiating the call, in the *Remove Prefix* column.

   For instance, when dialing "1-819-xxx-xxxx", specifying a value of "4" means that the call is started by using the number "xxx-xxxx".

   This rule is applied BEFORE applying both the *Add Prefix* (Step 4) and *Remove Suffix* (Step 5) rules.

4. Define the string to insert at the beginning of the dialed number before initiating the call in the *Add Prefix* column.

   For instance, let's say that you need to dial a special digit, "9", for all local calls. Dialing "xxx-xxxx" with a value of "9" would yield "9-xxx-xxxx" as the number with which to initiate the call.

   This rule is applied AFTER applying both the *Remove Prefix* (Step 3) and *Remove Suffix* (Step 5) rules.

5. Define the string to look for and remove, from the end of the dialed number, in the *Remove Suffix* column.

   This is helpful if one of the digit maps contains a terminating character that must not be dialed.

   For instance, in a digit map such as "25#", the "#" signals that the user has finished entering digits. To remove the "#", specify "#" in this field and the resulting number is "25".

   This rule is applied AFTER applying the *Remove Prefix* (Step 3) rule, but BEFORE applying *Add* the *Prefix* (Step 4) rule.

6. Specify the line(s) on which to apply the digit map in the *Line To Apply* column.

   The string has the following syntax:

   - ***all***: Applies to all lines.

- **,**: Separator between non-consecutive lists of lines or single line.
- ***n***: A single line, where n is the line number.
- ***m-n***: List of lines where m is the start line number and n is the end line number.

**Note:** Line duplication is not allowed. Lines must be specified in low to high order.

Example:

```
'1,3-6': Applies to lines 1, 3, 4, 5, and 6.
```

The default value is **all**.

7. Click *Submit* if you do not need to set other parameters.

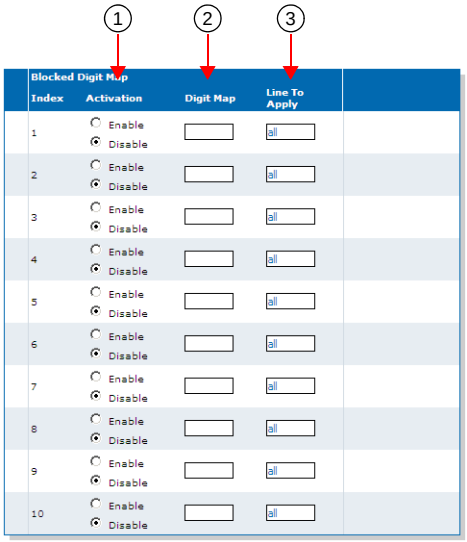
## Blocked Digit Maps

A blocked digit map forbids to call specific numbers; for instance, you want to accept all 1-8xx numbers except 1-801. You can create/edit ten blocked digit maps for the Mediatrix 4102.

**To set up blocked digit maps:**

1. In the *Blocked Digit Map* section – *Activation* column, enable one or more digit maps by selecting the corresponding **Enable** choice.

**Figure 46:** Blocked Digit Map Section




2. Define the digit map string that is considered invalid when dialed in the *Digit Map* column.

   The string must use the syntax described in "Digit Maps" on page 87. The string format is validated upon entry. Invalid entries are refused. A digit map string may have a maximum of 64 characters.

3. Specify the line(s) on which to apply the digit map in the *Line To Apply* column.

   The string has the following syntax:

   - ***all***: Applies to all lines.
   - **,**: Separator between non-consecutive lists of lines or single line.
   - ***n***: A single line, where n is the line number.
   - ***m-n***: List of lines where m is the start line number and n is the end line number.

**Note:** Line duplication is not allowed. Lines must be specified in low to high order.

Example:

`'1,3-6': Applies to lines 1, 3, 4, 5, and 6.`

The default value is **all**.

4. Click *Submit* if you do not need to set other parameters.

# Voice & Fax Codecs

The two lines of the Mediatrix 4102 can simultaneously use the same codec (for instance, G.711 PCMA), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

**Table 66:** Codecs Comparison

| | Compression | Voice Quality |
|---|---|---|
| **G.711** | None | Excellent |
| **G.726** | Medium | Fair |
| **G.729a/ab** | High | Fair/Good |

## G.711 PCMA and PCMU

Specified in ITU-T Recommendation G.711. The audio data is encoded as 8 bits per sample, after logarithmic scaling. PCMU denotes µ-law scaling, PCMA A-law scaling.

**Table 67:** G.711 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. See "G.711 Codec Parameters" on page 98 for more details. |
| Voice Activity Detection (VAD) | Can be enabled or disabled. See "G.711 Codec Parameters" on page 98 for more details. |
| Comfort noise | Supports comfort noise as defined in *draft-ietf-avt-rtp-cn-05.txt*. See "G.711 Codec Parameters" on page 98 for more details. |

### Analog Modem

The Mediatrix 4102 can send modem transmissions in clear channel (G.711). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

## G.726

Specified in ITU-T Recommendation G.726: 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM). It describes the algorithm recommended for conversion of a single 64 kbit/s A-law or U-law PCM channel encoded at 8000 samples/sec to and from a 40, 32, 24, or 16 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique.

**Table 68:** G.726 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. See "G.726 Codecs Parameters" on page 99 for more details. |
| Voice Activity Detection (VAD) | Can be enabled or disabled. See "G.726 Codecs Parameters" on page 99 for more details. |
| Comfort noise | Supports comfort noise as defined in *draft-ietf-avt-rtp-cn-05.txt*. See "G.726 Codecs Parameters" on page 99 for more details. |

### Analog Modem

The Mediatrix 4102 can send modem transmissions in clear channel (G.726). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

## G.729

Specified in ITU-T Recommendation G.729, coding of speech at 8 kbit/s using conjugate structure-algebraic code excited linear prediction (CS-ACELP). For all data rates, the sampling frequency (and RTP timestamp clock rate) is 8000 Hz.

A voice activity detector (VAD) and comfort noise generator (CNG) algorithm in Annex B of G.729 is recommended for digital simultaneous voice and data applications; they can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B comfort noise frame occupies 2 octets.

The Mediatrix 4102 supports G.729A and G.729AB for encoding and G.729, G.729A and G.729AB for decoding.

**Table 69:** G.729 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. See "G.729 Codec Parameters" on page 99 for more details. |
| Voice Activity Detection (VAD) | The Mediatrix 4102 supports the annex B. Annex B is the built-in support of VAD in G.729. See "G.729 Codec Parameters" on page 99 for more details. |

## General Parameters

The following are the general codecs parameters you can set.

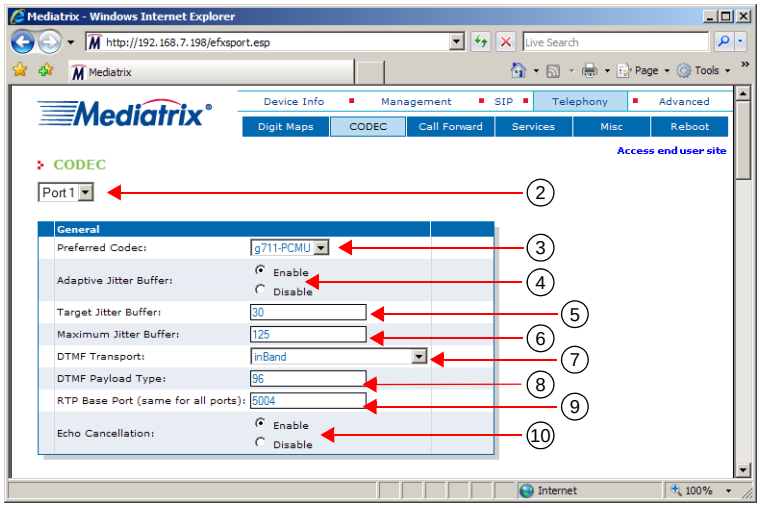
| | |
|---|---|
| **Standards Supported** | • draft-choudhuri-sip-info-digit-00.txt<br>• ITU-T Recommendation Q.24 : Multifrequency push-button signal reception<br>• RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals<br>• RFC 1890 – RTP Profile for Audio and Video Conferences with Minimal Control |

You can also set these parameters via SNMP, as described in "Chapter 17 - Voice Transmissions" on page 263.

▶ **To set the general codecs parameters:**

1. In the web interface, click the *Telephony* link, then the *CODEC* sub-link.

**Figure 47:** Telephony – CODEC Web Page




2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *General* section, choose the preferred codec you want to use in the *Preferred Codec* field. This is the codec you want to favour during negotiation. You have the following choices:
   - g711-PCMU
   - g711-PCMA
   - g729
   - g726-16kbps
   - g726-24kbps
   - g726-32kbps
   - g726-40kbps

   The default value is **pcmu**.

4. Enable the jitter buffer protection by selecting **Enable** in the *Adaptive Jitter Buffer* choice.

   The jitter buffer allows better protection against packet loss, but increases the voice delay. If the network to which the Mediatrix 4102 is connected suffers from a high level of congestion, the jitter buffer protection level should be higher. If the network to which the Mediatrix 4102 is connected suffers from a low level of congestion, the jitter buffer protection level should be lower.

**Note:** Do not put **0** as values for the *Target Jitter Buffer* and *Maximum Jitter Buffer* fields.

5. Define the target jitter buffer length in the *Target Jitter Buffer* field.

   The adaptive jitter buffer attempts to hold packets to the target holding time. This is the minimum delay the jitter buffer adds to the system. The target jitter buffer length is in ms and must be equal to or smaller than the maximum jitter buffer.

   Values range from 0 ms to 135 ms. The default value is 30 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

   It is best not to set target jitter values below the default value. Setting a target jitter buffer below 5 ms could cause an error. Jitter buffer adaptation behaviour varies from one codec to another. See "About Changing Jitter Buffer Values" on page 97 for more details.

6. Define the maximum jitter buffer length in the *Maximum Jitter Buffer* field.

   This is the maximum jitter the adaptive jitter buffer can handle. The jitter buffer length is in ms and must be equal to or greater than the target jitter buffer.

   Values range from 0 ms to 135 ms. The default value is 125 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

   The maximum jitter buffer value should be equal to the minimum jitter buffer value + 4 times the ptime value. Let's say for instance that:

   - Minimum jitter buffer value is 30 ms
   - Ptime value is 20 ms

     The maximum jitter buffer value should be: 30 + 4x20 = 110 ms

   See "About Changing Jitter Buffer Values" on page 97 for more details.

7. Set the DTMF transport type in the *DTMF Transport* field.

**Table 70:** DTMF Transport Type Parameters

| Transport Parameter | Description |
|---|---|
| inBand | The DTMFs are transmitted like the voice in the RTP stream. |
| outOfBandUsingRtp | The DTMFs are transmitted as per RFC 2833. |
| outOfBandUsingSignalingProtocol | The DTMFs are transmitted as per *draft-choudhuri-sip-info-digit-00.txt*.<br>**Note**: This feature and the Hook Flash processing feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See "Hook Flash Processing" on page 352 for more details. |

## DTMF out-of-band

Certain compression codecs such as G.729 effectively distort voice because they lose information from the incoming voice stream during the compression and decompression phases. For normal speech this is insignificant and becomes unimportant. In the case of pure tones (such as DTMF) this distortion means the receiver may no longer recognize the tones. The solution is to send this information as a separate packet to the other endpoint, which then plays the DTMF sequence back by re-generating the true tones. Such a mechanism is known as out-of-band DTMF. The Mediatrix 4102 receives and sends out-of-band DTMFs as per ITU Q.24. DTMFs supported are 0-9, A-D, *, #.

**Table 70:** DTMF Transport Type Parameters (Continued)

| Transport Parameter | Description |
|---|---|
| signalingProtocolDependent | The signalling protocol has the control to select the DTMF transport mode. The SDP body includes both RFC 2833 and *draft-choudhuri-sip-info-digit-00.txt* in that order of preference. |

8. Set the payload type in the *DTMF Payload Type* field.

   You can determine the actual RTP dynamic payload type used for the “telephone-event” in an initial offer. The payload types available are as per RFC 1890. Available values range from 96 to 127.

**Note:** This parameter applies only when selecting the *outOfBandUsingRtp* DTMF transport mode.

9. Set the *RTP Base Port* field with the port number you want to use as RTP/RTCP base port.

   The RTP/RTCP ports are allocated starting from the base port. The Mediatrix 4102 may use two or four RTP/RTCP ports for each FXS interface:

   - It uses two ports in case of a standard call.
   - It uses four ports in other types of calls such as a conference call, a call transfer, etc.

   The default RTP/RTCP base port is **5004**. In the case of the base port defined on 5004:

   - If there is currently no ongoing call and FXS connector 1 has an incoming or outgoing call, it uses the RTP/RTCP ports 5004 and 5005.
   - If there is currently a standard call on FXS connector 1 and FXS connector 2 has a conference call, then FXS connector 2 uses the RTP/RTCP ports 5006, 5007, 5008, and 5009, which are the next available ports.

10. Select whether the echo cancellation should be enabled or disabled in the *Echo Cancellation* choice.

**Table 71:** Echo Cancellation Parameters

| Parameter | Description |
|---|---|
| disable | The DSP does not use echo cancellation on the related port. |
| enable | The DSP proceeds to cancel signals that are recognized as echo when appropriate. This is the default value. |

   Turning off the echo cancellation feature may be useful to ensure the success of some modem transmissions.

11. Click *Submit* if you do not need to set other parameters.

## About Changing Jitter Buffer Values

Media5 recommends to avoid changing the target and maximum jitter buffer values unless experiencing or strongly expecting one of the following symptoms:

- If the voice is scattered, try to increase the maximum jitter buffer value.
- If the delay in the voice path (end to end) is too long, you can lower the target jitter value, but ONLY if the end-to-end delay measured matches the target jitter value.

  For instance, if the target jitter value is 50 ms, the maximum jitter is 135 ms and the delay measured is 130 ms, it would serve nothing to reduce the target jitter. However, if the target jitter value is 100 ms and the measured delay is between 100 ms and 110 ms, then you can lower the target jitter from 100 ms to 30 ms.

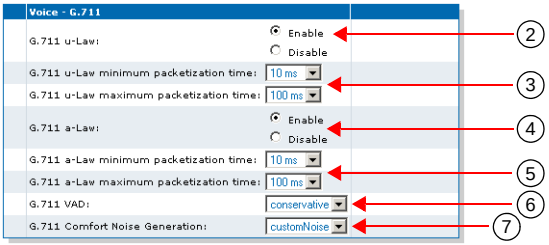
## G.711 Codec Parameters

The following are the G.711 codec parameters you can set.

▶ **To set the G.711 codec parameters:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Voice - G.711* section, enable the G.711 u-Law codec by selecting **Enable** in the *G.711 u-Law* choice.

**Figure 48:** Telephony – G.711 Section




3. Set the minimum and maximum packetization time values for the G.711 u-Law codec in the corresponding drop-down menu.

   The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

   - ***Minimum***: Shortest packetization period allowed for the G.711 u-Law codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
   - ***Maximum***: Longest packetization period allowed for the G.711 u-Law codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.
4. Enable the G.711 a-Law codec by selecting **Enable** in the *G.711 a-Law* field.
5. Set the minimum and maximum packetization time values for the G.711 a-Law codec in the corresponding drop-down menu.

   The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

   - ***Minimum***: Shortest packetization period allowed for the G.711 a-Law codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
   - ***Maximum***: Longest packetization period allowed for the G.711 a-Law codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.
6. Enable the G.711 Voice Activity Detection (VAD) by selecting **Enable** in the *G.711 VAD* choice.

   VAD defines how the Mediatrix 4102 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, VAD may affect packets that are not really silent (for instance, cut sounds that are too low). VAD can thus slightly affect the voice quality.
7. Enable the G.711 Comfort Noise Generation (CNG) by selecting **Enable** in the *G.711 Comfort Noise Generation* choice.

   Comfort Noise (CN) defines how the Mediatrix 4102 processes silence periods information it receives. During silence periods, the Mediatrix 4102 may receive CN packets containing information about background noise. Those packets are used to generate local comfort noise.
8. Click *Submit* if you do not need to set other parameters.

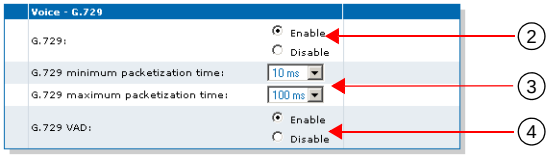
## G.729 Codec Parameters

The following are the G.729 codec parameters you can set.

▶ **To set the G.729 codec parameters:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Voice - G.729* section, enable the G.729 codec by selecting **Enable** in the *G.729* choice.

**Figure 49:** Telephony – G.729 Section




3. Set the minimum and maximum packetization time values for the G.729 codec in the corresponding drop-down menu.

   The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

   - ***Minimum***: Shortest packetization period allowed for the G.729 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
   - ***Maximum***: Longest packetization period allowed for the G.729 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.

4. Enable the G.729 Voice Activity Detection (VAD) by selecting **Enable** in the *G.729 VAD* choice.

   VAD defines how the Mediatrix 4102 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, VAD may affect packets that are not really silent (for instance, cut sounds that are too low). VAD can thus slightly affect the voice quality.

   G.729 has a built-in VAD in its Annex B version. It is recommended for digital simultaneous voice and data applications and can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B frame occupies 2 octets. The CN packets are sent in accordance with annex B of G.729.

5. Click *Submit* if you do not need to set other parameters.

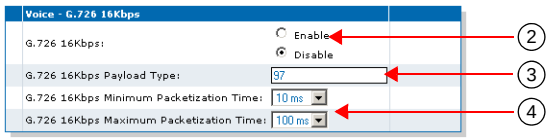
## G.726 Codecs Parameters

The following are the G.726 codecs parameters you can set. There are sections for each type of G.726 codec.

▶ **To set the G.726 codecs parameters:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In one or more of the *Voice - G.726* sections, enable the corresponding G.726 codec by selecting **Enable** in the *G.726* choice.

**Figure 50:** Telephony – G.726 Section



3. Set the G.726 actual RTP dynamic payload type used in an initial offer in the *G.726 Payload Type* field.

   The payload types available are as per RFC 3551. The values range from 96 to 127. The default values are as follows:

   - G.726 16 kbps: 97.
   - G.726 24 kbps: 98.
   - G.726 32 kbps: 99.
   - G.726 40 kbps: 100.

**Note:** When selecting the dynamic payload type, make sure that the value is not already used by another dynamic codec. If a value between 96 and 127 is refused, this means it is already used by another dynamic codec.

**Note:** If you set the *DTMF Transport* field to **outOfBandUsingSignalingProtocol** ("General Parameters" on page 95), you cannot configure a dynamic payload type to 111 because it is already used by the DTMF out-of-band using signalling protocol.

4. Set the minimum and maximum packetization time values for the G.726 codec in the corresponding drop-down menu.

   The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

   - ***Minimum***: Shortest packetization period allowed for the G.726 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
   - ***Maximum***: Longest packetization period allowed for the G.726 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.

5. Click *Submit* if you do not need to set other parameters.

## Fax Parameters

The Mediatrix 4102 handles G3 fax transmissions at speeds up to 14.4 kbps. Automatic fax mode detection is standard on all lines. Real-Time Fax Over UDP with the T.38 protocol stack is also available.

The quality of T.38 fax transmissions depends upon the system configuration, type of call control system used, type of Mediatrix units deployed, as well as the model of fax machines used. Should some of these conditions be unsatisfactory, performance of T.38 fax transmissions may vary and be reduced below expectations.

A fax call works much like a regular voice call, with the following differences:

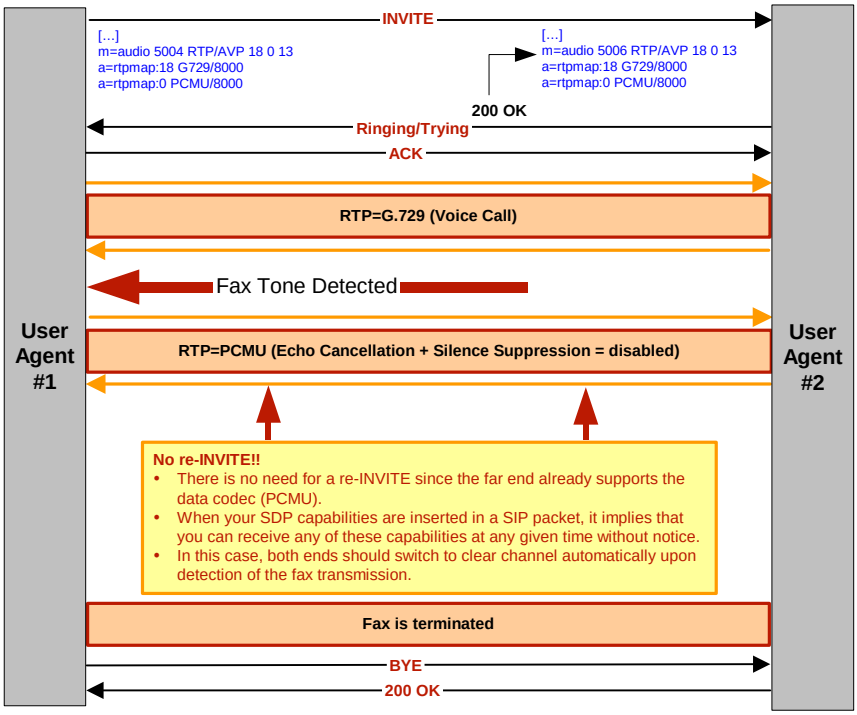
1. The fax codec may be re-negotiated by using a re-INVITE.
2. The goal of the re-INVITE is to allow both user agents to agree on a fax codec, which is either:
   a. Clear channel (PCMU/PCMA or G.726) without Echo Cancellation nor Silence Suppression (automatically disabled).
   b. T.38.
3. Upon fax termination, if the call is not BYE, the previous voice codec is recovered with another re-INVITE.

All lines of the Mediatrix 4102 can simultaneously use the same codec (for instance, T.38), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

## Clear Channel Fax

The Mediatrix 4102 can send faxes in clear channel. The following is a clear channel fax call flow:

**Figure 51:** Clear Channel Fax Call Flow




## T.38 Fax

| | |
|---|---|
| **Standards Supported** | • Based on *draft-ietf-sipping-realtimefax-01.txt*<br>• Recommendation ITU T.38 version 0 |

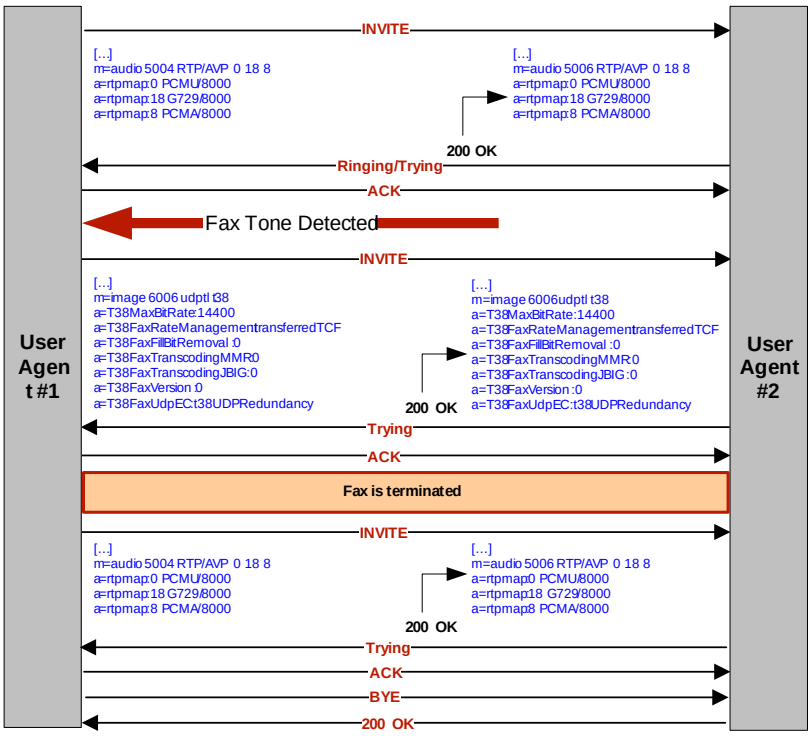
T.38 fax relay is a real-time fax transmission; that is, two fax machines communicating with each other as if there were a direct phone line between the two. T.38 is called a fax relay, which means that instead of sending inband fax signals, which implies a loss of signal quality, it sends those fax signals out-of-band in a T.38 payload, so that the remote end can reproduce the signal locally.

The Mediatrix 4102 can send faxes in T.38 mode over UDP or TCP. T.38 is used for fax if both units are T.38 capable; otherwise, transmission in clear channel over G.711 as defined is used (if *G.711 µ-law* and/or *G.711 A-law* are enabled). If no clear channel codecs are enabled and the other endpoint is not T.38 capable, the fax transmission fails.

**Caution:** The Mediatrix 4102 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4102 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message. See “Fax Issues” on page 385 for more details.

The following is a T.38 fax call flow:

**Figure 52:** T.38 Fax Call Flow



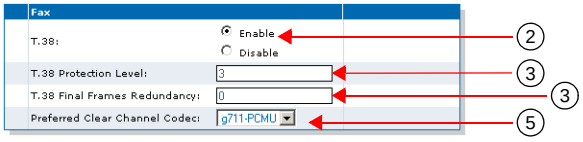

## Fax Parameters Configuration

The following are the fax codecs parameters you can set. You can also set these parameters via SNMP, as described in "Chapter 18 - Fax Transmission" on page 279.

▶ **To set the fax codecs parameters:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Fax* section, enable the T.38 codec by selecting **Enable** in the *T.38* choice.

**Figure 53:** Telephony – Fax Section




3. Set the number of redundancy packets sent with the current packet in the *T.38 Protection Level* field.

   This is the standard redundancy offered by T.38. Please see step ***4*** for additional reliability options for T.38. Available values range from 1 to 5.

4. For additional reliability, define the number of times T.38 packets are retransmitted in the *T.38 Final Frames Redundancy* field.

   This only applies to the T.38 packets where the PrimaryUDPTL contains the following T.38 data type:

   - HDLC_SIG_END,
   - HDLC_FCS_OK_SIG_END,
   - HDLC_FCS_BAD_SIG_END and
   - T4_NON_ECM_SIG_END

**5.** Set the clear channel codec to use upon detecting a fax tone in the *Preferred Clear Channel Codec* field.

This is used to decide which of the following codecs is preferred, even for voice transmissions:

- PCMU
- PCMA
- G.726 at 32 kbs
- G.726 at 40 kbs

**Note:** In clear channel, G.726 at 16 kbs and 24 kbs are not recommended for fax transmission.

- noPreferredCodec

  When *noPreferredCodec* is selected and no data-capable codecs are negociated, data transmission may fail.

This parameter increases the relative priority of the selected codec vs other data-capable codecs. However, the priority of the preferred clear channel codec remains lower than the voice's *Preferred Codec* field. (see "General Parameters" on page 90).

Moreover, when no data-capable codec is part of the list of negotiated codecs, this variable indicates which codec to use when fax or modem tones are detected. However, if the negotiated voice codec is data-capable, the voice codec will be used for data instead of the preferred data codec. See "Data Codec Selection Procedure" on page 283 for more details.

**6.** Click *Submit* if you do not need to set other parameters.

# Call Forward

The *Call Forward* sub-page of the *Telephony* page allows you to set the three types of Call Forward:

- On Busy
- On No Answer
- Unconditional

You can also set these parameters via SNMP, as described in "Call Forward" on page 336.

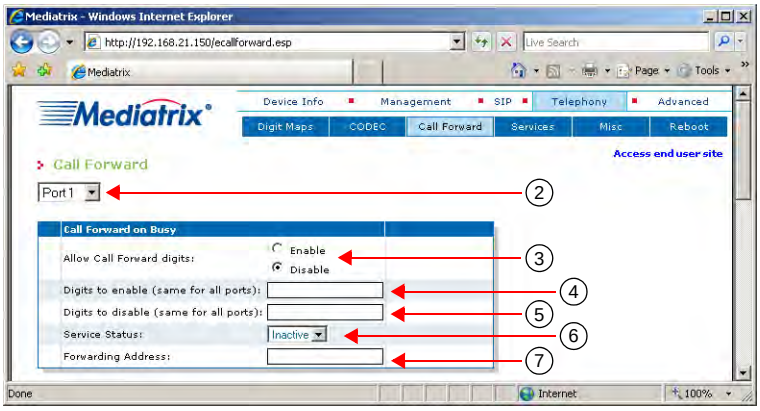
## On Busy

You can automatically forward the incoming calls of your users to a pre-determined target if they are already on the line. The user does not have any feedback that a call was forwarded.

**To set the Call Forward On Busy feature:**

1. In the web interface, click the *Telephony* link, then the *Call Forward* sub-link.

**Figure 54:** Telephony – Call Forward Web Page




2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Call Forward on Busy* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.

   If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.
4. Define the digits that users must dial to start the service in the *Digits to enable* field.

   Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.

   For instance, you could decide to put "*72" as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see "Digit Maps" on page 87). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
5. Define the digits that users must dial to stop the service in the *Digits to disable* field.

   Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.

   For instance, you could decide to put "*73" as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see "Digit Maps" on page 87). Dialing this digit map does not have any effect unless the service's status is "enabled".

The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

6. Set the activation status of the service in the *Service Status* field to **Inactive** or **Active**.

   This feature starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 4 and 5. The *Service Status* field is automatically updated to reflect the activation status according to the user's setting.

7. Define the address to which forward incoming calls in the *Forwarding Address* field.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

8. Click *Submit* if you do not need to set other parameters.

## Using Call Forward on Busy

The following is the procedure to use this service on the user's telephone.

**To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward on busy service.

   This sequence could be something like *72.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.

   The call forward is established.
7. Hang up your telephone.

**To cancel the call forward:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward on busy service.

   This sequence could be something like *73.
4. Wait for three "beeps" followed by a silent pause.

   The call forward is cancelled.
5. Hang up your telephone.

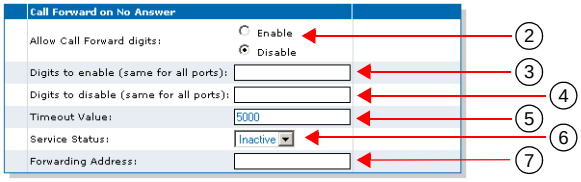
## On No Answer

You can forward the incoming calls of your users to a pre-determined target if they do not answer their telephone before a specific amount of time. The user does not have any feedback that a call was forwarded.

▶ **To set the Call Forward On No Answer feature:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Forward on No Answer* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.

**Figure 55:** Telephony – Call Forward on No Answer section




If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

3. Define the digits that users must dial to start the service in the *Digits to enable* field.

   Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.

   For instance, you could decide to put “*74” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see “Digit Maps” on page 87). Dialing this digit map does not have any effect unless the service’s status is “enabled”.

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

4. Define the digits that users must dial to stop the service in the *Digits to disable* field.

   Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.

   For instance, you could decide to put “*75” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see “Digit Maps” on page 87). Dialing this digit map does not have any effect unless the service’s status is “enabled”.

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

5. Define the time, in milliseconds, the telephone keeps ringing before the call forwarding activates in the *Timeout Value* field.
6. Set the status of the service in the *Service Status* field to **Inactive** or **Active**.

   This feature starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *Service Status* field is automatically updated to reflect the activation status according to the user’s setting.

7. Define the address to which forward incoming calls in the *Forwarding Address* field.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

8. Click *Submit* if you do not need to set other parameters.

### Using Call Forward on No Answer

The following is the procedure to use this service on the user's telephone.

▶ **To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward on no answer service.
   This sequence could be something like *74.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.
   The call forward is established.
7. Hang up your telephone.

▶ **To cancel the call forward:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward on no answer service.
   This sequence could be something like *75.
4. Wait for three "beeps" followed by a silent pause.
   The call forward is cancelled.
5. Hang up your telephone.

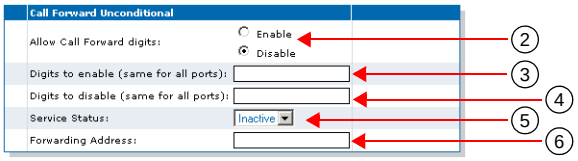
## Unconditional

The Call Forward Unconditional feature allows users to forward all of their calls to another extension or line.

▶ **To set the Call Forward Unconditional feature:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Forward Unconditional* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.

**Figure 56:** Telephony – Call Forward Unconditional Section




If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

3. Define the digits that users must dial to start the service in the *Digits to enable* field.

   Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 5.

   For instance, you could decide to put "*74" as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see "Digit Maps" on page 87). Dialing this digit map does not have any effect unless the service's status is "enabled".

The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

4. Define the digits that users must dial to stop the service in the *Digits to disable* field.

   Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 5.

   For instance, you could decide to put “*75” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see “Digit Maps” on page 87). Dialing this digit map does not have any effect unless the service’s status is “enabled”.

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

5. Set the status of the service in the *Service Status* field to **Inactive** or **Active**.

   This feature starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *Service Status* field is automatically updated to reflect the activation status according to the user’s setting.

6. Define the address to which forward incoming calls in the *Forwarding Address* field.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

7. Click *Submit* if you do not need to set other parameters.

## Using Call Forward Unconditional

When forwarding calls outside the system, a brief ring is heard on the telephone to remind the user that the call forward service is active. The user can still make calls from the telephone.

**To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward unconditional service.

   This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three “beeps” followed by a silent pause.

   The call forward is established.
7. Hang up your telephone.

**To check if the call forward has been properly established:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial your extension or telephone number.

   The call is forwarded to the desired telephone number.
4. Hang up your telephone.

**To cancel the call forward:**

1. Take the receiver off-hook.

2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward – unconditional service.
   This sequence could be something like *71.
4. Wait for three “beeps” followed by a silent pause.
   The call forward is cancelled.
5. Hang up your telephone.

# Services

The *Services* sub-page of the *Telephony* page allows you to set the following subscriber services:

- Call Transfer
- Call Waiting
- Conference
- Hold
- Second call
- Automatic call

You can also set these parameters via SNMP, as described in “Chapter 24 - Subscriber Services” on page 333.

## Call Transfer

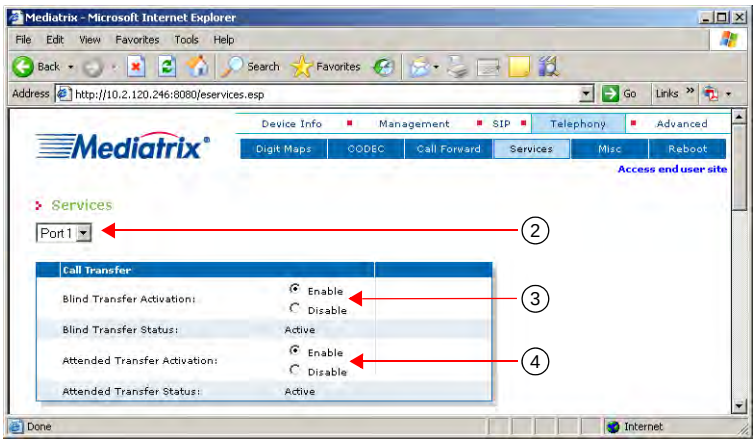
The Call Transfer service offers two ways to transfer calls:

- Blind Transfer
- Attended Transfer

▶ **To enable the Call Transfer services:**

1. In the web interface, click the *Telephony* link, then the *Services* sub-link.

**Figure 57:** Telephony – Call Transfer Web Page




2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Call Transfer* section, enable the Blind Transfer service by selecting **Enable** in the *Blind Transfer Activation* choice.

   The blind call transfer service is sometimes called Transfer without Consultation or Unattended Transfer. It allows a user to transfer a call on hold to a still ringing (unanswered) call. The individual at the other extension or telephone number does not need to answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See "Call Hold" on page 115 and "Second Call" on page 116.

4. Enable the Attended Transfer service by selecting **Enable** in the *Attended Transfer Activation* choice.

   The attended call transfer service is sometimes called Transfer with Consultation. It allows a user to transfer a call on hold to an active call. The individual at the other extension or telephone number must answer to complete the transfer.

   The call hold and second call services must be enabled for this service to work. See "Call Hold" on page 115 and "Second Call" on page 116.

5. Click *Submit* if you do not need to set other parameters.

### Using Blind Call Transfer

The following is the procedure to use this service on the user's telephone.

▶ **To transfer a current call blind:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
   This puts the call on hold.
2. Wait for the transfer tone (three "beeps").
3. Dial the number to which you want to transfer the call.
4. Wait for the ringback tone, then hang up your telephone.
   The call is transferred. You can also wait for the third party to answer if you want. In this case, the call transfer becomes attended.
   If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.
   You are back with the first call and the third party is released.

### Using Attended Call Transfer

The following is the procedure to use this service on the user's telephone.

▶ **To transfer a current call attended:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
   This puts the call on hold.
2. Wait for the transfer tone (three "beeps").
3. Dial the number to which you want to transfer the call.
   The third party answers.
4. Hang up your telephone.
   The call is transferred.
5. If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.
   You are back with the first call and the third party is released.

**Note:** If the number to which you want to transfer the call is busy or does not answer, quickly perform a Flash-Hook. The busy tone or ring tone is cancelled and you are back with the first call.

## Call Waiting

The call waiting tone indicates to an already active call that a new call is waiting on the second line.

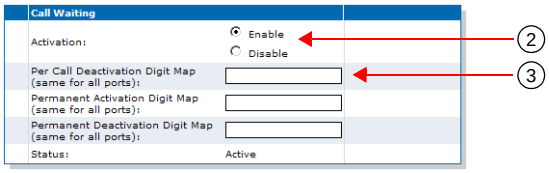
Your users can activate/deactivate the call waiting tone for their current call. This is especially useful when transmitting faxes. The user that is about to send a fax can thus deactivate the call waiting tone to ensure that the fax transmission will not be disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

Your users can also permanently activate/deactivate the call waiting service.

### ▶ To set the Call Waiting services:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Waiting* section, enable the service by selecting **Enable** in the *Activation* choice.

**Figure 58:** Call Waiting Section




This permanently activates the call waiting tone. When receiving new calls during an already active call, a special tone is heard to indicate that a call is waiting on the second line. The user can then answer that call by using the “flash” button. The user can switch between the two active calls by using the “flash” button.

The call hold service must be enabled for this service to work. See “Call Hold” on page 115.

If the user is exclusively using faxes, select **Disable** to permanently disable the call waiting tone.

The user may cancel this service on a per-call basis when dialing a DTMF sequence matching the digit map stored in the *Cancel Digit Map* field (see Step 3). The user may also disable or enable this service permanently with the *Permanent Activation Digit Map* and *Permanent Deactivation Digit Map* fields (See Step 4).

3. Define the digits that users must dial to disable the Call Waiting tone in the *Per Call Deactivation Digit Map* field.

   This allows a user who has call waiting enabled to disable that service on the next call only. If, for any reason, the user wishes to undo the cancel, simply unhook and re-hook the telephone to reset the service.

   For instance, you could decide to put “*76” as the sequence to disable the call waiting tone. This sequence must be unique and follow the syntax for digit maps (see “Digit Maps” on page 87). Dialing this digit map does not have any effect unless the service’s status is “enabled”.

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

4. Define the digits that users must dial to enable the call waiting service permanently in the *Permanent Activation Digit Map* field.

   This activation is permanent until the user deactivates the service as in Step 5.

   For instance, you could decide to put “*84” as the sequence to enable the service. This sequence must be unique and follow the syntax for digit maps (see “Chapter 22 - Digit Maps” on page 321).

   The sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

   When dialing this digit map, this sets the *Activation* frop-down menu for the line the user is currently using to **enable**.

5. Define the digits that users must dial to disable the call waiting service permanently in the *Permanent Deactivation Digit Map* field.

   This deactivation is permanent until the user enables the service as in Step 4.

   For instance, you could decide to put “*85” as the sequence to disable the service. This sequence must be unique and follow the syntax for digit maps (see “Chapter 22 - Digit Maps” on page 321).

   The sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

   When dialing this digit map, this sets the *Activation* drop-down menu for the line the user is currently using to **disable**.

6. Click *Submit* if you do not need to set other parameters.

## Using Call Waiting

The call waiting feature alerts the user if he or she is already on the telephone and a second call happens. A “beep” (the call waiting tone) is heard and repeated every ten seconds to indicate there is a second incoming call.

**To put the current call on hold:**

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
   This puts the call on hold and the second line is automatically connected to your line.
2. Answer the call on the second line.

**To switch from one line to the other:**

1. Perform a Flash-Hook each time you want to switch between lines.

**To terminate the first call before answering the second call:**

1. Hang up the telephone.
2. Wait for the telephone to ring.
3. Answer the telephone.
   The second call is on the line.

## Removing the Call Waiting Tone

You can temporarily activate/deactivate the call waiting tone indicating a call is waiting. This is especially useful when transmitting faxes. If you are about to send a fax, you can thus deactivate the call waiting tone to ensure that the fax transmission is not disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

**To deactivate the call waiting tone:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call waiting tone.
   This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
   The call waiting tone is disabled.

**To re-enable the call waiting tone:**

1. Take the receiver off-hook.
2. Replace the receiver on-hook.
   The call waiting tone is re-enabled.

## Permanently Removing the Call Waiting Tone

You can permanently activate/deactivate the call waiting service.

**To activate the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call waiting tone service.
   This sequence could be something like *84.

4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Hang up your telephone.
   The call waiting tone is enabled.

▶ **To cancel the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone service.
   This sequence could be something like *85.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
   The call waiting is cancelled.
5. Hang up your telephone.

## Conference

The Conference Call service allows a user to link two or more calls together to form a single conversation, called a conference.

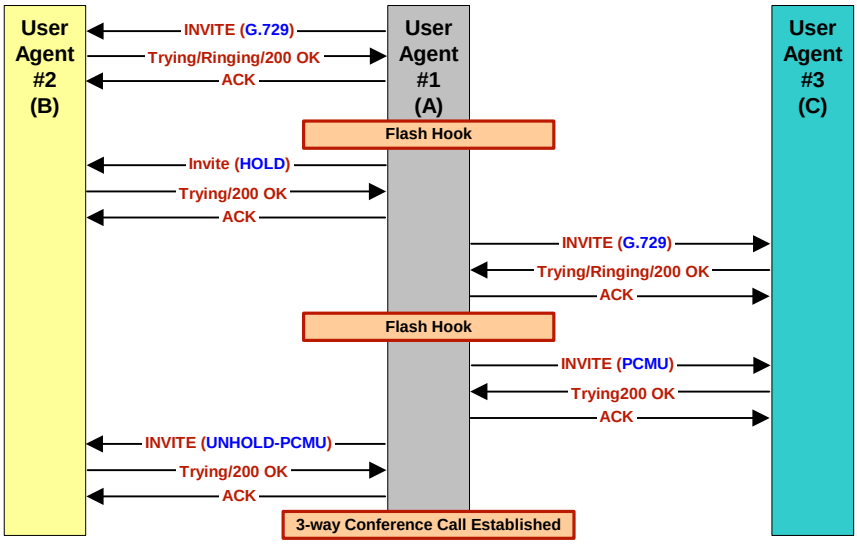
A participant of the conference can put the conference on hold and attempt other calls. This participant may then rejoin the conference at a later time by unholding it. The participant who initiated the conference cannot put it on hold.

You must enable the call hold and second call services for this service to work. See “Call Hold” on page 115 and “Second Call” on page 116.

Furthermore, you must also enable the Attended Transfer service for the two other participants to stay connected once the participant who initiated the conference has hung up. See “Call Transfer” on page 109.

The following is a conference call flow example:

**Figure 59:** Conference Call Flow




### Requirements

For the conference call to occur successfully, all parties must meet the following requirements:

- Support at least one of the PCM codecs (G.711 µ-law and G.711 A-law) enabled on the line that is having the conference. See “Voice & Fax Codecs” on page 93 for more details.
- Ability to dynamically change codec during a call.

- The packetization period (ptime) should be the same for all the participants of the conference. If this is not the case, then part of the conversation may be lost, resulting in a choppy voice. For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See "Voice & Fax Codecs" on page 93 for more information on how to set the packetization period of the Mediatrix 4102.

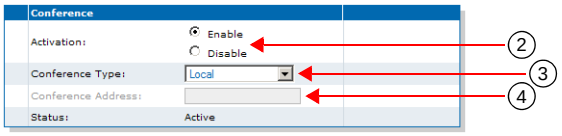
## Enabling the Conference Call Feature

You must enable this service before your users can use it.

**To enable the Conference service:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Conference* section, enable the service by selecting **Enable** in the *Activation* choice.

**Figure 60:** Conference Section




3. Specify how to manage a SIP conference in the *Conference Type* drop-down menu.
   This configuration only applies to a conference initiated by one of the unit's endpoint.

**Table 72:** Conference Type Parameters

| Parameter | Description |
|---|---|
| Local | The conference is managed locally by the unit. The maximum number of participants is 3. This is the default value. |
| Conference Server | The conference is managed by a remote SIP conference server.<br>When using this conference type, both the initiator and a participant of the conference can add new participants to the conference. |

4. If you have selected **Conference Server** in the previous step, set the *Conference Address* field with the address of a conference server.
   The format must be a SIP URI such as "scheme:user@host". For instance, "sip:user@foo.com".
5. Click *Submit* if you do not need to set other parameters.

## Managing a Conference Call

If you are on the telephone with one person and want to conference with a third one, you can do so. In the following examples, let's assume that:

- "A" is the conference initiator.
- "B" is the person called on the first line.
- "C" is the person called on the second line.
- "D" is a fourth person that "A" wants to add to the conference in **Conference Server** conference type.

**To initiate a conference ("A" and "B" already connected):**

1. "A" performs a Flash-Hook.
   This puts "B" on hold and the second line is automatically connected. "A" hears a dial tone.
2. "A" dials "C's" number.
   "A" and "C" are now connected.
3. "A" performs another Flash-Hook.

The call on hold (“B”) is reactivated. “A” is now conferencing with “B” and “C”.

▶ **“A” wants to transfer “B” to “C” during the conference:**

This is available only in the **Local** conference type.

1. “A” hangs up.

   The conference is terminated. “B” and “C” are now connected.

▶ **“A” wants to terminate the call with “C” and get back to the call with “B” during the conference:**

This is available only in the **Local** conference type.

1. “A” performs a Flash-Hook.

   The conference is terminated and the call with “C” is disconnected. “A” and “B” are still connected and can go on with their conversation.

▶ **“B” (or “C”) hangs up during the conference:**

This is available only in the **Local** conference type.

1. “B” (or “C”) hangs up during the conference.

   The conference is terminated, but the call between “A” and “C” (or “B”) is not affected and they are still connected.

▶ **“A” wants to add a fourth member to the conference:**

This is available only in the **Converence Server** conference type.

1. “A” performs a Flash-Hook.

   This puts “B” and “C” on hold and the second line is automatically connected. “A” hears a dial tone.

2. “A” dials “D’s” number.

   “A” and “D” are now connected.

3. “A” performs another Flash-Hook.

   The call on hold (“B” and “C”) is reactivated. “A” is now conferencing with “B”, “C”, and “D”.

## Call Hold

The Call Hold service allows the user to temporarily put an existing call on hold, usually by using the “flash” button of the telephone. The user can resume the call in the same way.

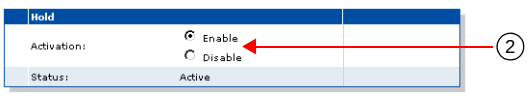
You must enable this service for the following services to work properly:

- Call Waiting
- Second Call
- Blind Transfer
- Attended Transfer
- Conference

▶ **To enable the Call Hold service:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Hold* section, enable the service by selecting **Enable** in the *Activation* choice.

**Figure 61:** Hold Section




3. Click *Submit* if you do not need to set other parameters.

### Using Call Hold

The following is the procedure to use this service on the user's telephone.

▶ **To put the current call on hold:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
   This puts the call on hold. You can resume the call in the same way.

## Second Call

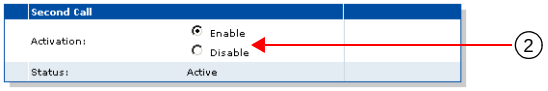
The Second Call service allows a user with an active call to put the call on hold, and then initiate a new call on a second line. This service is most useful with the transfer and conference services.

The call hold service must be enabled for this service to work. See "Call Hold" on page 115.

▶ **To enable the Second Call service:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Second Call* section, enable the service by selecting **Enable** in the *Activation* choice.

**Figure 62:** Second Call Section




3. Click *Submit* if you do not need to set other parameters.

### Using Second Call

The following is the procedure to use this service on the user's telephone.

▶ **To use the second call service:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
   This puts the call on hold and the second line is automatically connected to your line.
2. Initiate the second call.

## Automatic Call

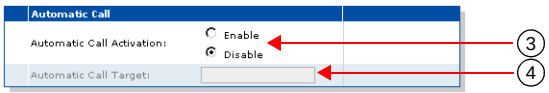
The automatic call feature allows you to define a telephone number that is automatically dialed when taking the handset off hook.

When this service is enabled, the second line service is disabled but the call waiting feature is still functional. The user can still accept incoming calls.

▶ **To set the automatic call feature:**

1. In the web interface, click the *Telephony* link, then the *Services* sub-link.
   This links to the *Telephony – Advanced* web page.

**Figure 63:** Telephony – Advanced Web Page




2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Automatic Call* section, enable the service by selecting **Enable** in the *Automatic Call Activation* choice.

4. Define the number to dial when the handset is taken off hook in the *Automatic Call Target* field.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

5. Click *Submit* if you do not need to set other parameters.

# Miscellaneous

The *Misc* sub-page of the *Telephony* page allows you to set the following parameters:

- Country
- Custom Tone Configuration
- Message Waiting Indicator

## Country Selection

It is very important to set the country in which the Mediatrix 4102 is used because a number of parameter values are set according to this choice. These parameters are:

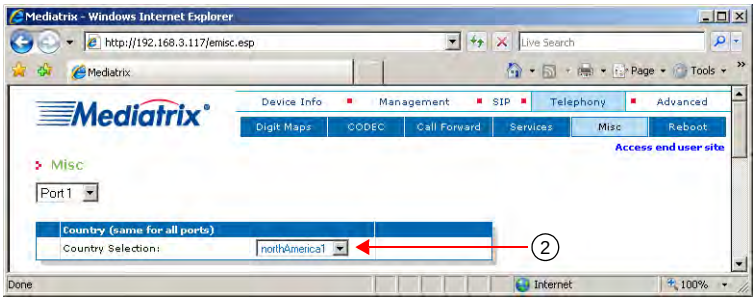
- Tones
- Rings
- Impedances
- Line Attenuations

See “Appendix D - Country-Specific Parameters” on page 411 for more information on these country-specific settings. You can also set these parameters via SNMP, as described in “Setting the Location (Country)” on page 190.

**To set a country location:**

1. In the web interface, click the *Telephony* link, then the *Misc* sub-link.

**Figure 64:** Telephony – Misc Web Page




2. In the *Country* section, select the country in which the Mediatrix 4102 is located in the *Country Selection* field.

3. Click *Submit* if you do not need to set other parameters.

   This parameter is set for all the lines of the Mediatrix 4102. You cannot have a different country for each line.

### Caller ID Selection

In countries that support more than one caller ID standard, this standard can be selected with the *Country Selection* field. Be careful to properly select the option corresponding to your caller ID.

**Table 73:** Caller ID Mappings

| Country | Caller ID | *Country Selection* field Mapping |
|---|---|---|
| UK | British Telecom | uk |
| | Bellcore | uk-bellcore |
| | CCA | uk-cca |
| | ETSI-FSK | uk-etsi-fsk |
| France | Bellcore | france |
| | ETSI-FSK | france-etsi-fsk |
| | ETSI-DTMF | france-etsi-dtmf |
| Austria1 | Bellcore | austria1 |
| | ETSI-FSK | austria-etsi-fsk |
| Austria2 | Bellcore | austria2 |
| | ETSI-FSK | austria2-etsi-fsk |

See "Caller ID Information" on page 189 for more details.

## Custom Tone Configuration

You can override the pattern for a specific tone defined for the selected country (see "Appendix D - Country-Specific Parameters" on page 411 for more details). You can define new patterns for the following tones:

- Busy
- Confirmation
- Congestion
- Dial
- Intercept
- Message Waiting
- Preemption
- Reorder
- Ringback
- Receiver Off Hook (ROH)
- Special Information Tone (SIT)
- Stutter

## Pattern Definition

The general format of the pattern string is defined in the following ABNF:

```
tone-pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

This general pattern uses the following three main categories

```
frequencies-section = "f1=" frequency-description
                      [ ",f2=" frequency-description
                        [ ",f3=" frequency-description
                          [ ",f4=" frequency-description ] ] ]

loop-counter-section = "l=" loop-counter

states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description
                     [ ",s4=" state-description
                       [ ",s5=" state-description
                         [ ",s6=" state-description
                           [ ",s7=" state-description
                             [ ",s8=" state-description ] ] ] ] ] ] ]
```

Finally, the three main categories use the following parameters and tags:

```
frequency-description = frequency ":" power
frequency = 2*4DIGIT
power = ( DIGIT / "-" 1*2DIGIT )
loop-counter = 1*2DIGIT
state-description = ( on-state-description / off-state-description )
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [ next-state
]
off-state-description = "off" [ time ][ loop-indicator ][ next-state]
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
time = ":" 2*5DIGIT
loop-indicator = ":l"
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

The following table describes the various tags used in the syntax.

**Table 74:** Pattern Definition Syntax

| Tag | Description |
|---|---|
| `tone-pattern` | String describing the pattern to use for the tone. An empty string means no tone. |
| `frequencies-section` | Description of the frequencies used by the tones used in `states-section`. You can define up to four frequencies (f1 to f4). You must enter at least one frequency if the `tone-pattern` is not empty. The frequencies to use are defined in the `state-description`. |
| `frequency-description` | Description of the frequency. This is described as "`frequency`:`power`". |
| `frequency` | Frequency value in Hz. The range is from 10 Hz to 4000 Hz. |
| `power` | Power level of the frequency in dBm. The range is from -99 dBm to 3 dBm. |
| `loop-counter-section` | Loop counters definition. The loop counter is used in `state-description`. |
| `loop-counter` | Value of the loop counter. The range is from 2 to 128. |
| `states-section` | Description of the tone state. You can define up to eight states (s1 to s8). You must enter at least one state if the `tone-pattern` is not empty. |
| `state-description` | Description of the tone state. |

**Table 74:** Pattern Definition Syntax (Continued)

| Tag | Description |
|---|---|
| `on-state-description` | Description of a state playing a tone. |
| `off-state-description` | Description of a state not playing a tone. |
| `frequency-selection` | Frequency to play in the state. You can use from one to four frequencies. The frequency must be defined in `frequencies-section`. |
| `time` | The number of times, in ms, to perform the action of the state. The range is from 10 ms to 56000 ms. The tone stays indefinitely in the state if no time is specified. |
| `loop-indicator` | Used to stop looping between states after a number of loops defined in `loop-counter-section`. When the number of loops is reached, the next state is s(n+1) for the state s(n) instead of the state defined in `next-state`. |
| `next-state` | The next tone state to use when the time has elapsed. This value is not present if the time is not present. |

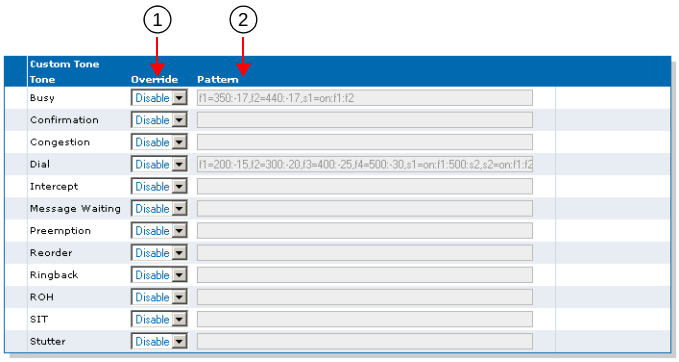
## Customizing the Tones

The *Custom Tone* section allows you to define new patterns as per the pattern syntax.

▶ **To customize one or more tones:**

1. In the *Custom Tone* section, of the *Misc* page, define whether or not you want to override the default tone configuration for a specific tone by setting the corresponding *Override* column.

**Figure 65:** Custom Tone Section




**Note:** The default hold tone value is mute (i.e., no tone).

2. Enter the override pattern in the corresponding *Pattern* column.

   You must follow the syntax described in "Pattern Definition" on page 119.

   See "Custom Tone Example" on page 121 for a detailed example on how to create a proper pattern.

   The following table gives some examples of custom tones. Note that the quotation marks are not part of the syntax and must not be included when entering the tone pattern.

**Table 75:** Pattern Examples

| Example | Pattern |
|---|---|
| No tone | "" |

**Table 75:** Pattern Examples (Continued)

| Example | Pattern |
|---|---|
| North America dial tone (continuous tone at 350 Hz and 440 Hz with a -17 dBm power level) | `"f1=350:-17,f2=440:-17,s1=on:f1:f2"` |
| North America Recall dial tone (three quick tones followed by a continuous tone) | `"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"` |
| Australia ring back tone (tone on 400 ms, off 200 ms, on 400 ms, and off 2000 ms and replay) | `"f1=425:-17,f2=400:-5,f3=450:-5,s1=on:f1:f2:f3:400:s2,s2=off:200:s3,s3=on:f1:f2:f3:400:s4,s4=off:2000:s1"` |

3. Click *Submit* if you do not need to set other parameters.

## Custom Tone Example

This section describes how to create the pattern for the North America recall dial tone (also called stutter dial tone), which is three quick tones followed by a continuous tone.

```
"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
```

**To create the pattern:**

1. Let's start with the general format of the pattern string:

```
Pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

2. Set the `frequencies-section` category, which is defined as follows:

```
frequencies-section = "f1=" frequency-description
                      [ ",f2=" frequency-description
                        [ ",f3=" frequency-description
                          [ ",f4=" frequency-description ] ] ]
```

*a.* The `frequency-description` parameter is described as follows:

```
frequency:power
```

*b.* In the North America stutter dial tone, two frequencies are used, 350 Hz and 440 Hz. Their power level is -17 dBm. You can thus complete the `frequencies-section` category as follows:

```
frequencies-section = "f1=" 350:-17 ",f2=" 440:-17 ]
```

*c.* The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," loop-counter-section ] "," states-section ]
```

3. Set the `loop-counter-section` category, which is defined as:

```
loop-counter-section = "l=" loop-counter
```

It defines the number of times to repeat the pattern.

*a.* The loop-counter part is defined as follows:

```
loop-counter = 1*2DIGIT
```

*b.* In the North America stutter dial tone, the pattern is repeated three times, thus:

```
loop-counter = 3
```

*c.* The `loop-counter-section` category now looks as follows:

```
loop-counter-section = "l=" 3
```

*d.* The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," states-section ]
```

**4.** Set the `states-section` category, which is defined as:

```
states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description
                     [ ",s4=" state-description
                       [ ",s5=" state-description
                         [ ",s6=" state-description
                           [ ",s7=" state-description
                             [ ",s8=" state-description ] ] ] ] ] ] ]
```

*a.* `state-description` is defined as:

```
state-description = ( on-state-description / off-state-description )
```

*b.* There are three states in the North America stutter dial tone: 0.1 on, 0.1 off, and continuous. The pattern that must be described is thus:

```
states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

**5.** Let's define the first state. Since the first state describes an on tone, `off-state-description` is not required for this state.

*a.* You now have to complete the `on-state-description` parameter for the first state, which is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

*b.* `frequency-selection` is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

*c.* The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] ] [ time] [ loop-indicator ]
[ next-state ]
```

*d.* The `time` parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The on state is 100 ms, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ loop-indicator ] [
next-state ]
```

*e.* The `loop-indicator` parameter is not used in this state. The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ next-state ]
```

*f.* The `next-state` parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the `next-state` parameter is the off state, which is designated as `s2`.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ]
```

*g.* You can now complete the first `state-description` parameter:

```
states-section = ",s1=" ( on-state-description)
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

**6.** Let's define the second state. Since the first state describes an off tone, `on-state-description` is not required for this state.

*a.* You now have to complete the `off-state-description` parameter for the first state.

The `off-state-description` parameter is defined as:

```
off-state-description = "off" [ time ][ loop-indicator ][ next-state]
```

*b.* The `time` parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The off state is 100 ms, thus,

```
off-state-description = "off" [ :100 ][ loop-indicator ][ next-state]
```

*c.* The `loop-indicator` parameter is defined as:

```
loop-indicator = ":l"
```

It is used to stop looping between states. It indicates that the loop stops after three times. Once the loop is completed, the pattern goes to the next state (which is state 3).

The `off-state-description` parameter is now:

```
off-state-description = "off" [ :100 ] [ :1 ][ next-state]
```

*d.* The `next-state` parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the `next-state` parameter is the on state, which is designated as `s1`.

The `off-state-description` parameter is now:

```
off-state-description = "off" [ :100 ] [: 1 ][ :s1]
```

*e.* You can now complete the second `state-description` parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                  [ ",s2=" state-description
                    [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [100] [ s2 ])
                  [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                    [ ",s3=" state-description ] ]
```

**7.** Let's define the third and last state. Since the third state describes an on tone, `off-state-description` is not required for this state.

*a.* You now have to complete the `on-state-description` parameter for the first state.

The `on-state-description` parameter is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

*b.* `frequency-selection` is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

*c.* The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ time] [ loop-indicator ] [
next-state ]
```

*d.* The `time` parameter is the number of milliseconds to perform the action of the state. Since the third state is a continuous tone, the `time` parameter is not required, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ loop-indicator ] [ next-
state ]
```

*e.* The `loop-indicator` parameter is used to stop looping between states. Since the third state is a continuous tone and does not use loops, this parameter is not required.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ next-state ]
```

*f.* The `next-state` parameter is the next tone state to use when the time has elapsed. This value is not present if the `time` parameter is not present. You have already discarded the `time` parameter, so the `next-state` parameter is not required.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ]
```

*g.* You can now complete the third `state-description` parameter and the `states-section` parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ]
```

**8.** Now that you have the three main categories completed, you can finish the pattern:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," ",s1=" ( "on" [ ":f1" ]
[ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ] ]
```

If you remove all the brackets and quotation marks, which are not to be included, the pattern is:

```
Pattern = f1=350:-17,f2=440:-
17,l=3,s1=on:f1:f2:100:s2,s2=off:100:1:s1,s3=on:f1:f2
```

Ther pattern could also be defined as follows:

```
Pattern = f1=350:-17,f2=440:-
17,s1=on:f1:f2:100:s2,s2=off:100:s3,s3=on:f1:f2:100:s4,s4=off:100:s5,s5=on:f
1:f2:100:s6,s6=off:100:s7,s7=on:f1:f2
```

## Message Waiting Indicator

The Message Waiting Indicator (MWI) service alerts the user when new messages have been recorded on a voice mailbox.

When the user receives a call and does not answer, the notification mechanism detects this situation and starts the auto attendant. The caller can then leave a message.

After the message is recorded, the server sends a message to the Mediatrix 4102 listing how many new and old messages are available. The Mediatrix 4102 alerts the user of the new message in two different ways:

- The telephone's LED blinks (if present).
- A message waiting stutter dial tone replaces the normal dial tone when the user picks up the first line.

**Note:** The message waiting state does not affect the Second Line feature. When in an active call, performing a flash-hook to get access to the second line plays the usual dial tone.

### Standard MWI Methods

The Mediatrix 4102 supports two MWI methods.

*MWI Method #1*

| | |
|---|---|
| **Standards Supported** | • draft-ietf-sipping-mwi-01.txt (MWI draft)<br>• "Telecordia GR-1401-CORE (Issue 1, June 2000)" specification (visual message indication (LED blinking)<br>• "GR-506-CORE (Issue 1, with Revision 1, November 1996)" specification (message waiting indicator tone) |

The Mediatrix 4102 sends SUBSCRIBE requests to the server for each line, unless there is no subscription address defined. The Mediatrix 4102 then waits for NOTIFY requests containing the relevant message waiting information.

You can also set these parameters via SNMP, as described in "Chapter 26 - Message Waiting Indicator" on page 359.

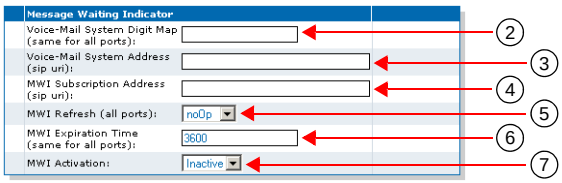
**To configure the MWI service:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Message Waiting Indicator* section, define the digits that users must dial to retrieve messages in the *Voice-Mail System Digit Map* field.

   Dialing these digits initiates a call to the voice messaging system. For instance, you could decide to put "*50" as the sequence a user must dial to retrieve voice messages. This sequence must be unique and follow the syntax for digit maps (see "Digit Maps" on page 87). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have different sequences for each line.

**Figure 66:** Message Waiting Indicator Section




3. Set the destination to call to retrieve messages in the *Voice-Mail System Address* field.

   The user typically initiates a call to the voice messaging system, and then uses an auto-attendant to get the messages. Available formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

4. Set the notification mechanism server address to which the Mediatrix 4102 subscribes in the *MWI Subscription Address* field.

   This mechanism notifies the Mediatrix 4102 when new messages are available. The address is a SIP URL such as "scheme:user@host". For instance, "sip:user@foo.com".

5. Set the subscription refresh in the *MWI Refresh* field.

   Available values are:

   - noOp: No operation.
   - refresh: Refresh message waiting subscriptions. All enabled endpoints unsubscribe themselves from the service and re-subscribe by using the current provisioning.

   This parameter is set for all the lines of the Mediatrix 4102. You cannot have a different behaviour for each line.

6. Define the duration, in seconds, of dynamic subscription to a messaging service in the *MWI Expiration Time* field.

   This parameter is set for all the lines of the Mediatrix 4102. You cannot have a different behaviour for each line.

7. Enable the MWI by selecting **Active** in the *MWI Activation* field.

   The MWI subscription refresh is not supported when the caller ID is DTMF-based.

8. Click *Submit* if you do not need to set other parameters.

### *MWI Method #2*

| | |
|---|---|
| **Standards Supported** | draft-mahy-sip-message-waiting-02.txt (expired) with proprietary modifications |

This method does not require any special settings or configuration.

## MWI Notify Service

The Mediatrix 4102 offers the possibility to extend some key features to remote extensions located in Branch or Home Offices across the SCN.

This service is available only when using the IP Communication Server v3.1 product as a SIP Redirect server.

For instance, a designated analog voice mail system at a main site can provide voice mail for the home or branch office. The home office user is notified of the message waiting via a message waiting LED on the telephone or a special tone when picking up the telephone.

### *How does the Service Work?*

The MWI Notify service is a proprietary feature. In this solution, the analog voice mail system is configured to seize a designated outgoing line and dial a pre-defined string such as “*72xxx” to notify the server it must give a message waiting indication to extension “xxx”. Once voice messages have been retrieved, the analog voice mail system seizes the designated outgoing line and dials a pre-defined string such as “*73xxx” to notify the server to turn off the message waiting indicator for extension “xxx”.

The service uses the Route Manager currently available in the IP Communication Server v3.1 to send a special command to the Mediatrix unit.

The following is the basic sequence of operations for the MWI Notify service:

1. The analog voice mail system dials the following digits:
   `*72101`

   where `*72` is a prefix and `101` the user extension.

2. The Mediatrix unit sends a standard INVITE to the IP Communication Server v3.1 containing the complete dialed string (`*72101`).

   *a.* The IP Communication Server looks for the registered user “*72101” in the Registrar database.

   *b.* The IP Communication Server cannot find the user, so it asks the Route Manager to process the request.

   *c.* Provided that the Route Manager is properly configured, it recognizes the “*72” prefix and associates it to the proper route conditions.

3. The IP Communication Server answers the request with a “Moved Temporarily”. It contains information about the target(s) in the *Contact* header plus a proprietary *p-MxBlindMWINotify=yes/no* field.

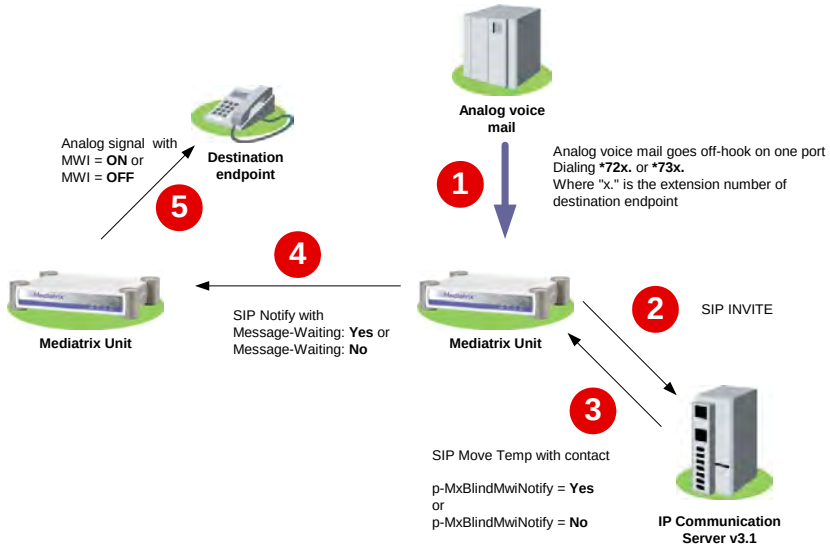
   *a.* The Mediatrix unit retrieves the location from the IP Communication Server’s answer and the *p-MxBlindMWINotify* field.

   *b.* The Mediatrix unit parses the answer from the IP Communication Server and recognizes *p-MxBlindMWINotify* as a special command.

4. The Mediatrix unit sends a NOTIFY to the location received from the IP Communication Server by using the proper yes or no value (*72 = yes, *73 = no) specified by the route condition.

**5.** The unit receiving the NOTIFY enables or disables the MWI service for the specified port/user.

**Figure 67:** Example of the MWI Notify Service




### *Configuring the IP Communication Server*

In the Route Manager of the IP Communication Server, you must configure routes that would be triggered by a pre-defined prefix. The prefix could be any valid digits (DTMF). The example described above uses “*72” to enable the MWI and “*73” to disable the MWI.

For more information on how to configure the Route Manager, please refer to the *IP Communication Server Administration Manual* or the IP Communication Server contextual help.

### *Configuring the Mediatrix 4102*

There is no special unit configuration required. The Mediatrix unit behaves as if in a standard call until it receives one of the following parameters in the *Contact* field:

- p-MxBlindMwiNotify=Yes

  or
- p-MxBlindMwiNotify=No

Upon receiving one of these parameters, the unit sends a NOTIFY to the destination endpoint instead of an INVITE. The sent NOTIFY is compliant with <draft-mahy-sip-message-waiting-02.txt>.

CHAPTER 7

# Web Interface – Advanced

The *Advanced* page allows you to configure various system and network parameters of the Mediatrix 4102.

## Quality of Service (QoS)

QoS (Quality of Service) features enable network managers to decide on packet priority queuing. The Mediatrix 4102 supports the Differentiated Services (DS) field and 802.1q taggings. The Mediatrix 4102 supports the Real Time Control Protocol (RTCP), which is used to send packets to convey feedback on quality of data delivery.

The Mediatrix 4102 does not support RSVP (Resource Reservation Protocol).

### 802.1q Configuration

The 802.1q standard recommends the use of the 802.1q VLAN tags for Ethernet frames traffic prioritization. VLAN tags are 4-byte headers in which three bits are reserved for priority indication. The values of the priority bits shall be provisioned.

The 802.1q standard comprises the 802.1p standard.

You can also set these parameters via SNMP, as described in "Chapter 28 - Quality of Service (QoS)" on page 365.

#### VLANs

VLANs are created with standard Layer 2 Ethernet. A VLAN Identifier (VID) is associated with each VLAN. VLANs offer the following benefits:

- VLANs are supported over all IEEE 802 LAN MAC protocols, and over shared media LANs as well as point-to-point LANs.
- VLANs facilitate easy administration of logical groups of stations that can communicate as if they were on the same LAN. They also facilitate easier administration of moves, adds, and changes in members of these groups.
- Traffic between VLANs is restricted. Bridges forward unicast, multicast, and broadcast traffic only on LAN segments that serve the VLAN to which the traffic belongs.

The VLAN field in the Ethernet file is located after both destination and source addresses:

```
    0                   1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7   (byte)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Dest Addr | Src Addr  | VLAN  | Type/Length | ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The VLAN field is separated as follows:

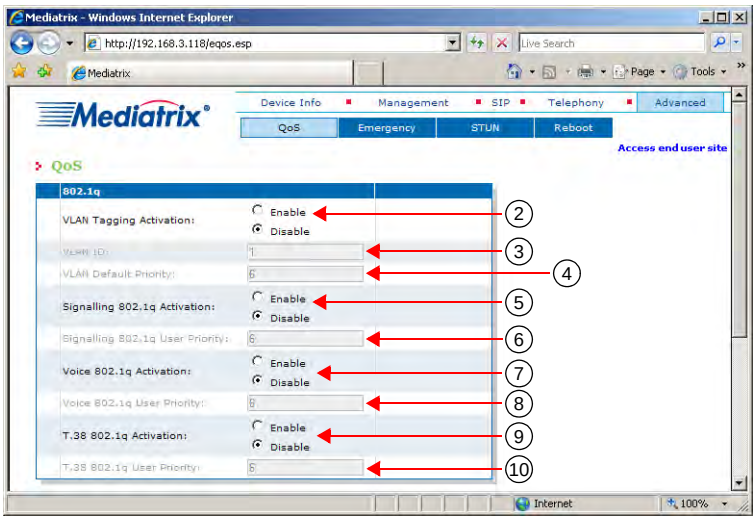
```
   0  (bit)           1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |             0x8100            | Pri |T|          VID          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

For both signalling and media packets, the VLAN priority section is configurable independently.

### ▶ To enable the IEEE 802.1q user priority configuration:

1. In the web interface, click the *Advanced* link, then the *QoS* sub-link.

**Figure 68:** Advanced – QoS Web Page




2. Enable the VLAN tagging by selecting **Enable** in the *VLAN Tagging Activation* field.

   All packets are tagged with the Virtual ID (VID) specified in the *VLAN ID* field.

   Enable this option only on compatible LAN with equipment supporting the VLAN tagging. Otherwise, the Mediatrix 4102 may be unreachable. In this case, use the *Reset / Default* button to access and disable VLAN tagging – in the recovery mode, tagging is not permitted.

3. Set the 802.1Q Virtual LAN ID in the *VLAN ID* field.

   This is the VID to be applied in the TCI field when tagging is enabled. The value 1 is the default Port VID (PVID) for bridge port. The 4095 VID (0xFFF) is reserved and it must not be used in tag header.

   When both VLAN tagging and VLAN Substitution are enabled and their VLAN ID is the same, VLAN Tagging has precedence over VLAN Substitution. If VLAN Substitution has the same ID as VLAN Tagging, VLAN Substitution is not enabled and the Mediatrix 4102 behaves as such. You should change the ID of one of the features to enable VLAN Substitution. See "VLAN Substitution" on page 369 for more details.

4. Set the 802.1Q Virtual LAN default user priority in the *VLAN Default Priority* field.

   This is the user priority to be applied in the TCI field when tagging is enabled. This value applies to all protocols for which no priority filtering is enabled (e.g. ARP, ICMP).

   - 7 = High priority
   - 0 = Low priority

5. Enable the 802.1Q VLAN user priority tagging for VoIP signalling packet by selecting **Enable** in the *Signalling 802.1q Activation* field.

   This filter applies to any VoIP signalling protocol in use. Each signalling packet is tagged with the user priority defined in the *Signalling 802.1q User Priority* field.

6. Set the IEEE 802.1Q VLAN user priority value for VoIP signalling packet in the *Signalling 802.1q User Priority* field.

   - 7 = High priority
   - 0 = Low priority

7. Enable 802.1Q VLAN user priority tagging for VoIP packet by selecting **Enable** in the *Voice 802.1q Activation* field.

   This filter applies to any VoIP voice protocol in use (e.g. RTP). Each signalling packet is tagged with the user priority defined in the *Voice 802.1q User Priority* field.

8. Set the 802.1Q VLAN user priority value for VoIP packet in the *Voice 802.1q User Priority* field.
   - 7 = High priority
   - 0 = Low priority
9. Enable 802.1Q VLAN user priority tagging for T.38 fax packet by selecting **Enable** in the *T.38 802.1q Activation* field.

   Each signalling packet is tagged with the user priority defined in the *T.38 802.1q User Priority* field.
10. Set the 802.1Q VLAN user priority value for T38 fax packet in the *T.38 802.1q User Priority* field.
    - 7 = High priority
    - 0 = Low priority
11. Click *Submit* if you do not need to set other parameters.

### VLAN User Priority

The VLAN user priority values are used to set the user priority in the TCI field of the VLAN tag. Tagging user priority is applied only when the filter is enabled. When the filter for signalling protocol is disabled and the VLAN option is enabled, the Mediatrix 4102 uses the default user priority defined in the *VLAN Default Priority* field. Otherwise, the user priority set for signalling has precedence over the VLAN default user priority.

## DiffServ Configuration

| **Standards Supported** | RFC 2475 – An Architecture for Differentiated Services |
|---|---|

Differentiated Services (DiffServ, or DS) is a protocol for specifying and controlling network traffic by class so that certain types of traffic – for example, voice traffic, which requires a relatively uninterrupted flow of data, might get precedence over other kinds of traffic.

DiffServ replaces the first bits in the ToS byte with a differentiated services code point (DSCP). It uses the existing IPv4 Type of Service octet.

▶ **To set the DiffServ configuration:**

1. In the *DiffServ* section of the *QoS* page, set the following DiffServ fields:
   - Voice DiffServ Value:
   - T.38 DiffServ Value

### What are Differentiated Services?

Differentiated Services avoids simple priority tagging and depends on more complex policy or rule statements to determine how to forward a given network packet. An analogy is made to travel services, in which a person can choose among different modes of travel – train, bus, airplane – degree of comfort, the number of stops on the route, standby status, the time of day or period of year for the trip, and so forth.

For a given set of packet travel rules, a packet is given one of 64 possible forwarding behaviors – known as per hop behaviors (PHBs). A six-bit field, known as the Differentiated Services Code Point (DSCP), in the Internet Protocol header specifies the per hop behavior for a given flow of packets. The DS field structure is presented below:

```
0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| DSCP                  |  CU   |
+---+---+---+---+---+---+---+---+
MSB                           LSB
```

- *DSCP*: Differentiated Services CodePoint.
- *CU*: Currently Unused. The CU bits should always be set to 0.

For both signalling and media packets, the DSCP field is configurable independently. The entire DS field (TOS byte) is currently configurable.

- Signalling DiffServ Value

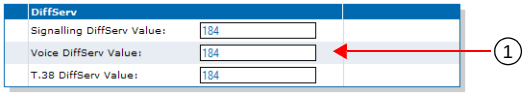
These fields are 1 octet scalar ranging from 0 to 255. The DSCP default value should be 101110. This results in the DS field value of 10111000 (184d).

This default value would result in a value of “101” precedence bits, low delay, high throughput, and normal reliability in the legacy IP networks (RFC 791, RFC 1812). Network managers of legacy IP networks could use the above-mentioned values to define filters on their routers to take advantage of priority queuing. The default value is based on the Expedited Forwarding PHB (RFC 2598) recommendation.

**Note:** RFC 3168 now defines the state in which to set the two least significant bits in the TOS byte. On the other hand, this RFC only applies to TCP transmissions and the bits are thus set to “0” in the Mediatrix 4102. This has the following effects:

- The TOS values for UDP packets are the same as in the MIB.
- The TOS values for TCP packets are equal to the closest multiple of 4 value that is not greater than the value in the MIB.

**Figure 69:** DiffServ Section




2. Click *Submit* if you do not need to set other parameters.

# Emergency Page

The *Emergency* sub-page of the *Misc* page allows you to configure the Emergency Call parameters of the Mediatrix 4102.

## Emergency Call Configuration

The Emergency Call service (also called urgent gateway) allows a “911”-style service. It allows a user to dial a special digit map resulting in a message being sent to a specified urgent gateway, bypassing any other intermediaries.

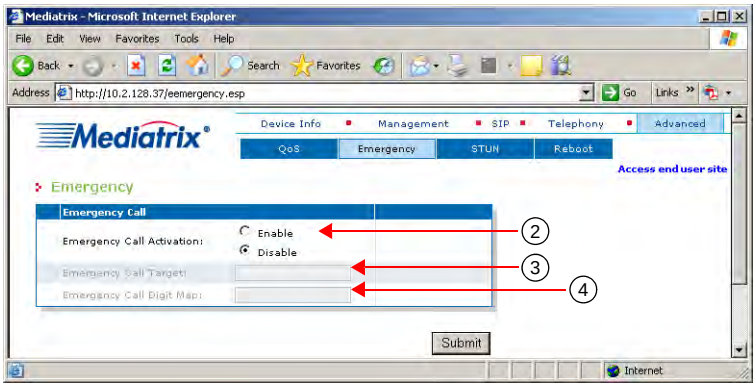
If enabled, whenever the user dials the specified digit map, a message is sent to the target address.

You can also set these parameters via SNMP, as described in “Emergency Call” on page 332.

▶ **To configure the emergency call service:**

1. In the web interface, click the *Advanced* link, then the *Emergency* sub-link.

**Figure 70:** Advanced – Emergency Web Page




2. Enable the emergency call feature by selecting **Enable** in the *Emergency Call Activation* choices.

3. Set the number to reach for an urgent call in the *Emergency Call Target* field.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   Note that this string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

4. Define the digits that users must dial to start the urgent gateway call feature in the *Emergency Call Digit Map* field.

   For instance, you could decide to put “*60” as the sequence a user must dial to start the urgent gateway service. This sequence must follow the syntax for digit maps (see “Digit Maps” on page 87). Dialing this digit map does not have any effect unless the service's status is “enabled”.

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have different sequences for each line.

5. Click *Submit* if you do not need to set other parameters.

# STUN Configuration

| | |
|---|---|
| **Standards Supported** | RFC 3489 – STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) |

STUN (Simple Traversal of UDP through NATs) is a simple client / server protocol that uses UDP packets to discover the configuration information of NATs and firewalls between a device and the public Internet:

- NAT type
- NAT binding public address
- NAT binding time to live

NAT (Network Address Translator) is a device that translates the IP address used within a “private” network to a different IP address known in another “public” network. See “NAT Traversal” on page 292 for more details.

STUN supports a variety of existing NAT devices and does not require any additional hardware or software upgrades on the NAT device.

The Mediatrix 4102 uses the STUN protocol to discover its NAT binding for the following three IP addresses/ports (sockets):

- Signalling protocol (SIP) IP address/port
- RTP IP address/port
- T.38 IP address/port

## SIP Outbound Proxy

For a unit to work properly behind a firewall, it must keep a pinhole opened by sending keepalive packets through the firewall.

The Mediatrix 4102 only sends keepalive packets to the last destination for a specific socket. When a unit is not configured with an outbound proxy, it can send, through its SIP socket, messages to various destinations, such as a SIP redirect server, another SIP unit, or a MWI server. If, for instance, the last SIP message was sent to the MWI server, the Mediatrix 4102 will keep the pinhole opened for the MWI server only (sending keepalive message to the MWI server) and won't be reacheable by other units outside the firewall.

To avoid those issues, all SIP message should come and go from the same source/destination on the public side of the firewall, i.e., a SIP outbound proxy. Media5 thus recommends that you use a SIP outbound proxy. See “SIP Servers Configuration” on page 77 for more details.

## Restrictions on the Media5 STUN Implementation

- The Mediatrix 4102 does not currently support NAT type discovery.
- The Mediatrix 4102 does not currently support STUN NAT binding time to live discovery.
- The Mediatrix 4102 does not currently support the TLS security mechanism.
- Due to a limitation of most routers, an RTP portal might be required in order for two units behind the same NAT/firewall to be able to communicate with each other.

## STUN Client Configuration

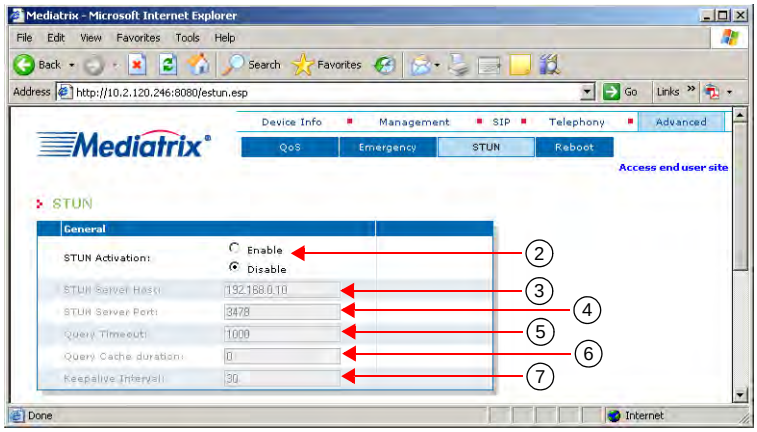
The *STUN* sub-page of the *Advanced* page allows you to configure the STUN client of the Mediatrix 4102.

You can also set these parameters via SNMP, as described in "Chapter 20 - STUN Configuration" on page 315.

**To set STUN parameters:**

1. In the *Advanced* pages, click the *STUN* link.

**Figure 71:** STUN Web Page




2. Enable the STUN client by selecting the **Enable** option in the *STUN Activation* choices.
3. Set the static STUN server IP address or Fully Qualified Domain Name (FQDN) in the *STUN Server Host* field.
4. Set the static STUN server IP port number in the *STUN Server Port* field.

   The default value is **3478**.
5. Set the maximum amount of time, in milliseconds, the Mediatrix 4102 should wait for an answer to a STUN query sent to a STUN server in the *Query Timeout* field.

   Available values range from 500 ms to 10000 ms.

   Caution is advised in setting long timeouts. In the advent of an unresponsive STUN server, the unit may end up waiting a long time before it determines that a call cannot be made due to the STUN server failure.
6. Set the amount of time, in seconds, the Mediatrix 4102 should keep a STUN query result in its internal cache in the *Query Cache duration* field.

   Keeping a query in a cache helps to reduce the amount of unnecessary STUN queries when an RTP or T.38 socket is re-used within a short period of time. Available values range from 0 s to 3600 s.

   When set to **0**, the cache is disabled and the unit performs a STUN query each time a socket needs to be used.
7. Define the interval, in seconds, at which the Mediatrix 4102 sends blank keepalive messages to keep a firewall hole opened in the *Keepalive Interval* field.

   Keepalive messages are used by both the signalling protocol socket and the RTP socket to keep those connections opened through a firewall. Available values range from 0 s to 120 s.

When set to **0**, no keepalive packet is sent.

**Note:** Keepalive messages are not supported on the T.38 socket.

8. Click *Submit* if you do not need to set other parameters.

## SIP Custom NAT Traversal

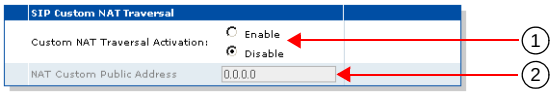
The Mediatrix 4102 may be used in a private domain that is not directly connected to the IP network. For instance, this may be the case for ITSP (Internet Telephony Service Provider) clients that have a small private network. This private network is connected to the public IP network through the NAT (Name Address Translation) technology.

You can configure the Mediatrix 4102 with the public IP address of the NAT system, which allows to reach the unit. SIP packets sent by the Mediatrix 4102 contain the NAT address configured as SIP contact. If the NAT service is not activated, the real IP address of the Mediatrix 4102 is used.

**To set SIP custom NAT traversal parameters:**

1. In the *SIP Custom NAT Traversal* section of the *STUN* page, enable the custom NAT traversal by selecting the **Enable** option in the *Custom NAT Traversal Activation* choices.

**Figure 72:** SIP Custom NAT Traversal Section




2. Enter the public IP address of the NAT system in the *NAT Custom Public Address* variable.

   This is the public IP address used as Contact address by outgoing SIP packets crossing a NAT system.

3. Click *Submit* if you do not need to set other parameters.

# SNMP Configuration

**Page Left Intentionally Blank**

CHAPTER 8

# MIB Structure and SNMP

This chapter describes how the Mediatrix 4102 uses the SNMP protocol for its configuration.

## SNMP Overview

The Mediatrix 4102 uses the Simple Network Management Protocol (SNMP) for initial software configuration provisioning and subsequent software configuration.

SNMP is a simple request-reply protocol for Internet network management services. It consists of *network management stations* (in this document, they are referred to as a management server) communicating with *network elements*. Management stations are normally workstations that display relevant facts about the elements being monitored.

SNMP works over the IP (Internet Protocol) communication stack. SNMP network management consists of three pieces:

- The protocol between the manager and the element (SNMP). This details the format of the packets exchanged. Although a wide variety of transport protocols could be used, UDP is normally used with SNMP.
- A set of common structures and an identification scheme used to reference the variables in the MIB. This is called the *Structure of Management Information* (SMI).
- A *Management Information Base* (MIB) that specifies what variables the network elements maintain (the information that can be queried and set by the manager).

### Definitions

#### Structure of Management Information (SMI)

The SMI is the set of rules for specifying the management information that a device maintains. The management information is actually a collection of managed objects, and these rules are used to both name and define these managed objects.

#### Management Information Base (MIB)

A MIB is a structured collection of all the managed objects a device maintains. The managed objects are structured in the form of a hierarchical tree. At the top of the tree is the most general information available about a network. Each branch of the tree then gets more detailed into a specific network area, with the leaves of the tree as specific as the MIB can get.

#### Object Identifier (OID)

*Object Identifiers* (OID) are strings of numbers. They are allocated in a hierarchical manner, so that, for instance, the authority for “1.2.3” is the only one that can say what “1.2.3.4” means. The formal definition of OIDs comes from ITU-T recommendation X.208 (ASN.1), which is available from the ITU.

## SNMP Versions

The Mediatrix 4102 supports three versions of SNMP: SNMPv1, SNMPv2c and SNMPv3. SNMP defines a few types of messages that are exchanged between the manager and agent.

### SNMPv1 Messages

The following messages are specific to SNMPv1.

**Table 76:** SNMPv1 Message Types

| Operator | Description |
|---|---|
| messages sent from the manager to the agent | |
| get-request | Get the value of one or more variables. |
| get-next-request | Get the next variable after one or more specified variables. |
| set-request | Set the value of one or more variables. |
| messages sent from the agent to the manager | |
| get-response | Return the value of one or more variables. This is the message returned by the agent to the manager in response to the **get-request**, **get-next-request**, and **set-request** operators. |
| trap | Notify the manager when something happens on the agent. |

### SNMPv2c Messages

There are a few flavours of SNMPv2, SNMPv2c being the most common. The following message is specific to SNMPv2.

**Table 77:** SNMPv2 Message Type

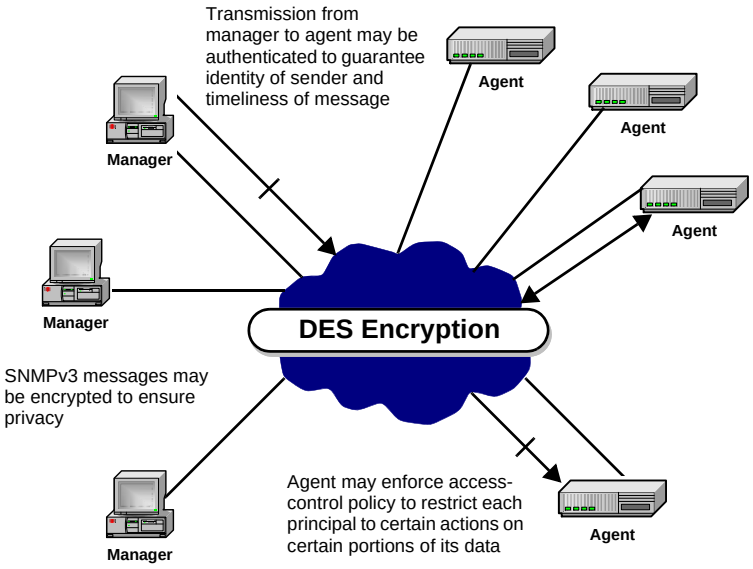
| Operator | Description |
|---|---|
| get-bulk | Uses BULK Requests to query for a tree of information about a network entity. A variable put in command line specifies which portion of the object identifier space will be searched using BULK Requests. All variables in the subtree below the given variable are queried as a single request and their values presented to the user. |

This message is sent from the manager to the agent.

### SNMPv3 Messages

To correct the security deficiencies of SNMPv1/v2, SNMPv3 was defined with an overall SNMP architecture and a set of security capabilities. SNMPv3 includes three important services: *authentication*, *privacy*, and *access control* (Figure 73). To deliver these services in a flexible and efficient manner, SNMPv3 introduces the concept of a *principal*, which is the entity on whose behalf services are provided or processing takes place.

**Figure 73:** SNMPv3 Services




## SNMP Behaviour

When using SNMP, the following rules apply:

- Media5 recommends to copy the SNMPv3 user attributes only twice.
- The administrator may edit the SNMPv3 user attributes:
  - Authentication algorithm (none, MD5, or SHA)
  - Authentication password
  - Encryption algorithm (NULL or DES)
  - Encryption password
  - All SNMPv3 passwords (encryption and authentication) must be at least 8 characters long. You should use the *Unit Manager Network* product to perform SNMPv3 setup. Whatever the MIB browser you use, the unit follows the SNMPv3 standard RFCs.

SNMP can be used in a non-secure or secure management mode.

**Warning:** The SNMPv3 method for changing the password or encryption key contains a flaw which may result in setting the incorrect password. This problem can happen if you use an incorrect “oldpassword” when changing your password. Always exercise great caution when changing your password or encryption key. Note that you can use the factory reset to clear the SNMPv3 password. See “Factory Reset” on page 22 for more details. See also the *Unit Manager Network Administration Manual*.

### Non-Secure Management Mode

In non-secure management mode, the unit responds to SNMP requests as follows:

- SNMPv1: read-write on all MIB tree
- SNMPv2c: read-write on all MIB tree
- SNMPv3: read-write on all MIB tree by using:
  - MD5 authentication
  - Authentication password: “Md5Password” (initial password)
  - DES encryption
  - Encryption password: “DesPassword” (initial password)
- SNMPv3: read-write on all MIB tree by using:
  - SHA authentication
  - Authentication password: “ShaPassword” (initial password)
  - DES encryption
  - Encryption password: “DesPassword” (initial password)

### Secure Management Mode

In secure management mode, the unit responds to SNMP requests as follows:

- SNMPv1: read-only on all MIB tree
- SNMPv2c: read-only on all MIB tree
- SNMPv3: the same values as for SNMPv3 in non-secure management mode

**Note:** If you forget or lose a password, perform a Factory Reset to reset the unit to the non-secure management mode. See “Factory Reset” on page 22 for more details.

### Notes

- When using SNMPv3 with encryption (DES), you may experience delays when accessing MIB variables. This is normal because encrypting an IP packet takes in general longer than sending it over IP. If you experience any timeout, add some seconds to the timeout period of your MIB browser, and then try to reach the unit again.
- Suppose that the Mediatrix 4102 accepts requests with authentication only. If you perform requests by using encryption and authentication, assuming that the authentication password is valid, the SNMP agent still responds as if the requests were only authenticated.
- If you clone an SNMPv3 user, and then remove authentication or privacy for it, ensure that a row in *vacmGroupName* matches its new constraints. If not, the unit is not accessible by using the new clone parameters.

## SNMPv3 Special Behaviour

Mediatrix units coming out of factory are set so that you can use all MIB variables by using SNMPv1, SNMPv2c, or SNMPv3. However, you can decide to accept only SNMPv3 access by using passwords known by administrators only for enhanced security. In this case, you should manually disable SNMPv1 / SNMPv2 so that SNMPv3 works properly. The Mediatrix 4102 thus refuses any SNMPv1 or SNMPv2 request it receives.

You can disable / enable SNMPv1 / SNMPv2 by using the MIB Browser included in the Media5 Unit Manager Network (or any other MIB Browser) to modify the permissions related to SNMPv1 / SNMPv2 (security model). These permissions are located in the *VacmAccessTable* of the SNMP-VIEW-BASED-ACM-MIB (RFC 2575).

When using exclusively SNMPv3, a row from one of the following tables:

- usmUserTable
- vacmSecurityToGroupTable
- vacmAccessTable
- vacmViewTreeFamilyTable

is saved in flash memory only if these conditions are met:

- The RowStatus variable (e.g., *vacmAccessRowStatus*) is equal to **active(1)**.
- The StorageType variable (e.g., *vacmAccessStorageType*) is equal to **nonVolatile(3)**.

**Note:** The *vacmContextTable* is not saved under any condition.

## SNMP Configuration via a Configuration File

You can modify the SNMP configuration of the Mediatrix 4102 by inserting an SNMP Agent section in a configuration file and then transferring this configuration file into the unit. This configuration replaces any configuration set in a profile. For more information on how to use a configuration file for updating the Mediatrix 4102, see "Chapter 14 - Configuration File Download" on page 219.

**Caution:** The SNMP Agent section contains the default Media5 parameters related to SNMP. Default values enable SNMPv1, SNMPv2, and SNMPv3 and provide default Media5 credentials for SNMPv3.

The SNMP Agent section is located in the *SnmpGenericTemplate.xml* file located under *Unit Manager Network 3.2\UnitManager\DefaultCfgFile* folder. The contents of the *SnmpGenericTemplate.xml* file may be appended at the end of the generated XML file. See the Unit Manager Network documentation for more details.

The SNMP agent section must not be separated by other comments or OIDs in the configuration file.

If you transfer a configuration file with an SNMP Agent section that constitutes a change from the SNMPv3 configuration currently in use, the new configuration is applied and the unit then restarts so that the changes take effect.

A few notes:

- Once an SNMPv3 configuration is in effect in the Mediatrix 4102, it is not possible to revert the unit back to SNMPv1 or SNMPv2c by sending it a configuration file that does not include an SNMP Agent configuration section.
- If you perform a factory reset, all settings previously applied via the configuration file (including the SNMPv3 configuration) are lost and the unit reinitializes by using the current profile.

**Figure 74:** SNMP Agent Section Example

```
<SnmpAgentConfig>
 <snmpV2>
  <snmpModules>
   <snmpUsmMIB>
    <usmMIBObjects>
     <usmUser>
      <usmUserTable>
       <a>
        <usmUserSecurityName ValueType="OCTET_STRING" Value="Md5DesUser"/>
        <usmUserAuthProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.1.2"/>
        <usmUserAuthPassword ValueType="OCTET_STRING" Value="Md5DesUser"/>
        <usmUserPrivProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.2.2"/>
        <usmUserPrivPassword ValueType="OCTET_STRING" Value="Md5DesUser"/>
        <usmUserStorageType ValueType="UINT32" Value="4"/>
        <usmUserStatus ValueType="UINT32" Value="1"/>
       </a>
       <b>
        <usmUserSecurityName ValueType="OCTET_STRING" Value="ShaDesUser"/>
        <usmUserAuthProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.1.3"/>
        <usmUserAuthPassword ValueType="OCTET_STRING" Value="ShaDesUser"/>
        <usmUserPrivProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.2.2"/>
        <usmUserPrivPassword ValueType="OCTET_STRING" Value="ShaDesUser"/>
        <usmUserStorageType ValueType="UINT32" Value="4"/>
        <usmUserStatus ValueType="UINT32" Value="1"/>
       </b>
      </usmUserTable>
     </usmUser>
    </usmMIBObjects>
   </snmpUsmMIB>
   <snmpVacmMIB>
    <vacmMIBObjects>
     <vacmMIBViews>
```

```
  <vacmViewTreeFamilyTable>
   <a>
    <vacmViewTreeFamilyViewName ValueType="OCTET_STRING" Value="PublicView"/>
    <vacmViewTreeFamilySubtree ValueType="OID" Value="1"/>
    <vacmViewTreeFamilyMask ValueType="OCTET_STRING" Value="&#128;"/>
    <vacmViewTreeFamilyType ValueType="UINT32" Value="1"/>
   <vacmViewTreeFamilyStorageType ValueType="UINT32" Value="4"/>
   <vacmViewTreeFamilyStatus ValueType="UINT32" Value="1"/>
  </a>
  <b>
   <vacmViewTreeFamilyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
   <vacmViewTreeFamilySubtree ValueType="OID" Value="1"/>
   <vacmViewTreeFamilyMask ValueType="OCTET_STRING" Value="&#128;"/>
   <vacmViewTreeFamilyType ValueType="UINT32" Value="1"/>
   <vacmViewTreeFamilyStorageType ValueType="UINT32" Value="4"/>
   <vacmViewTreeFamilyStatus ValueType="UINT32" Value="1"/>
  </b>
 </vacmViewTreeFamilyTable>
</vacmMIBViews>
 <vacmSecurityToGroupTable>
  <a>
   <vacmSecurityModel ValueType="UINT32" Value="1"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="PublicUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv1RWPublicGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </a>
  <b>
   <vacmSecurityModel ValueType="UINT32" Value="1"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="PrivateUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv1RWPrivateGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </b>
  <c>
   <vacmSecurityModel ValueType="UINT32" Value="2"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="PublicUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv2cRWPublicGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </c>
  <d>
   <vacmSecurityModel ValueType="UINT32" Value="2"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="PrivateUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv2cRWPublicGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </d>
  <e>
   <vacmSecurityModel ValueType="UINT32" Value="3"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="Md5DesUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </e>
  <f>
   <vacmSecurityModel ValueType="UINT32" Value="3"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="ShaDesUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </f>
  <g>
   <vacmSecurityModel ValueType="UINT32" Value="3"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="AuthNoPrivUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="AuthNoPrivGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </g>
  <h>
   <vacmSecurityModel ValueType="UINT32" Value="3"/>
   <vacmSecurityName ValueType="OCTET_STRING" Value="NoAuthNoPrivUser"/>
   <vacmGroupName ValueType="OCTET_STRING" Value="NoAuthNoPrivGrp"/>
   <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
   <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </h>
 </vacmSecurityToGroupTable>
 <vacmAccessTable>
  <a>
   <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
   <vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
   <vacmAccessSecurityLevel ValueType="UINT32" Value="3"/>
```

```
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </a>
     <b>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="AuthNoPrivGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="2"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </b>
     <c>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="NoAuthNoPrivGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </c>
     <d>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv1RWPublicGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="1"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </d>
     <e>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv1RWPrivateGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="1"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </e>
     <f>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv2cRWPublicGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="2"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </f>
     <g>
      <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv2cRWPrivateGrp"/>
      <vacmAccessSecurityModel ValueType="UINT32" Value="2"/>
      <vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
      <vacmAccessContextMatch ValueType="UINT32" Value="1"/>
      <vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/>
      <vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
      <vacmAccessStorageType ValueType="UINT32" Value="4"/>
      <vacmAccessStatus ValueType="UINT32" Value="1"/>
     </g>
    </vacmAccessTable>
   </vacmMIBObjects>
  </snmpVacmMIB>
 </snmpModules>
</snmpV2>
</SnmpAgentConfig>
```

Enable/Disable SNMPv3
Enable/Disable SNMPv3
Enable/Disable SNMPv3
Enable/Disable SNMPv3
Enable/Disable SNMPv3
Enable/Disable SNMPv3
Enable/Disable SNMPv1
Enable/Disable SNMPv1
Enable/Disable SNMPv1
Enable/Disable SNMPv1
Enable/Disable SNMPv2
Enable/Disable SNMPv2
Enable/Disable SNMPv2
Enable/Disable SNMPv2

## Enabling/Disabling SNMPv1, SNMPv2 and SNMPv3

By default, the parameters in the SNMP Agent section enable SNMPv1 and SNMPv2. However, you may want to disable them.

**To enable SNMPv1 and SNMPv2:**

1. Ensure that the *Value* parameters of the fields <*vacmAccessReadViewName*> and <*vacmAccessWriteViewName*> are set to “PublicView” in the following groups:
   - Snmpv1RWPublicGrp
   - Snmpv1RWPrivateGrp
   - Snmpv2cRWPublicGrp
   - Snmpv2cRWPrivateGrp

   These fields are identified in Figure 74 on page 143 with the following icon: ← **Enable/Disable SNMPv*x***.

**To disable SNMPv1 and SNMPv2:**

1. Ensure that the fields <*vacmAccessReadViewName*> and <*vacmAccessWriteViewName*> are empty in the following groups:
   - Snmpv1RWPublicGrp
   - Snmpv1RWPrivateGrp
   - Snmpv2cRWPublicGrp
   - Snmpv2cRWPrivateGrp

   These fields are identified in Figure 74 on page 143 with the following icon: ← **Enable/Disable SNMPv*x***.

**To enable SNMPv3:**

1. Ensure that the *Value* parameters of the fields <*vacmAccessReadViewName*> and <*vacmAccessWriteViewName*> are set to “PublicView” in the following groups:
   - AuthPrivGrp
   - AuthNoPrivGrp
   - NoAuthNoPrivGrp

   These fields are identified in Figure 74 on page 143 with the following icon:
   ← **Enable/Disable SNMPv3**.

**To disable SNMPv3:**

1. Ensure that the fields <*vacmAccessReadViewName*> and <*vacmAccessWriteViewName*> are empty in the following groups:
   - AuthPrivGrp
   - AuthNoPrivGrp
   - NoAuthNoPrivGrp

   These fields are identified in Figure 74 on page 143 with the following icon:
   ← **Enable/Disable SNMPv3**.

## Changing SNMPv3 Credentials

The SNMP Agent section provides default Media5 credentials for SNMPv3. You can change these credentials.

**To change SNMPv3 credentials:**

1. Change the password in the following fields:
   - usmUserAuthPassword (section **Md5DesUser**)
   - usmUserPrivPassword (section **Md5DesUser**)
   - usmUserAuthPassword (section **ShaDesUser**)
   - usmUserPrivPassword (section **ShaDesUser**)

These fields are identified in Figure 74 on page 143 with the following icon : .

**Caution:** SNMPv3 passwords must be at least 8 characters long.

# MIB Structure

The current MIB structure is defined in the SMI file, called *MX-SMI.my*. The SMI contains seven main groups.

**Table 78:** Structure of Management Information

| Group | Description |
|---|---|
| mediatrixProducts | Each Media5 product has been assigned with its own sysObjectID value. |
| mediatrixAdmin | Root of the modules used for the administration of the products. |
| mediatrixMgmt | Root of the modules used to manage the products. |
| mediatrixConfig | Root of the modules used to configure the products. |
| mediatrixIpTelephony Signaling | Root of the modules used to configure the signalling protocols. |
| mediatrixModules | Provides a root in which modules can register their module entity. No MIB variables actually appear under this node. |
| mediatrixExperimental | The experimental sub-tree is the area where objects and events in MIBs under development can be placed without fear of conflicting with other MIBs. When the items rooted under an experimental sub-tree are ready for release, the sub-tree is re-attached under a permanent branch.<br>Please note that Media5' configuration tool – the Unit Manager Network – does not support MIBs that are located under the *mediatrixExperimental* branch of the MIB structure. The Unit Manager Network does not have specific tasks to manage variables in experimental MIBs.<br>Even though the Unit Manager Network can view experimental MIBs, SNMP operations may not work properly on them. |

All parameters in the MIBs have been configured by default upon start up. However, if you need to modify some of these parameters (for example, parameters related to the country in which you are), use a MIB browser.

## Textual Conventions

Textual conventions are defined in a module to ensure that all variables throughout the MIB structure use the same syntax and types. The type of each variable is defined in the *Composed syntax* line.

**Table 79:** Textual Conventions

| Type | Definition |
|---|---|
| MxIpHostName | Represents an IP address or a domain name. |
| MxIpAddress | Represents an IP address. |
| MxIpPort | The TCP or UDP port number range. Values can be between 1 and 65535. |
| MxIpSubnetMask | Represents an Internet subnet mask. |
| MxIpSelect ConfigSource | Indicates the source to use during the next restart sequence for the provisioning of the localHost MIB objects.<br>• static: uses static values provided by the user (such as DNS addresses, router, etc.).<br>• dhcp: uses the DHCP server to retrieve the configuration of the localHost MIB objects. |
| MxIpConfigSource | Indicates the source used during the last restart sequence for the provisioning of the localHost MIB objects.<br>• static: the user provided static values such as DNS addresses, router, etc.<br>• dhcp: the DHCP server was used to retrieve the configuration of the localHost MIB objects.<br>• Default: hardcoded values for recovery mode were used. |
| MxIpDhcpSite SpecificCode | Represents a DHCP site specific code. Values can be between 128 and 254 or 0. You can enter this code in your DHCP server to define IP addresses. Refer to “Chapter 9 - IP Address and Network Configuration” on page 155 for more details. |
| MxFloatingPoint | Represents a floating point number. |
| MxAdvancedIpPort | The TCP or UDP port number range. Values can be between 0 and 65535. The port number value 0 is used for special functionality defined in the variable definition. |
| MxEnableState | Represents an enabled/disabled state (boolean value). |
| MxActivationState | Represents an active/inactive state (boolean value). |
| MxSignalingAddress | Represents a valid signalling address. |
| MxDigitMap | A digit map is a sequence used to determine when the dialing of DTMFs is completed. See “Chapter 22 - Digit Maps” on page 321 fore more details. |

## Objects, Conformance, and Events

Each MIB may have three types of data.

**Table 80:** MIB Data Types

| Type | Description |
|---|---|
| Object | Represents the actual variables that can be set. |
| Conformance | Describes one or more groups to which the product may conform. This allows to have an exact idea of what a unit supports by glancing at the conformance information. |
| Event | An event is sent to tell what type of data will be received, but not the data itself. This is used to “warn” in advance what is coming. |

## IP Addresses

The MIB structure contains many IP addresses that can be set or viewed. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

## Persistence

A variable may either be persistent or volatile.

**Table 81:** Storage Clauses

| Clause | Definition |
|---|---|
| Persistent | *Persistent* parameters are saved into the unit's memory and restored when it restarts. All the variables with the *Access* = *Read Write* attribute are persistent, except the variables representing commands (such as *sysAdminCommand*). |
| Volatile | *Volatile* parameters are lost every time the unit restarts. This type of parameter includes toggling parameters such as requesting a configuration file or a software download. *Statistics* are also volatile parameters that are lost every time the unit restarts. |

## Changing a Parameter Value

The Mediatrix 4102 software parameters are fully programmable by using the SNMP protocol. There are two ways to set up and configure a unit:

- By using a SNMP browser to contact the MIBs of the Mediatrix 4102. It is assumed that you have basic knowledge of TCP/IP network administration.

  You can use the MIB browser built in the Media5' Unit Manager Network. See "Unit Manager Network – Element Management System" on page xxiv for more details.

  You can also use any third-party SNMP browser or network management application running the SNMP protocol to monitor and configure the Mediatrix 4102. However, the information may not be presented in the same manner depending on the SNMP browser used.

- By using the graphical user interface of the Management Server.

  The Management Server could be Media5's Unit Manager Network. See "Unit Manager Network – Element Management System" on page xxiv for more details.

Be sure to use the MIB files that match the version of the MIB located inside the current software build of the unit.

Locate the proper parameter to modify and change (SET) its value. Most of the parameters require to restart the Mediatrix 4102 unit. A restart may be software-initiated or manually initiated by unplugging the unit. It deletes all statistics stored and overwrites all volatile parameter values in the configuration file. A restart also reinitiates the entire unit's initial provisioning sequence.

**Note:** When performing a SET operation on any MIB variable, Media5 recommends to wait at least 30 seconds before shutting down the unit. This gives time to the software to update configuration data in flash memory.

# Tables

There are two types of tables used in the MIB structure. They contain:

- Generic variables that apply to each line of a unit. This avoids to repeat each set of variables for each line it has.
- The administrative commands and status related to a managed object.

## Generic Variables

All tables used to set variables for one or more lines (such as the *voiceIfTable*) are based on the *ifTable*, or interface table.

The *ifTable* lists the interfaces of a unit. In other words, it basically defines the lines that are used by the unit. It contains an *ifIndex*, which defines the interfaces. It may also contain interfaces such as:

- the LoopBack (*lo*) and Ethernet (*eth0*) interfaces.
- the actual voice interfaces (lines) of the unit.

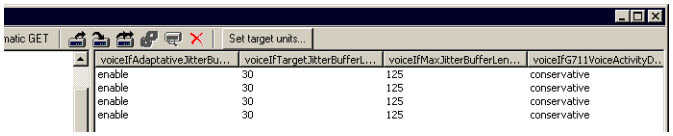
Table 82 gives an example of the *ifTable*.

**Table 82:** ifTable Example

| ifIndex | Type | Description |
|---|---|---|
| 1 | LoopBack | lo |
| 2 | Ethernet(0) | eth0 |
| 3 | Voice FXS | (0) |
| 4 | Voice FXS | (1) |
| 5 | Voice FXS | (2) |
| ... | ... | ... |
| 26 | Voice FXS | (24) |

Figure 75 shows a table built in the Unit Manager Network from the *voiceifTable* parameters.

**Figure 75:** voiceIftable Example




You can perform GET and SET operations on these parameters.

## Variables for Administrative Commands

Administrative commands are built on a hierarchical structure of parents-children. A command applied on a parent is propagated to all of its children.

There are two tables used to define administrative commands to groups:

- *groupAdmin*: A group may be the unit itself (gateway) or other instances. There are no instances other than the gateway defined at this moment.
- *ifAdmin*: This table applies to each interface of the unit.

### groupAdmin Table

The *groupAdmin* table sends administrative commands at the highest instance in a hierarchy (such as the gateway).

**Table 83:** groupAdmin Parameters

| Parameter | Description |
|---|---|
| groupSetAdmin | Command to set the administrative state of the system. |
| groupAdminState | The administrative state of the group. Indicates the current maintenance state of a group. Available states are unlocked, shutting down, and locked. |
| groupOpState | The operational state of the group. It reflects the group's internal state. Available states are enabled and disabled. |
| groupUsageState | The usage state of the group. Indicates the running state of a group. Available states are idle, active, busy, and idle-unusable. |
| groupAdminType | The type of resources managed by the group. |
| groupAdminDescription | The description of the group. |
| groupAdminParent Group | The parent's group. This is the index (*groupAdminIndex*), taken from this table (*groupAdminTable*), of the group that is the parent. If there is no parent, the value "-1" is used. |

### ifAdmin Table

The *ifAdmin* table is similar to the *groupAdmin* table, except that it applies to interfaces.

**Table 84:** ifAdmin Parameters

| Parameter | Description |
|---|---|
| ifAdminSetAdmin | Command to set the administrative state of the current interface. |
| ifAdminAdminState | The administrative state of the current interface. It indicates the current maintenance state of a gateway component. Available states are unlocked, shutting down, locked, and permanentLock. |
| ifAdminOpState | The operational state of the current interface. This state reflects the component's internal state. Available states are enabled and disabled. |
| ifAdminUsageState | The usage state of the current interface. It indicates the running state of a voice component. Available states are idle, active, busy, and idle-unusable. |
| ifAdminParentType | The parents type of this interface. |
| ifAdminParent | The index of the parent of this interface. |

# SNMP Access Limitation

The SNMP access to the Mediatrix 4102 can be limited to only one of its interface or all interfaces.

**To limit the access to the SNMP interface:**

1. In the *snmpAgentMIB*, select the interface where the Mediatrix 4102 can be accessed via SNMP in the *snmpAgentAccess* variable.

   You have the following choices:

**Table 85:** SNMP Access Limitation Parameters

| Access | Description |
|---|---|
| lanOnly | SNMP connections are only permitted on the LAN side, which is usually associated with the *LAN* connector. The LAN IP address is provisioned by the *lanStaticAddress* variable. |
| wanOnly | SNMP connections are only permitted on the WAN side, which is usually associated with the *WAN* connector.<br>However, if the WAN interface is down and the unit reverts to its LAN configuration, the SNMP agent can access the Mediatrix 4102 on its LAN interface. |
| all | SNMP connections are permitted on both the LAN and WAN sides. |

# Current MIB Version

You can find out the version of the MIB currently in the Mediatrix 4102.

1. In the *sysMgmtMIB*, locate the *sysMibVersion* variable.

   This variable displays the current version of the MIB.

# Sending Configuration Data to the Mediatrix 4102

The configuration data can be provisioned into the Mediatrix 4102 in two ways:

- as a configuration file sent from the Management Server to the Mediatrix 4102 via TFTP
- as a MIB sent from the Management Server to the Mediatrix 4102 via SNMP

## Configuration File

The configuration file is the fastest way to deliver the necessary information. This may be important when initializing a large number of units at the same time. The configuration file is mostly used for the initial provisioning sequence (see "Initial Provisioning Sequence" on page 14 for more details).

For more information on how to use a configuration file for updating the Mediatrix 4102, see "Chapter 14 - Configuration File Download" on page 219.

## Management Information Base – MIB

Sending information via SNMP means that individual variables can be changed without sending the whole MIB. You could use a dual system where a configuration file is sent for initial configuration and a MIB browser / SNMP browser is used to implement minor changes.

The Mediatrix 4102 has several configurable MIBs. All variables in these MIBs have been configured by default upon start up. However, if you need to modify some of these variables, use a MIB browser.

CHAPTER 9

# IP Address and Network Configuration

The Mediatrix 4102 must be provisioned with various IP addresses and network parameters to be fully functional. This occurs each time the Mediatrix 4102 is started or when an IP address value is changed in the MIB. The Mediatrix 4102 can use static network parameters as well as parameters provided by a DHCP server, an access concentrator, or even a DNS.

This chapter assumes that you know how to set up and use a DHCP and DNS server. If not, ask your network administrator to set up DHCP-related variables.

This chapter also refers to the MIB structure of the configuration variables. Refer to "Chapter 8 - MIB Structure and SNMP" on page 139 for more details.

## IP Addresses

The MIB structure contains IP addresses that can be set or viewed. These IP addresses are physically located in their relevant MIB. For instance, the IP addresses for the Syslog daemon are located in the *syslogMIB*. However, when viewing the MIB structure in a MIB browser such as the Media5 Unit Manager Network, the IP addresses are grouped in two distinct folders for easy management.

**Table 86:** IP Addresses Folders

| Folder | Description |
|---|---|
| ipAddressStatus | Lists all the IP addresses used by the unit, in read-only format. |
| ipAddressConfig | Lists all the IP addresses you can set. Changes made in this folder are reflected in the *ipAddressStatus* folder. |

### IP Addresses Formats in the DHCP Server

You can use a number of formats when defining IP addresses in the DHCP server.

**Table 87:** IP Addresses Formats in DHCP Server

| Format | Description | Allowed Char. |
|---|---|---|
| Decimal | You can enter IP addresses in the widely-used (base 10) decimal format. For instance, a decimal IP address would be 192.168.0.9.<br>IP addresses cannot contain decimal numbers higher than 255. | 0..9 |
| Hexadecimal | You can enter IP addresses in (base 16) hexadecimal format. Prepending "0x" to the value instructs the unit to interpret it as hexadecimal. For instance, the decimal IP address 192.168.0.9 translates to 0xC0.0xA8.0x0.0x9 in hexadecimal format. | 0..9, A..F |
| Octal | You can enter IP addresses in (base 8) octal format. Prepending "0" to the value instructs the unit to interpret it as octal. For instance, the decimal IP address 192.168.0.9 translates to 0300.0250.00.011 in octal format. | 0..7 |

You can make combinations of the three bases in a single string, because each number in the string is interpreted separately. For instance, 0300.0xA8.000.9 translates to the decimal IP address 192.168.0.9.

There may be some confusion between the three available IP address formats. In particular, it is important to understand that prefixing “0” to the values makes them interpreted as octal values. For instance, the string 192.168.0.009 is not valid because 009 is interpreted in octal, and the digit “9” does not exist in that base.

### Provisioning Source

The Mediatrix 4102 IP information may come from a variety of sources.

**Table 88:** IP Address Provisioning Sources

| Source | Description |
|---|---|
| Static | You manually enter the value and it remains the same every time the Mediatrix 4102 restarts. If you do not specify a value, a default static value applies. |
| DHCP | The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. See RFC 2131 section 2 and RFC 2132. |
| DHCP – Site specific options | The value is obtained at start-time by querying a DHCP server and using a non-standard option specific to the site where the Mediatrix 4102 is used. See “Site Specific Options” on page 170 for more details. |
| DHCP – Vendor specific options | The value is obtained at start-time by querying a DHCP server and using a standard option that is reserved for storing vendor specific information. See “Vendor Specific Options” on page 169 for more details. |
| DNS | The value is obtained at start-time by querying a DNS server. |
| None | The value is not provisioned. The application provides an acceptable default. |
| Automatic | The configuration source is selected by the Mediatrix 4102, following a preference order and the availability of some services. |
| PPP-IPCP[a] | The value is obtained during the PPP network-layer protocol phase from the IPCP configuration options. |

a. See RFC 1332 “The PPP Internet Protocol Control Protocol (IPCP)”, for more details.

# Services

This section describes the services the Mediatrix 4102 uses and their settings. Most of these services require that you define their IP address and, if required, port number. See “DHCP Server Configuration” on page 168 for more details.

Configuration variables of network parameters are defined in the MIB structure under the *ipAddressConfig* folder. This folder is subdivided into groups, one for each service that requires a network parameter.

### Configuration Source

The configuration your Mediatrix 4102 uses can either be:

- dynamically assigned (network parameters assigned by a DHCP Server)
- static (network parameters you manually defined in the MIB structure)

You can also set these parameters via the web interface, as described in “Network Settings” on page 49.

## DHCP Configuration

Using DHCP-assigned IP addresses ensures that the Mediatrix 4102 receives the addresses that are stored in the DHCP server. This assumes that you have previously set the DHCP server with the proper values. See "DHCP Server Configuration" on page 168 for more details.

The Mediatrix 4102 can receive numerous information from the DHCP server, including the vendor or site specific information. Note that the Mediatrix 4102 does not make a DHCP request in the following cases:

- If all MIB variables *xxSelectConfigSource* are set to **static** at start-up.
- If one of the MIB variables *xxSelectConfigSource* is set to **dhcp** after the initialization process.

When the Mediatrix 4102 uses a DHCP server for network parameters, it must always have at least the following three valid parameters:

- IP Address
- Subnet Mask
- Default Gateway

If the parameters are not valid (i.e., the default gateway is not in the same subnet as the IP address), the Mediatrix 4102 will not work properly.

## Verifying the DHCP-Assigned IP Addresses

You can query the MIB structure to see the IP addresses that have been assigned to the Mediatrix 4102. Those IP addresses are located under the *ipAddressStatus* folder in read-only variables.

This assumes that you know the local host IP address. There are two ways to get the local host IP address of a Mediatrix unit:

- Connect a telephone into one of the FXS ports of the Mediatrix unit, dial "*#*0" and listen for the IP address that is given.
- Use the autodetect feature of the Media5 Unit Manager Network product. See "Unit Manager Network – Element Management System" on page xxiv for more details.

## Static Configuration

Using static IP addresses allows you to bypass the DHCP server or still be able to use the Mediatrix 4102 if you are not running a DHCP server.

In this case, having one or more configuration source variable set to DHCP slows down the restart process. If any information is set to come from the DHCP server (for example, SNTP address), the restarting unit waits for a maximum period of two minutes if the DHCP server cannot be reached, even if most other settings are set to "static".

The reason for this delay is that the Mediatrix 4102 cannot function as configured if part of its configuration (the DHCP information) is unavailable. To avoid this problem, you can set all configuration sources the Mediatrix 4102 supports to "static".

In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Removing all DHCP Options*.

**To set all configuration sources to static:**

1. In the *sysAdminMIB*, set the *sysAdminCommand* variable to *setConfigSourcesStatic*.

## Local Host

The *ipAddressConfigLocalHost* group allows you to set the IP information the Mediatrix 4102 needs to work properly. This group is vital to the proper operation of the Mediatrix 4102. If a variable of this group is not properly set, the Mediatrix 4102 may not be able to restart and be contacted after it has restarted.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *IP Configuration*.

▶ **To select the local host configuration source:**

1. In the *ipAddressConfig* folder, locate the *localHostSelectConfigSource* variable (under the *ipAdressConfigLocalHost* group).
2. Set this variable to either **static** or **dhcp**.

**Table 89:** Local Host Variables

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| localHostAddress | “192.168.0.1” | Yiaddr field |
| localHostPrimaryDns[a] | “192.168.0.10” | Option 6 (first of the list) |
| localHostSecondaryDns[a] | “192.168.0.10” | Option 6 (second of the list) |
| localHostDefaultRouter[b] | “192.168.0.10” | Option 3 (first of the list) |
| localHostSubnetMask | “255.255.255.0” | Option 1 |
| localHostDhcpServer | “” (cannot be set) | Siaddr field |

a. If you do not want to use a DNS, set the variable to **0**.

b. If you are not using a default router, set the variable to **0.0.0.0**. Setting the default router IP address to “0.0.0.0” may lead to software download problems. See the troubleshooting section “Software Upgrade Issues” on page 388 for more details.

**Note:** Media5 recommends not to set a static subnet mask address of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

**Note:** If the *localHostDnsOverrideEnable* or *telephonyDnsOverrideEnable* variable is enabled, the primary and secondary DNS addresses are set with static values. See “Static DNS” on page 160 for more details.

In the table above, the only variables that allow an empty string are: *localHostPrimaryDns*, *localHostSecondaryDns* and *localHostDefaultRouter*.

3. Restart the Mediatrix 4102 so that the changes may take effect.

## WAN Address Configuration Source

The Wide Area Network (WAN) address is the public IP address attributed to the Mediatrix 4102. This address is used for incoming signalling, media and management traffic.

▶ **To set the WAN IP address configuration source:**

1. In the *ipAddressConfig* folder, locate the *localHostWanAddressSelectConfigSource* variable (under the *ipAddressConfigLocalHost* group).

   This variable indicates the source to be used for the provisioning of the WAN address. It offers the following choices:

**Table 90:** WAN IP Address Source Settings

| Option | Description |
|---|---|
| localAddress | The WAN address is the one that is set in the *localHostAddress* variable, whereas the *localHostStaticWanAddress* is ignored. |
| static | The Mediatrix 4102 has a static WAN address. The address is configured in the *localHostStaticWanAddress* variable. Note that this setting allows a limited NAT traversal scheme. |
| pppoe | The Mediatrix 4102 uses the PPP over Ethernet in order to obtain its WAN IP address. The PPPoE service must be enabled for the WAN address to be configured. |
| automatic | If the PPPoE service is enabled, the Mediatrix 4102 uses *pppoe* as the configuration source. Otherwise it uses *localAddress*. |

**Table 91:** WAN IP Address Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| localHostWanAddress | “192.168.0.1” | Option IP-Address |

2. Restart the Mediatrix 4102 so that the changes may take effect.

## LAN Interface Configuration

No DHCP value is available, you can define LAN information with only static values.

**Table 92:** LAN Interface Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| lanStaticAddress | 192.168.10.1 | N/A |
| lanStaticNetworkMask | 255.255.255.0 | N/A |

**Note:** Do not set the *lanStaticAddress* variable to 0.0.0.0. This could prevent the unit from properly sending a DHCP discover request.

**Note:** Media5 recommends not to set the *lanStaticNetworkMask* variable to 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

### SNMP Configuration

No DHCP value is available, you can define SNMP information with only static values.

**Table 93:** SNMP Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| localHostSnmpPort | 161 | N/A |

In the *Unit Manager Network Administration Manual*, refer to chapter *Working with SNMP*, section *Setting Unit SNMP Preferences*.

The Mediatrix 4102 uses the SNMP protocol for software configuration. Set the following SNMP-related variable to properly use the protocol.

**Table 94:** SNMP Configuration Variables

| Variable | Description |
|---|---|
| localHostStaticSnmpPort | Default SNMP agent port, which is the port number to use to reach the local host via SNMP protocol. Restart the unit to update this parameter.<br>**Default Value**: 161<br>**Note**: If you change the SNMP agent port, change the port used in the management server or MIB Browser. Not doing so will prevent you from contacting the unit.<br>The Management Server could be the Media5 Unit Manager Network. See "Unit Manager Network – Element Management System" on page xxiv for more details. |

You can query the SNMP information assigned by the DHCP server in the following variables (in the *ipAddressStatus* folder):

- localHostSnmpPort
- msTrapPort

## Static DNS

By default, the Mediatrix 4102 and the networked PC (linked in a LAN with the unit) receive DNS IP addresses according to the configuration source you have defined in the *localHostSelectConfigSource* variable. In general, these addresses are provided by an ISP (Internet Service Provider) via PPPoE or DHCP.

However, you may require that the Mediatrix 4102 and the networked PC use different DNS addresses. If that is the case, you can set static values for the primary and secondary DNS IP addresses, even when the Mediatrix 4102 is set by DHCP. These static values can thus override PPPoE and DHCP provisioning. This feature could be useful in the case where your ISP (Internet Service Provider) and your ITSP (Internet Telephony Service Provider) use different DNS IP addresses or when a Mediatrix 4102 and a networked PC need to use a different DNS.e Mediatrix 4102 may receive DNS addresses from three sources:

- PPPoE or from the static local host DNS IP addresses
- from the static telephony DNS IP addresses

Table 95 explains how DNS addresses are attributed to the Mediatrix 4102 and the networked PC.

**Table 95:** DNS Addresses Possibilities

| localHostDns OverrideEnable | telephonyDns OverrideEnable | DNS address of Mediatrix 4102 | DNS address of the Networked PC |
|---|---|---|---|
| disabled | disabled | DNS from ISP | DNS from ISP |
| disabled | enabled | static telephony DNS | DNS from ISP |
| enabled | disabled | static local host DNS | static local host DNS |

**Table 95:** DNS Addresses Possibilities

| localHostDns OverrideEnable | telephonyDns OverrideEnable | DNS address of Mediatrix 4102 | DNS address of the Networked PC |
|---|---|---|---|
| enabled | enabled | static telephony DNS | static local host DNS |

▶ **To use static DNS IP addresses:**

1. In the *ipAddressConfig* folder, set the *localHostDnsOverrideEnable* variable (under the *ipAdressConfigLocalHost* group) to **enable**.

   The primary DNS and secondary DNS addresses are set with the static values defined in the *localHostStaticPrimaryDns* and *localHostStaticSecondaryDns* variables.

   If you set the variable to **disable**, the primary DNS and secondary DNS addresses provisioning depends on the setting of the *telephonyDnsOverrideEnable* variable.

2. In the *ipAddressConfig* folder, set the *telephonyDnsOverrideEnable* variable (under the *ipAddressConfigTelephonyDns* group) to **enable**.

   The primary DNS and secondary DNS addresses are set with the static values you define in the next step.

3. Set the *telephonyDnsStaticPrimaryDns* and *telephonyDnsStaticSecondaryDns* variables with the proper static DNS IP addresses of your ITSP.

   If you set the *telephonyDnsOverrideEnable* variable to **disable**, the primary DNS and secondary DNS addresses provisioning depends on the setting of the *localHostDnsOverrideEnable* variable.

4. Restart the Mediatrix 4102 so that the changes may take effect.

## Image

The *ipAddressConfigImage* group provides the configuration necessary to download applications into the Mediatrix 4102. This includes emergency downloads in case of repetitive failure to start the main application.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

▶ **To select the Image configuration source:**

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable (under the *ipAddressConfigImage* group).

2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

**Table 96:** Image Information Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| imagePrimaryHost | “192.168.0.10” | Use option specified in variable *imageDhcpPrimarySiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 117, bytes 0-3. |
| imagePrimaryPort | 69[a] | Use option specified in variable *imageDhcpPrimarySiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 117, bytes 4-5. If bytes 4-5 are not present, use the default static value. |

**Table 96:** Image Information Source (Continued)

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| imageSecondaryHost | “192.168.0.10” | Use option specified in variable *imageDhcpSecondarySiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 118, bytes 0-3. |
| imageSecondaryPort | 69[a] | Use option specified in variable *imageDhcpSecondarySiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 118, bytes 4-5. If bytes 4-5 are not present, use the default static value. |

a. This is the well-known TFTP port number as per RFC 1340.

## Management Server

The *ipAddressConfigMs* group provides the configuration necessary for contacting a SNMP management server such as the Media5 Unit Manager Network.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Unit Manager Server*.

▶ **To select the Management Server configuration source:**

1. In the *ipAddressConfig* folder, locate the *msSelectConfigSource* variable (under the *ipAddressConfigMs* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

**Table 97:** Management Server Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| msHost | N/A | Use option specified in variable *msDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 200, bytes 0-3. |
| msStaticHost | “192.168.0.10” | N/A |
| msTrapPort | N/A | Use option specified in variable *msDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 200, bytes 4-5. If bytes 4-5 are not present, use the default static value. |
| msStaticPort | 162 | N/A |
| msStaticTrapPort | 162 | N/A |

**Note:** If you change the value of the *msStaticTrapPort* variable, change the port used in the management server. Not doing so will prevent you from viewing the received traps from the unit.

## Configuration File Fetching

The *ipAddressConfigFileFetching* group provides the configuration necessary to contact the configuration file server when fetching a configuration file.

▶ **To select the configuration file fetching server configuration source:**

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable (under the *ipAddressConfigFileFetching* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

**Table 98:** Configuration File Fetching Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| configFileFetching Host | N/A | Use option specified in variable *configFileFetchingDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 201, bytes 0-3. |
| configFileFetching Port | N/A | Use option specified in variable *configFileFetchingDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 201, bytes 4-5. If bytes 4-5 are not present, use the default static value. |
| configFileFetching StaticHost | “192.168.0.10” | N/A |
| configFileFetching Static Port | 69 | N/A |

3. Restart the Mediatrix 4102 so that the changes may take effect.

## Syslog

The *ipAddressConfigSyslog* group provides the configuration necessary for contacting a Syslog server.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Syslog Daemon*.

▶ **To select the Syslog configuration source:**

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfigSource* variable (under the *ipAddressConfigSyslog* group).
2. Set this variable to either **static** or **dhcp**.

**Table 99:** Syslog Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| syslogHost | “192.168.0.10” | Use option specified in variable *syslogDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 110, bytes 0-3. |
| syslogPort | 514[a] | Not provided by the DHCP, use the default static value. |

a. The port number is as per RFC 1340.

## SIP Servers

The *ipAddressConfigSipServer* group provides the configuration necessary for contacting different SIP servers.

In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

**Note:** Although the DHCP option #120 is reserved for SIP servers, no standard currently defines the content and layout of this option.

**Note:** If, for a given server, the port is 0, then the host and port for this server are obtained through a DNS SRV request. See "Chapter 11 - DNS SRV Configuration" on page 185 for more details.

▶ **To select the SIP Servers configuration source:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

**Table 100:** SIP Servers Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| sipHomeDomain ProxyHost | "192.168.0.10" | Use option specified in variable *sipHomeDomainProxyDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 204, bytes 0-3 |
| sipHomeDomain ProxyPort | 0 | Use option specified in variable *sipHomeDomainProxyDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 204, bytes 4-5. If bytes 4-5 are not present, use the default static value. |
| sipOutboundProxy Host | "0.0.0.0" | Use option specified in variable *sipOutboundProxyDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 205, bytes 0-3. |
| sipOutboundProxy Port | 0 | Use option specified in variable *sipOutboundProxyDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 205, bytes 4-5. If bytes 4-5 are not present, use the default static value. |
| sipRegistrarHost | "192.168.0.10" | Use option specified in variable *sipRegistrarDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 203, bytes 0-3. |
| sipRegistrarPort | 0 | Use option specified in variable *sipRegistrarDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 203, bytes 4-5. If bytes 4-5 are not present, use the default static value. |

**Table 100:** SIP Servers Source (Continued)

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| sipPresenceCompositorHost | "0.0.0.0" | Use option specified in variable *sipPresenceCompositorDhcpSiteSpecificCode*, bytes 0-3.<br>If not specified (0), use option 43, sub-option 206, bytes 0-3. |
| sipPresenceCompositorPort | 0 | Use option specified in variable *sipPresenceCompositorDhcpSiteSpecificCode*, bytes 4-5.<br>If not specified (0), use option 43, sub-option 206, bytes 4-5. If bytes 4-5 are not present, use the default static value. |

## SNTP

The *ipAddressConfigSntp* group provides the configuration necessary for contacting a NTP/SNTP server.

If you are using a NTP or SNTP server (see "Chapter 21 - SNTP Settings" on page 317 for more details), the DHCP server already has options that can be set to provide time server addresses, and the order in which clients use them to attempt to discover servers.

The Mediatrix 4102 uses *Option 42* to specify the IP address corresponding to the server that provides NTP/SNTP (RFC 1769).

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *SNTP*.

**To select the SNTP configuration source:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfigSource* variable (under the *ipAddressConfigSntp* group).
2. Set this variable to either **static** or **dhcp**.

**Table 101:** SNTP Source

| Variable | Default Static Value | DHCP Source |
|---|---|---|
| sntpHost | "192.168.0.10" | Option 42 (first of the list). |
| sntpPort | 123 | Not provided by the DHCP, use the default static value. |

# LAN Connector Static IP Address

You can use the *LAN* connector of the Mediatrix 4102 with the network card of a computer. You could then use this computer to directly access the unit via its LAN interface.

This section describes how to set the static IP address of the *LAN* connector and use this LAN static IP address according to the VLAN substitution or always enabled configuration.

See also "LAN and WAN with VLAN substitution" on page 370 for information on the *LAN* connector behaviour when forwarding network traffic between the WAN and the LAN interfaces of the Mediatrix 4102.

▶ **To set the *LAN* connector static IP address:**

1. In the *ipRoutingMIB*, set the *lanStaticAddressActivation* variable to one of the following values:

**Table 102:** LAN Connector Static IP Address

| Parameter | Description |
|---|---|
| ipRouting | To set the *lanStaticAddress* variable as the Local Area Network (LAN) IP address used by the Mediatrix 4102's *LAN* interface, set the variable *ipRoutingEnable* to **enable** as described in "Enabling TAS" on page 209. |
| vlanSubstitution | To set the *lanStaticAddress* as the Local Area Network (LAN) IP address used by the Mediatrix 4102's *LAN* interface, set the variable *qosVlanIeee8021qSubstitutionEnable* to **enable**. Also set the VLAN Substitution feature as described in "VLAN Substitution" on page 369.<br>See also "LAN and WAN with VLAN substitution" on page 370 for information on the *LAN* connector behaviour when forwarding network traffic between the WAN and the LAN interfaces of the Mediatrix 4102. |
| always | Sets the *lanStaticAddress* as the Local Area Network (LAN) IP address used by the Mediatrix 4102's *LAN* interface. |

2. In the *ipAddressConfig* folder, set the static *LAN* connector information as follows:

**Table 103:** LAN Interface Source

| Variable | Default Static Value | Description |
|---|---|---|
| lanStaticAddress | 192.168.10.1 | LAN IP address used by the unit's *LAN* interface. |
| lanStaticNetworkMask | 255.255.255.0 | LAN subnet mask used by the unit's *LAN* interface. |

**Note:** Do not set the *lanStaticAddress* variable to 0.0.0.0. This could prevent the unit from properly sending a DHCP discover request.

**Note:** Media5 recommends not to set the *lanStaticNetworkMask* variable to 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

3. Restart the Mediatrix 4102 so that the changes may take effect.

# DHCP Configuration

The following sections describes paramaters that you can set on the Mediatrix 4102 to better interact with a DHCP server.

## DHCP Options Waiting Time

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. If the Mediatrix 4102 is connected to such a switch, the switch may shut down the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent proper communication. It may thus take up to three minutes before the unit receives an answer to its request for DHCP options.

You can configure the Mediatrix 4102 to bypass this waiting period by restarting without a valid address for the requested servers.

▶ **To configure how much time the Mediatrix 4102 waits for DHCP options:**

1. In the *sysConfigMIB*, set the amount of time (in seconds) the Mediatrix 4102 will wait for DHCP options in the *sysConfigDhcpWaitDelay* variable.

   This variable is only effective when the variable *sysConfigDhcpWait* is set to **disable**. The default value is **10** seconds.

2. Define the restart behaviour of a unit that needs to issue a DHCP request to receive some options in the *sysConfigDhcpWait* variable.

   This variable has no influence on the wait behaviour in effect when the unit is requesting an IP address lease from the DHCP server.

**Table 104:** DHCP Wait Parameters

| Parameter | Description |
|---|---|
| enable | The unit waits up to 130 seconds for DHCP options and then, in the absence of a response, proceeds to restart without a valid address for the requested servers. This is the default value. |
| disable | The unit restarts, without valid server addresses, if a DHCP response has not been received after the amount of time specified in the variable *sysConfigDhcpWaitDelay*. |

## Bootp BROADCAST Flag in DHCP Requests

| | |
|---|---|
| **Standards Supported** | • RFC 1542 – Clarifications and Extensions for the Bootstrap Protocol |

You can define whether the Mediatrix 4102 sets the Bootp BROADCAST flag in DHCP requests it issues.

This applies to the DHCP Discover requests the unit sends on startup. It does not apply to the DHCP Discover requests sent when renewing the lease.

▶ **To define the BROADCAST flag behaviour:**

1. In the *sysConfigMIB*, define the BROADCAST flag behaviour in the *sysConfigBootpFlags* variable.

**Table 105:** BROADCAST Flag Parameters

| Parameter | Description |
|---|---|
| noFlags | The Bootp BROADCAST flag is not set in DHCP requests that the Mediatrix 4102 issues. The DHCP server may thus answer by using unicast delivery. |
| broadcastFlag | The Bootp BROADCAST flag is set in DHCP requests that the Mediatrix 4102 issues. |

## Changing the Size of DHCP Requests

You can append a string to the value used as Vendor Class ID (Option 60) in a DHCP request. This option is useful when servers require that DHCP packets sent to them be of a minimum size. A string of arbitrary characters (including blanks) can then be used to artificially increase the size of DHCP requests.

See "Vendor Class ID" on page 170 for more details.

▶ **To change the size of DHCP requests:**

1. In the *sysConfigMIB*, define the string to append in the *sysConfigProductNamePadding* variable.

# DHCP Server Configuration

| | |
|---|---|
| **Standards Supported** | • RFC 2131 – Dynamic Host Configuration Protocol, section 2<br>• RFC 2132 – DHCP Options and BOOTP Vendor Extensions |

**Note:** This section applies only if you are using the DHCP connection type.

DHCP servers generally allocate a range of IP addresses for use on a network and reserve IP addresses for specific devices using a unique identifier for each device. The Mediatrix 4102 unique identifier is its media access control (MAC) address.

**Note:** Media5 recommends to use a Windows 2000- or Unix-based DHCP server. If you run Windows NT 4.0 and use the built-in Microsoft DHCP Server, use the Site Specific instead of Vendor Specific information.

You can locate the MAC address as follows:

- on the label located on the bottom side of the unit.
- in the *sysMgmtMIB* under the *sysMacAddress* variable.
- you can dial the following digits on a telephone connected to the Mediatrix 4102:

  `*#*1`

  The Mediatrix 4102 answers back with its MAC address. See "Special Vocal Features" on page 16 for more details.

Media5 recommends to reserve an IP address with an infinite lease for each Mediatrix 4102 on the network.

## Connection to the DHCP Behaviour

When the Mediatrix 4102 restarts, it requests a DHCP offer to get its IP addresses and network information. The Mediatrix 4102 waits four seconds before sending another request. The delay between each request is increased exponentially after each request up to a maximum delay of 64 seconds, and then restarts at a 4 seconds delay.

- first request: 4 seconds delay

- second request: 8 seconds delay
- third request: 16 seconds delay
- fourth request: 32 seconds delay
- fifth request: 64 seconds delay
- sixth request: 4 seconds delay
- seventh request: 8 seconds delay
- etc.

The Mediatrix 4102 stops broadcasting as soon as it receives at least one reply. If the offer is valid, the Mediatrix 4102 takes it and continues its initialization procedure.

**Note:** If the *localHostSelectConfigSource* variable is set to **static** and any other *xxSelectConfigSource* variable is set to **dhcp**, the Mediatrix 4102 makes its DHCP request that will be released immediately.

### Network Configuration

Table 106 lists some of the network options to configure in the DHCP server:

**Table 106:** Network Configuration

| Information | Description | Option | Data Format | Example |
|---|---|---|---|---|
| Subnet Mask | Specifies subnet configuration | 001 | xxx.xxx.xxx.xxx | 255.255.255.0 |
| Routers | List of routers on your network | 003 | Array of IP Addresses | 192.168.10.1<br>192.168.10.2 |
| DNS Servers | List of DNS servers on your network | 006 | Array of IP Addresses | 192.168.10.11<br>192.168.10.12 |

# Vendor and Site Specific DHCP Options

**Note:** This section applies only if you are using the DHCP connection type.

This section briefly describes vendor and site specific DHCP options.

Most of the MIB variables described in "Services" on page 156 require that you define their IP address and, if required, port number. When defining these variables, you can do so in two ways: via vendor specific options or site specific options.

The default value is to use the vendor specific codes. In this case, the *xxSiteSpecificCode* MIB variables are set to 0.

If you want to use site specific codes instead, change the value of the *xxSiteSpecificCode* MIB variables from the default value (0) to the value you select in the DHCP server. See "Site specific code definition" on page 174 for an example of vendor specific and site specific settings.

### Vendor Specific Options

| | |
|---|---|
| **Standards Supported** | RFC 2132 – DHCP Options and BOOTP Vendor Extensions, section 8.4 ("Vendor-specific options") |

The vendor specific DHCP option is a standard DHCP option used to store information specific to the vendor of the DHCP client. The vendor specific option code is 43. Because there are different information elements that can be stored in this option, each element has been allocated a "sub-option" number. See Table 107 on page 171 for the complete list.

Like all other options, the vendor specific information field (option 43) first contains a code (43), a length (in byte) and some data that spans the number of bytes specified in the length.

The data is organized as a series of sub-options, each of them laid-out like a regular option (code, length, data). The codes can be anything between 1 and 254, and the vendor, Media5, chooses these codes. See Table 107 on page 171 for actual codes.

The following figures show the general and encapsulated layout of the vendor specific information option.

**Figure 76:** General Layout of a Vendor Specific Information Option

**Figure 77:** Layout for Encapsulated Vendor Specific Options

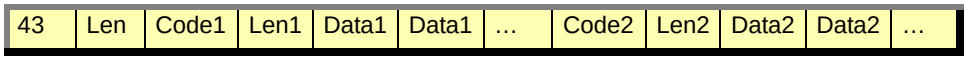
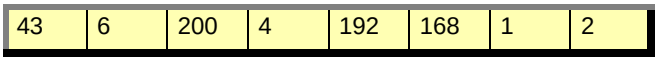



Figure 78 is an example of a vendor specific option containing an *msHost* IP address (192.168.1.2).

**Figure 78:** Example of Encapsulated Vendor Specific Option




Mediatrix units store two types of information in vendor specific options: IP addresses with optional port number and FQDNs with optional port number. The layout for storing IP addresses is explained in section "Entering IP Addresses" on page 171. The layout for storing FQDNs is explained in section "Entering FQDNs" on page 172.

### Vendor Class ID

When using the vendor specific option, first define a Vendor Class ID for the Mediatrix 4102 (not supported in Windows NT servers). A Vendor Class ID can be used by DHCP clients to identify their vendor type and configuration. When using this option, vendors can define their own specific identifier values to convey a particular hardware or operating system configuration or other identifying information.

Where vendor classes are used, the DHCP server responds to identifying clients by using option code 43, the reserved option type for returning vendor specific information to the client.

DHCP servers that do not interpret this option type are expected to ignore it when it is specified by clients.

Please refer to your DHCP server's documentation to learn how to create a new vendor class.

**Note:** The class to add is *Mediatrix 4102*.

### Creating Vendor Specific Information

Once the Vendor ID Class is created, place the proper values in the 43 option of the DHCP server. The 43 option contains sub-options that are encapsulated (according to the format described in RFC 2132).

If the option is not in the DHCP server, the Mediatrix 4102 uses an invalid value (0.0.0.0:0).

Please refer to your DHCP server's documentation to learn how to create vendor specific information. See "Entering IP Addresses" on page 171 for more details on the syntax to use.

## Site Specific Options

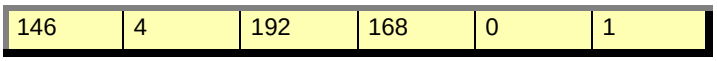
| | |
|---|---|
| **Standards Supported** | RFC 2132 – DHCP Options and BOOTP Vendor Extensions, section 2 ("BOOTP Extension/DHCP Option Field Format"). |

Site specific options are non-standard DHCP options specific to the network where the Mediatrix 4102 is used. You are responsible to allocate an option number (between 128 and 254) for each information element to be stored.

Mediatrix units store two types of information in site specific options: IP addresses with optional port number and FQDNs with optional port number. The layout for storing IP addresses is explained in section "Entering IP Addresses" on page 171. The layout for storing FQDNs is explained in section "Entering FQDNs" on page 172.

Figure 79 is an example of site specific option #146, containing address 192.168.0.1.

**Figure 79:** Site Specific Option Example




When using the site specific option, you can place the values in the site specific options of your choice in the DHCP server. You must then enter the values in the proper MIB variables.

Please refer to your DHCP server's documentation to learn how to create site specific information. See "Entering IP Addresses" on page 171 for more details on the syntax to use.

## Option Codes

This table lists all vendor specific sub-option codes.

**Table 107:** Sub-Option Codes

| Code | | Description |
|---|---|---|
| **Decimal** | **Hexadec.** | |
| 110 | 0x6E | Syslog Server address and port. |
| 117 | 0x75 | Image Primary Server host address and port.<br>The default port number is **69** if you are using TFTP as protocol.<br>The default port number is **80** if you are using HTTP as protocol. |
| 118 | 0x76 | Image Secondary Server host address and port.<br>The default port number is **69** if you are using TFTP as protocol.<br>The default port number is **80** if you are using HTTP as protocol. |
| 200 | 0xC8 | Management Server SNMP Trap host address and port. |
| 201 | 0xC9 | Configuration file fetching host address and port.<br>The default port number is **69** if you are using TFTP as protocol.<br>The default port number is **80** if you are using HTTP as protocol. |
| 203 | 0xCB | SIP Registrar host address and port. |
| 204 | 0xCC | SIP Home Domain Proxy host address and port. |
| 205 | 0xCD | SIP Outbound Proxy host address and port. |
| 206 | 0xCE | SIP Presence Compositor host address and port. |

## Entering IP Addresses

In the DHCP server, IP addresses can be entered in decimal, hexadecimal or octal format. See "IP Addresses" on page 155 for more details.

There are two formats of address string:

- Long: Has a size of 6 bytes (12 hexadecimal characters) and includes the IP address and port.
- Short: Has a size of 4 bytes (8 hexadecimal characters) and includes only the IP address. In this case, the default port is used.

Numeric values are stored in network byte order (Big-Endian).

**Table 108:** Address String Formats

| Variable | Valid Range | Typical Value | Note |
|---|---|---|---|
| IP Address | Any valid IP address | 192.168.0.2<br>(hex. 0xC0.0xA8.0x0.0x2) | N/A |
| Port | 1 - 32,768 | 162<br>(hex. 0xA2) | Not present in the format with dimension 4. |

When entering IP addresses in the DHCP server, there is a difference between the vendor specific option and the site specific option.

The vendor specific options must be encapsulated because more than one information can be stored in this option:

```
[code][length][4-6 bytes address][another code][another length][another
address]...
```

The site specific options can have only one information per option:

```
[4-6 bytes address]
```

The DHCP server adds the proper code and length in the packet it sends out.

### Example

The following example shows how to enter the Syslog (code 110) IP address 192.168.0.10 (with the default port used) and the same address at port 2545 in hexadecimal format.

**Figure 80:** Example – Short Address String

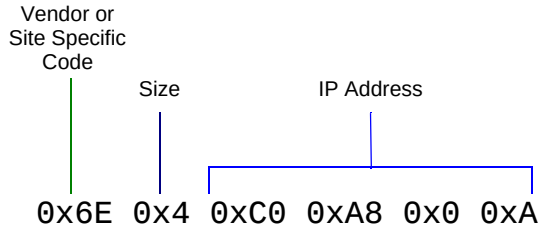
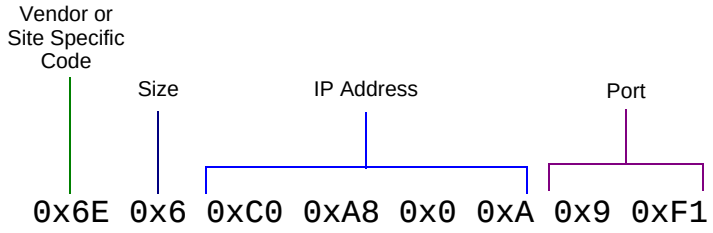



**Figure 81:** Example – Long Address String




## Entering FQDNs

The FQDN address layout is a Media5 proprietary extension to the IP address layout. This format allows the configuration of an IP address in binary format (with or without port) or a FQDN in string format (with or without port) in the same option. The method to decode the information is based on the length of the option: a length of 4 or 6 is decoded as an IP address in binary format and a length higher than 6 is decoded as a FQDN in string format.

The IP address in binary format (with and without port) is explained in section "Entering IP Addresses" on page 171.

The FQDN in string format consists of an array of characters representing the FQDN address.

**Figure 82:** FQDN String Format (without a port number)

| Code | Len (7 to n) | FQDN char 1 | FQDN char 1 | ... | FQDN char n |
|---|---|---|---|---|---|

You can specify a port by adding the port number in string format after a ':' at the end or the FQDN.

**Figure 83:** FQDN String Format (without a port number)

| Code | Len (7 to n) | FQDN char 1 | FQDN char 1 | ... | FQDN char n | ":" (0x3A) | Port char 1 | ... | Port char y |
|---|---|---|---|---|---|---|---|---|---|

The space or null (ASCII code 0) character can be used as padding at the end of the string to have a length higher than 6, since all spaces and nulls are ignored. Note that an IP address can be defined in string format.

## Examples

The following are some examples of the DHCP server configuration (based on linux dhcpd).

**Vendor specific options – option vendor-encapsulated-options**

- Syslog Server (IP address "192.168.0.1" in binary format).
  `6e:04:c0:a8:00:01:`
- Primary Image Server (IP addess and port "192.168.0.10:6000" in binary format).
  `75:06:c0:a8:00:01:17:70:`
- Secondary Image Server (IP address "192.168.0.1" in string format).
  `76:0b:31:39:32:2e:31:36:38:2e:30:2e:31:`
- Management Server (IP address and port "192.168.0.1:6000" in string format.
  `c8:10:31:39:32:2e:31:36:38:2e:30:2e:31:3a:36:30:30:30:`
- Configuration File Fetching (FQDN "server.com").
  `c9:0a:73:65:72:76:65:72:2e:63:6f:6d:`
- SIP Registrar Server (FQDN and port "server.com:6000").
  `cb:0f:73:65:72:76:65:72:2e:63:6f:6d:3a:36:30:30:30:`
- SIP Proxy Server (FQDN with space padding "svr").
  `cc:07:73:76:72:20:20:20:20:`
- SIP Outbound Proxy Server (FQDN and port with space padding "svr:12").
  `cd:07:73:76:72:3a:31:32:20:`
- SIP Presence Server (FQDN with a null terminated string "server.com").
  `ce:0b:73:65:72:76:65:72:2e:63:6f:6d:00;`

**Site specific options**

- Syslog Server (IP address "192.168.0.1" in binary format). The IP-address or string format of dhcpd can be used.
  `option mx-syslog-ip 192.168.0.1;`
  Or "option mx-syslog-str c0:a8:00:01;"
- Primary Image Server (IP addess and port "192.168.0.10:6000" in binary format).
  `option mx-primary-image-str c0:a8:00:01:17:70;`
- Secondary Image Server (IP address "192.168.0.1" in string format).
  `option mx-secondary-image-str "192.168.0.1";`
- Management Server (IP address and port "192.168.0.1:6000" in #string format.
  `option mx-ms-str "192.168.0.1:6000";`
- Configuration File Fetching (FQDN "server.com").
  `option mx-filefetching-str "server.com";`
- SIP Registrar Server (FQDN and port "server.com:6000").
  `option mx-sip-registrar-str "server.com:6000";`

- SIP Proxy Server (FQDN with space padding "svr").

```
option mx-sip-proxy-str "svr    ";
```

- SIP Outbound Proxy Server (FQDN and port with space padding "svr:12").

```
option mx-sip-outbound-proxy-str "svr:12 ";
```

- SIP Presence Server (FQDN with a null terminated string "server.com"). The FQDN is expressed in hexadecimal to be able to put a null character.

```
option mx-sip-presence-proxy-str 73:65:72:76:65:72:2e:63:6f:6d:00;
```

**Site specific code definition**

```
option mx-syslog-ip code 129 = ip-address;
option mx-syslog-str code 129 = string;
option mx-primary-image-ip code 130 = ip-address;
option mx-primary-image-str code 130 = string;
option mx-secondary-image-ip code 131 = ip-address;
option mx-secondary-image-str code 131 = string;
option mx-ms-ip code 132 = ip-address;
option mx-ms-str code 132 = string;
option mx-filefetching-ip code 133 = ip-address;
option mx-filefetching-str code 133 = string;
option mx-sip-registrar-ip code 134 = ip-address;
option mx-sip-registrar-str code 134 = string;
option mx-sip-proxy-ip code 135 = ip-address;
option mx-sip-proxy-str code 135 = string;
option mx-sip-outbound-proxy-ip code 136 = ip-address;
option mx-sip-outbound-proxy-str code 136 = string;
option mx-sip-presence-ip code 137 = ip-address;
option mx-sip-presence-str code 137 = string;
```

## Settings Example

Let’s say for instance you want:

- the Image server at 10.3.2.154 (static)
- the Management Server via DHCP in the vendor specific options
- the Syslog server via DHCP in the site specific option #250

The following are the corresponding MIB values:

- imageSelectConfigSource = static
- imageStaticPrimaryHost = 10.3.2.154
- msSelectConfigSource = dhcp
- msDhcpSiteSpecificCode = 0
- syslogSelectConfigSource = dhcp
- syslogDhcpSiteSpecificCode = 250

The following is the corresponding DHCP setup, assuming the Management server is located at 10.3.2.201 and the Syslog server is located at 10.3.2.200 (port 1024):

- Option 43 (vendor specific option) contains the hexadecimal sequence 0xC80x40xA0x30x20xC9 **inserted among other sequences**.

**Table 109:** Hexadecimal Sequence - Option 43

| Hexadecimal Part | Corresponding Information |
|---|---|
| 0xC8 | code 200 (management server) |
| 0x4 | size of 4 bytes |
| 0xA0x30x20xC9 | IP address 10.3.2.201 |

- Option 250 (site specific option) contains the hexadecimal sequence 0xA0x30x20xC80x400.

**Table 110:** Hexadecimal Sequence - Option 250

| Hexadecimal Part | Corresponding Information |
|---|---|
| 0xA0x30x20xC8 | IP address 10.3.2.200 |
| 0x400 | port 1024 |

# Error Handling

In the event of a network or server failure, this section describes the application behaviour and/or replacement values to use.

**Table 111:** Replacement Values for Error Recovery

| Type | Variable | Replacement value |
|---|---|---|
| IP address | (All variables of that type) | 0.0.0.0 |
| String | (All variables of that type) | “” |

## DHCP Server Failures

If the Mediatrix 4102 cannot contact the DHCP server, it performs one of the following actions:

1. Retries contacting the DHCP server until it answers. The Mediatrix 4102 does not restart.
2. Uses the replacement value from Table 111 for all variables that depend on the DHCP.

This assumes that the Mediatrix 4102 is set to get its IP information via a DHCP server.

If the Mediatrix 4102 is configured to request some DHCP options but does not require an IP address from the server, the amount of time it will wait for an answer before proceeding further is configurable as defined in “Appendix - DHCP Options Waiting Time” on page 167 .

## Vendor/Site Specific Option Missing

If a vendor specific or site specific option is missing from the DHCP server answer, the Mediatrix 4102 uses the replacement value from Table 111 for each variable that depends on missing vendor/site specific options.

## DNS Failures

If the DNS cannot be contacted, the Mediatrix 4102 performs the following steps:

1. The Mediatrix 4102 sends a first request to the primary DNS server.
2. If the DNS server cannot be contacted within two seconds, the Mediatrix 4102 sends a request to the secondary DNS server.
3. If the secondary DNS server cannot be contacted, the Mediatrix 4102 uses the replacement value from Table 111 for all variables that depend on the DNS.

# Ethernet Connection Speed

You can set the speed of the Ethernet connection of the Mediatrix 4102.

You can also set these parameters via the web interface, as described in "Ethernet Connection Speed" on page 49.

▶ **To set the Ethernet connection speed:**

1. In the *sysConfigMIB*, set the Ethernet connection speed of the:
   - *WAN* connector in the *sysConfigNetworkEthernetSpeed* variable
   - *LAN* connector in the *sysConfigComputerEthernetSpeed* variable.

   The following values are available:
   - Auto detect
   - 10Mbs-HalfDuplex
   - 100Mbs-HalfDuplex
   - 10Mbs-FullDuplex
   - 100Mbs-FullDuplex

   A half-duplex connection refers to a transmission using two separate channels for transmission and reception, while a full-duplex connection refers to a transmission using the same channel for both transmission and reception.

   If unknown, set the variable to **Auto detect** so that the Mediatrix 4102 can automatically detect the network speed.

**Caution:** Whenever you force a connection speed / duplex mode, be sure that the other device and all other intermediary nodes used in the communication between the two devices have the same configuration. See "Speed and Duplex Detection Issues" on page 176 for more details.

## Speed and Duplex Detection Issues

There are two protocols for detecting the Ethernet link speed:

- An older protocol called parallel detection.
- A more recent protocol called auto-negotiation (IEEE 802.3u).

The auto-negotiation protocol allows to detect the connection speed and duplex mode. It exchanges capabilities and establishes the most efficient connection. When both endpoints support the auto-negotiation, there are no problems. However, when only one endpoint supports auto-negotiation, the parallel detection protocol is used. This protocol can only detect the connection speed; the duplex mode cannot be detected. In this case, the connection may not be established.

The Mediatrix 4102 has the possibility to force the desired Ethernet link speed and duplex mode by disabling the auto-negotiation and selecting the proper setting (*sysConfigNetworkEthernetSpeed* or *sysConfigComputerEthernetSpeed* variable). When forcing a link speed at one end, be sure that the other end (a hub, switch, etc.) has the same configuration. To avoid any problem, the link speed and duplex mode of the other endpoint must be exactly the same.

CHAPTER 10

# SIP Servers

The Mediatrix 4102 uses the following types of servers:

- Registrar server
- Proxy server
- Outbound Proxy server
- Presence Compositor server

This chapter describes how to configure the Mediatrix 4102 to properly use these servers.

You can also set these parameters via the web interface, as described in "SIP Servers Configuration" on page 77.

## Registrar Server

The registrar server accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.

In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

### Configuration Source

The Mediatrix 4102 must know the IP address and port number of the Registrar server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

**Note:** When defining whether or not the Mediatrix 4102 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

#### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

**To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its registrar server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **dhcp**.

   You can query the registrar server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):

   - sipRegistrarHost
   - sipRegistrarPort

3. Set how you want to define the registrar server information in the DHCP server.

**Table 112:** Registrar Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *sipRegistrarDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to **0**. Set the registrar server IP address in the DHCP server inside the vendor specific sub-option 203 (hexadecimal 0xCB). |
| site specific code | The *sipRegistrarDhcpSiteSpecificCode* variable to any value between 128 and 254. Set the registrar server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *sipRegistrarDhcpSiteSpecificCode* variable in the unit's configuration). |

## Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its registrar server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 113:** Registrar Server Static Information

| Variable | Description |
|---|---|
| sipRegistrarStaticHost | Registrar server static IP address or domain name.<br>**Default Value**: 192.168.0.10 |
| sipRegistrarStaticPort | Registrar server static IP port number.<br>**Note**: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.<br>**Default Value**: 0 |

# Proxy Server

The proxy server is an intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is passed on to another entity that can further process the request. Proxies are also useful for enforcing policy and for firewall traversal. A proxy interprets, and, if necessary, rewrites parts of a request message before forwarding it.

In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

## Configuration Source

The Mediatrix 4102 must know the IP address and port number of the proxy server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

**Note:** When defining whether or not the Mediatrix 4102 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

▶ **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its proxy server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **dhcp**.

   You can query the proxy server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):

   - sipHomeDomainProxyHost
   - sipHomeDomainProxyPort

3. Set how you want to define the proxy server information in the DHCP server.

**Table 114:** Proxy Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *sipHomeDomainProxyDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to **0**. Set the proxy server IP address in the DHCP server inside the vendor specific sub-option 204 (hexadecimal 0xCC). |
| site specific code | The *sipHomeDomainProxyDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to any value between 128 and 254. Set the proxy server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *sipHomeDomainProxyDhcpSiteSpecificCode* variable in the unit's configuration). |

### Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its proxy server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 115:** Proxy Server Static Information

| Variable | Description |
|---|---|
| sipHomeDomainProxyStaticHost | Proxy server static IP address or domain name.<br>**Default Value**: 192.168.0.10 |
| sipHomeDomainProxyStaticPort | Proxy server static IP port number.<br>**Note**: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.<br>**Default Value**: 0 |

# Outbound Proxy Server

An outbound proxy is an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. The outbound proxy receives all outbound traffic and forwards it. Incoming traffic may or may not go through the outbound proxy. The outbound proxy's address is never used in the SIP packets, it is only used as a physical network destination for the packets.

When the outbound proxy is enabled, the proxy is still used to create the *To* and the *From* headers, but the packets are physically sent to the outbound proxy.

The outbound proxy is enabled if the IP address is valid (i.e., not 0.0.0.0). The default static value in the MIB is 0.0.0.0.

**SIP Outbound Proxy (From RFC 3261)**

A proxy that receives requests from a client, even though it may not be the server resolved by the Request-URI. Typically, a user agent is manually configured with an outbound proxy, or can learn about one through auto-configuration protocols.

When enabled, the initial route for all SIP requests containa the outbound proxy address, suffixed with the loose routing parameter "lr". The Request-URI still contains the home domain proxy address. Requests are directed to the first route (the outbound proxy).

## Configuration Source

The Mediatrix 4102 must know the IP address and port number of the outbound proxy. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

**Note:** When defining whether or not the Mediatrix 4102 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

**To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must ask for its outbound proxy settings through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **dhcp**.

   You can query the outbound proxy's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):

   - sipOutboundProxyHost
   - sipOutboundProxyPort

3. Set how you want to define the outbound proxy server information in the DHCP server.

**Table 116:** Outbound Proxy Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *sipOutboundProxyDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to **0**. Set the outbound proxy server IP address in the DHCP server inside the vendor specific sub-option 205 (hexadecimal 0xCD). |
| site specific code | The *sipOutboundProxyDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to any value between 128 and 254. Set the outbound proxy server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *sipOutboundProxyDhcpSiteSpecificCode* variable in the unit's configuration). |

### Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

**To use static information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must ask for its outbound proxy settings through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 117:** Outbound Proxy Static Information

| Variable | Description |
|---|---|
| sipOutboundProxyStaticHost | Static outbound proxy server IP address or domain name.<br>**Default Value**: 192.168.0.10 |
| sipOutboundProxyStaticPort | Static outbound proxy server IP port number.<br>**Note**: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.<br>**Default Value**: 0 |

▶ **To disable the outbound proxy:**

1. In the *ipAddressConfig* folder, set the *sipOutboundProxyStaticHost* variable to **0.0.0.0**.
   To re-enable the outbound proxy, enter a valid IP address.
   You can now specify if the outbound proxy uses a loose routing or strict routing type.

## Loose Router Configuration

| | |
|---|---|
| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol, section 6<br>RFC 2543 – SIP: Session Initiation Protocol |

You must specify the type of routing of the outbound proxy configured in *sipOutboundProxyHost* does.

**Note:** This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To set the outbound proxy router status:**

1. In the *sipMIB*, locate the *sipOutboundProxyConfig* variable.
   The following values are available:

**Table 118:** Outbound Proxy Router Status

| Value | Description |
|---|---|
| looseRouter | This is the most current method for SIP routing, as per RFC 3261, and will become the standard behaviour once RFC 3261 compliance is achieved. See "SIP Outbound Proxy (From RFC 3261)" on page 180 for details. |

### Loose Router

A proxy is said to be loose routing if it follows the procedures defined in the *RFC 3261* specification (section 6) for processing of the *Route* header field. These procedures separate the destination of the request (present in the Request-URI) from the set of proxies that need to be visited along the way (present in the *Route* header field). A proxy compliant to these mechanisms is also known as a loose router.

**Table 118:** Outbound Proxy Router Status (Continued)

| Value | Description |
|---|---|
| strictRouter | Pre-RFC 3261, RFC 2543 compatible SIP routing.<br>The initial route for all SIP requests contains the home domain proxy address (the Request-URI). Requests are directed to the outbound proxy.<br>In other words, the Request-URI is constructed as usual, using the home domain proxy and the user name, but is used in the route set. The Request-URI is filled by the outbound proxy address. |

# Presence Compositor Server

| | |
|---|---|
| **Standards Supported** | • RFC 3863 – Presence Information Data Format (PIDF)<br>• RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication |

The Presence Compositor server is a User Agent Server (UAS) that processes PUBLISH requests and is responsible for compositing event state into a complete, composite event state of a resource for a presentity.

The presence Compositor is enabled if the IP address is valid (i.e., not 0.0.0.0). The default static value in the MIB is 0.0.0.0.

## Configuration Source

The Mediatrix 4102 must know the IP address and port number of the Presence Compositor server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

**Note:** When defining whether or not the Mediatrix 4102 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

▶ **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its Presence Compositor server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **dhcp**.

   You can query the Presence Compositor server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):

   - sipPresenceCompositorHost
   - sipPresenceCompositorPort

3. Set how you want to define the Presence Compositor server information in the DHCP server.

**Table 119:** Presence Compositor Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *sipPresenceCompositorDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to **0**. Set the proxy server IP address in the DHCP server inside the vendor specific sub-option 206 (hexadecimal 0xCE). |
| site specific code | The *sipPresenceCompositorDhcpSiteSpecificCode* variable (under the *ipAddressConfigSipServer* group) to any value between 128 and 254. Set the Presence Compositor server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *sipPresenceCompositorDhcpSiteSpecificCode* variable in the unit's configuration). |

## Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether or not the Mediatrix 4102 must get its Presence Compositor server configuration through a DHCP server.

2. Set the *sipServerSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 120:** Presence Compositor Server Static Information

| Variable | Description |
|---|---|
| sipPresenceCompositorStaticHost | Presence Compositor server static IP address or domain name.<br>**Default Value**: 0.0.0.0 |
| sipPresenceCompositorStaticPort | Presence Compositor server static IP port number.<br>**Note**: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.<br>**Default Value**: 0 |

CHAPTER 11

# DNS SRV Configuration

This chapter describes the configuration required for the Mediatrix 4102 to work with a DNS SRV.

## What is a DNS SRV?

| | |
|---|---|
| **Standards Supported** | • RFC 2782 – A DNS RR for specifying the location of services (DNS SRV)<br>• RFC 3263 - Session Initiation Protocol (SIP): Locating SIP Servers |

Currently, one must either know the exact address of a server to contact it, or broadcast a question.

DNS SRV is an extension of the standard DNS server. SRV (Service Record) is a type of entry a network administrator may put into the DNS answers. A DNS SRV is used to get one or more IP addresses of servers, each one having its own weight and priority.

Each server received when using DNS SRV, depending on its weight and priority, can be used as a primary or backup server or can be part of a load balancing system.

For instance, the client requests the SRV for SIP servers in some domain. The DNS server may return the A, B, and C addresses, which are all SIP servers. Each address has a weight and the client must choose one of those three addresses by using a random algorithm that considers the weight.

To use DNS SRV, an administrator must set a service records (SRV) into the DNS servers available on the network.

DNS SRV implementation should imply a shared database between servers since a REGISTER and an INVITE can be sent to any server, not necessarily the same one.

DNS SRV applies to both TCP and UDP transport types.

### Priority vs Weight

A DNS SRV uses the *priority* and *weight* concepts to distribute the requests.

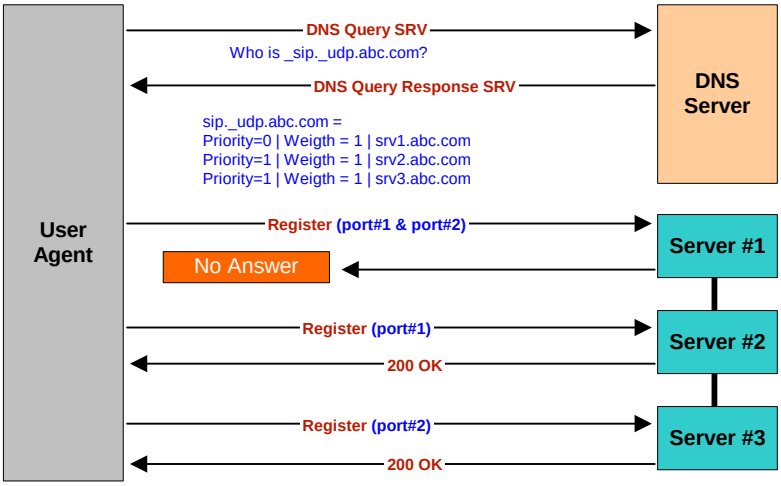
**Table 121:** Priority vs. Weight

| Parameter | Description |
|---|---|
| Priority | A client must attempt to contact the target host with the lowest-numbered priority it can reach. |
| Weight | Specifies a relative weight for entries with the same priority. Larger weights should be given a proportionately higher probability of being selected. |

### DNS SRV Call Flow

The following is a standard DNS SRV call flow:

**Figure 84:** DNS SRV Call Flow




# Enabling DNS SRV on the Mediatrix 4102

If the address of a service corresponds to a domain name that is bound to a SRV record, the port this service uses must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. See "Chapter 10 - SIP Servers" on page 177 for more details.

▶ **To enable DNS SRV:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).

   This variable defines whether the Mediatrix 4102 must get its proxy server configuration through a DHCP server or not.

2. Set the *sipServerSelectConfigSource* variable to **static**.

3. Set one or more of the following variables to **0**:

**Table 122:** Variables to Enable DNS SRV

| Server | Variable to enable |
|---|---|
| SIP Registrar server | sipRegistrarStaticPort |
| SIP Proxy server | sipHomeDomainProxyStaticPort |
| SIP Outbound Proxy server | sipOutboundProxyStaticPort |

**Note:** Any "SRV enabled" service must have a host name recognized by the DNS SRV server. "*_sip._udp*" or "*_sip._tcp*" (depending on the transport type) is automatically added to the host name.

# DNS SRV Record Lock

You can configure the Mediatrix 4102 to always use the same DNS SRV record for a SIP call ID. As a result, a call or registration always uses the same destination until the destination is unreachable or the unit receives a different DNS SRV result.

▶ **To enable the DNS SRV record lock feature:**

1. In the *sipInteropMIB*, set the *sipInteropLockDnsSrvRecordPerCallEnable* variable to **enable**.

   All messages during a call or registration use the same SRV record.

   If you set this variable to **disable** (which is the default value), the Mediatrix 4102 rather follows the behaviour as described in RFC 3263.

2. Restart the Mediatrix 4102 so that the changes may take effect.

# DNS SRV-Oriented Settings

The following parameters have an effect on the DNS SRV behaviour.

**Table 123:** DNS SRV-Oriented Settings

| Parameter | Description |
|---|---|
| sipInteropTransmissionTimeout | • Has a dramatic effect should a server time out, since a default 32 s delay would be introduced at every call.<br>• Media5 recommends a maximum of 2-3 s when using DNS SRV.<br>• See "Transmission Timeout" on page 301 for more details. |
| sipPenaltyBoxTime | If *sipPenaltyBoxEnable* is set to **enable**:<br>• A "timed out" server is considered "not responding" for this amount of time.<br>• Can be seen as the time it will take to retry a server that failed responding.<br>• See "SIP Penalty Box" on page 295 for more details. |
| sipInteropReuseCredentialEnable | If *sipInteropReuseCredentialEnable* is set to **enable**:<br>• If there is not a shared database between servers, this could lead to authentication problems because a REGISTER and an INVITE can be sent to any server, not necessarily the same one.<br>• See "SIP Credential" on page 305 for more details. |

CHAPTER 12

# Country-Specific Configuration

This chapter describes how to set the Mediatrix 4102 with the proper country settings.

## Caller ID Information

The caller ID is a generic name for the service provided by telephone utilities that supply information such as the telephone number or the name of the calling party to the called subscriber at the start of a call. In call waiting, the caller ID service supplies information about a second incoming caller to a subscriber already busy with a phone call. However, note that caller ID on call waiting is not supported by all caller ID-capable telephone displays.

In typical caller ID systems, the coded calling number information is sent from the central exchange to the called telephone. This information can be shown on a display of the subscriber telephone set. In this case, the caller ID information is usually displayed before the subscriber decides to answer the incoming call. If the line is connected to a computer, caller information can be used to search in databases and additional services can be offered.

The following basic caller ID features are supported:

- Date and Time
- Calling Line Identity
- Reason for Absence of Calling Line Identity
- Calling Party Name
- Reason for Absence of Calling Party Name
- Visual Indicator (MWI)

### Caller ID Generation

There are two methods used for sending caller ID information depending on the application and country-specific requirements:

- caller ID generation using DTMF signalling
- caller ID generation using Frequency Shift Keying (FSK)

Both methods can be used on different lines at the same time.

The displayed caller ID for all countries may be up to 20 digits for numbers and 50 digits for names.

#### DTMF Signalling

The data transmission using DTMF signalling is performed during or before ringing depending on the country settings or line configuration. The Mediatrix 4102 provides the calling line identity according to the following standards:

- Europe: ETSI 300 659-1 January 2001 (Annex B) : Access and Terminals (AT); Analogue access to the Public Switched Telephone Network (PSTN); Subscriber line protocol over the local loop for display (and related) services; Part 1: On-hook data transmission.
- Brazil: STD 220-250-713 Issue 01. November 1993: General specification “identification of the calling party for SPC with DTMF”.

**Note:** For units in Brazil, set the *analogScnGwInterDigitDialDelay* and *analogScnGwDtmfDuration* value to **70** ms (in the *analogScnGwMIB*). This will ensure that the caller ID displays properly.

- Denmark: TDK-TS 900 301-1 January 2003: Public Switched Telephone Network (PSTN)

Calling Line Identification presentation (CLIP) supplementary service Specification of the NTP.

### FSK Generation

Different countries use different standards to send caller ID information. The Mediatrix 4102 is compatible with the following widely used standards:

- Bellcore GR-30-CORE
- British Telecom (BT) SIN227, SIN242
- UK Cable Communications Association (CCA) specification TW/P&E/312
- ETSI 300 659-1

**Note:** The compatibility of the Mediatrix 4102 is not limited to the above caller ID standards.

Continuous phase binary FSK modulation is used for coding which is compatible with:

- BELL 202
- ITU-T V.23, the most common standard

## ADSI

ADSI (Analog Display Service Interface) is a telecommunications protocol standard that enables alternate voice and data capability over the existing analog telephone network. It is an extension to basic caller ID. To use ADSI, you would need an ADSI capable device.

ADSI can display the basic caller ID parameters and the following additional parameters:

- Call Type
- First Called Line Identity
- Number of Messages (MWI)
- Type of Forwarded Call
- Type of Calling User
- Redirecting Number
- Charge
- Duration of the Call
- Network Provider Identity

**Note:** Currently, very few ADSI-capable devices support these additional information.

# Setting the Location (Country)

It is very important to set variables according to the country in which the Mediatrix 4102 is used because a number of parameter values are set according to this choice. These parameters are:

- Tones
- Rings
- Impedances
- Line Attenuations

See "Appendix D - Country-Specific Parameters" on page 411 for more information on these country-specific settings.

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.t

You can also set these parameters via the web interface, as described in "Country Selection" on page 117.

▶ **To set a country location:**

1. In the *telephonyMIB,* locate the *telephonyCountrySelection* variable.

   This variable indicates the current country used by the Mediatrix 4102. It can also be used to select a caller ID standard in countries that support more than one caller ID standard.

2. Set the variable with one of the following parameters:

| | | |
|---|---|---|
| North America 1 | Australia 2 | New Zealand |
| North America 2 | Australia 3 | UAE 2 (United Arab Emirates) |
| Austria 1 | Japan | Czech Republic |
| Austria 2 | Israel | Chile1 |
| France | Thailand | Chile2 |
| Germany 1 | Indonesia | South Africa |
| Germany 2 | China | uk-bellcore |
| Germany 3 | Hong Kong | uk-cca |
| UK | Malaysia | uk-etsi-fsk |
| Italy | Russia | france-etsi-fsk |
| Spain | Netherlands | france-etsi-dtmf |
| Switzerland | Brazil | austria-etsi-fsk |
| Sweden | Mexico | austria2-etsi-fsk |
| Australia 1 | Denmark | |

3. Restart the Mediatrix 4102 so that the changes may take effect.

## Caller ID Selection

In countries that support more than one caller ID standard, this standard can be selected with the *telephonyCountrySelection* variable. Be careful to properly select the option corresponding to your caller ID.

**Table 124:** Caller ID Mappings

| Country | Caller ID | *telephonyCountrySelection* variable Mapping |
|---|---|---|
| UK | British Telecom | uk |
| | Bellcore | uk-bellcore |
| | CCA | uk-cca |
| | ETSI-FSK | uk-etsi-fsk |
| France | Bellcore | france |
| | ETSI-FSK | france-etsi-fsk |
| | ETSI-DTMF | france-etsi-dtmf |
| Austria 1 | Bellcore | austria1 |
| | ETSI-FSK | austria-etsi-fsk |
| Austria 2 | Bellcore | austria2 |
| | ETSI-FSK | austria2-etsi-fsk |

See "Caller ID Information" on page 189 for more details.

# Custom Tone Configuration

You can override the pattern for a specific tone defined for the selected country (see "Appendix D - Country-Specific Parameters" on page 411 for more details). You can define new patterns for the following tones:

- Busy
- Confirmation
- Congestion
- Dial
- Intercept
- Message Waiting
- Preemption
- Reorder
- Ringback
- Receiver Off Hook (ROH)
- Special Information Tone (SIT)
- Stutter

## Pattern Definition

The general format of the pattern string is defined in the following ABNF:

```
tone-pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

This general pattern uses the following three main categories

```
frequencies-section = "f1=" frequency-description
                      [ ",f2=" frequency-description
                        [ ",f3=" frequency-description
                          [ ",f4=" frequency-description ] ] ]

loop-counter-section = "l=" loop-counter

states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description
                     [ ",s4=" state-description
                       [ ",s5=" state-description
                         [ ",s6=" state-description
                           [ ",s7=" state-description
                             [ ",s8=" state-description ] ] ] ] ] ] ]
```

Finally, the three main categories use the following parameters and tags:

```
frequency-description = frequency ":" power
frequency = 2*4DIGIT
power = ( DIGIT / "-" 1*2DIGIT )
loop-counter = 1*2DIGIT
state-description = ( on-state-description / off-state-description )
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [ next-state
]
off-state-description = "off" [ time ][ loop-indicator ][ next-state]
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
time = ":" 2*5DIGIT
loop-indicator = ":l"
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

The following table describes the various tags used in the syntax.

**Table 125:** Pattern Definition Syntax

| Tag | Description |
|---|---|
| `tone-pattern` | String describing the pattern to use for the tone. An empty string means no tone. |
| `frequencies-section` | Description of the frequencies used by the tones used in `states-section`. You can define up to four frequencies (f1 to f4). You must enter at least one frequency if the `tone-pattern` is not empty. The frequencies to use are defined in the `state-description`. |
| `frequency-description` | Description of the frequency. This is described as "`frequency`:`power`". |
| `frequency` | Frequency value in Hz. The range is from 10 Hz to 4000 Hz. |
| `power` | Power level of the frequency in dBm. The range is from -99 dBm to 3 dBm. |
| `loop-counter-section` | Loop counters definition. The loop counter is used in `state-description`. |
| `loop-counter` | Value of the loop counter. The range is from 2 to 128. |
| `states-section` | Description of the tone state. You can define up to eight states (s1 to s8). You must enter at least one state if the `tone-pattern` is not empty. |
| `state-description` | Description of the tone state. |
| `on-state-description` | Description of a state playing a tone. |
| `off-state-description` | Description of a state not playing a tone. |
| `frequency-selection` | Frequency to play in the state. You can use from one to four frequencies. The frequency must be defined in `frequencies-section`. |
| `time` | The number of times, in ms, to perform the action of the state. The range is from 10 ms to 56000 ms. The tone stays indefinitely in the state if no time is specified. |
| `loop-indicator` | Used to stop looping between states after a number of loops defined in `loop-counter-section`. When the number of loops is reached, the next state is s(n+1) for the state s(n) instead of the state defined in `next-state`. |
| `next-state` | The next tone state to use when the time has elapsed. This value is not present if the time is not present. |

## Customizing the Tones

The *Custom Tone* section allows you to define new patterns as per the pattern syntax.

▶ **To customize one or more tones:**

1. In the *telephonyMIB,* locate the *countryCustomizationToneTable* table.
2. Define whether or not you want to override the default tone configuration for a specific tone by setting the *countryCustomizationToneOverride* variable.
3. Enter the override pattern in the corresponding *countryCustomizationToneTone* variable.

   You must follow the syntax as described in "Pattern Definition" on page 192.

   See "Custom Tone Example" on page 194 for a detailed example on how to create a proper pattern.

The following table gives some examples of custom tones. Note that the quotation marks are not part of the syntax and must not be included when entering the tone pattern.

**Table 126:** Pattern Examples

| Example | Pattern |
|---|---|
| No tone | `""` |
| North America dial tone (continous tone at 350 Hz and 440 Hz with a -17 power level) | `"f1=350:-17,f2=440:-17,s1=on:f1:f2"` |
| North America Recall dial tone (three quick tones followed by a continuous tone) | `"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"` |
| Australia ring back tone (tone on 400 ms, off 200 ms, on 400 ms, and off 2000 ms and replay) | `"f1=425:-17,f2=400:-5,f3=450:-5,s1=on:f1:f2:f3:400:s2,s2=off:200:s3,s3=on:f1:f2:f3:400:s4,s4=off:2000:s1"` |

## Custom Tone Example

This section describes how to create the pattern for the North America recall dial tone (also called stutter dial tone), which is three quick tones followed by a continuous tone.

```
"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
```

▶ **To create the pattern:**

1. Let's start with the general format of the pattern string:

```
Pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

2. Set the `frequencies-section` category, which is defined as follows:

```
frequencies-section = "f1=" frequency-description
                      [ ",f2=" frequency-description
                        [ ",f3=" frequency-description
                          [ ",f4=" frequency-description ] ] ]
```

*a.* The `frequency-description` parameter is described as follows:

```
frequency:power
```

*b.* In the North America stutter dial tone, two frequencies are used, 350 Hz and 440 Hz. Their power level is -17 dBm. You can thus complete the `frequencies-section` category as follows:

```
frequencies-section = "f1=" 350:-17 ",f2=" 440:-17 ]
```

*c.* The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," loop-counter-section ] "," states-section ]
```

3. Set the `loop-counter-section` category, which is defined as:

```
loop-counter-section = "l=" loop-counter
```

It defines the number of times to repeat the pattern.

*a.* The loop-counter part is defined as follows:

```
loop-counter = 1*2DIGIT
```

*b.* In the North America stutter dial tone, the pattern is repeated three times, thus:

```
loop-counter = 3
```

*c.* The `loop-counter-section` category now looks as follows:

```
loop-counter-section = "l=" 3
```

*d.* The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," states-section ]
```

**4.** Set the `states-section` category, which is defined as:

```
states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description
                     [ ",s4=" state-description
                       [ ",s5=" state-description
                         [ ",s6=" state-description
                           [ ",s7=" state-description
                             [ ",s8=" state-description ] ] ] ] ] ] ]
```

*a.* `state-description` is defined as:

```
state-description = ( on-state-description / off-state-description )
```

*b.* There are three states in the North America stutter dial tone: 0.1 on, 0.1 off, and continuous. The pattern that must be described is thus:

```
states-section = ",s1=" state-description
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

**5.** Let's define the first state. Since the first state describes an on tone, `off-state-description` is not required for this state.

*a.* You now have to complete the `on-state-description` parameter for the first state, which is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

*b.* `frequency-selection` is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

*c.* The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] ] [ time] [ loop-indicator ]
[ next-state ]
```

*d.* The `time` parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The on state is 100 ms, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ loop-indicator ] [
next-state ]
```

*e.* The `loop-indicator` parameter is not used in this state. The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ next-state ]
```

*f.* The `next-state` parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the `next-state` parameter is the off state, which is designated as `s2`.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ]
```

*g.* You can now complete the first `state-description` parameter:

```
states-section = ",s1=" ( on-state-description)
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" state-description
                   [ ",s3=" state-description ] ]
```

**6.** Let's define the second state. Since the first state describes an off tone, `on-state-description` is not required for this state.

*a.* You now have to complete the `off-state-description` parameter for the first state.

The `off-state-description` parameter is defined as:

```
off-state-description = "off" [ time ][ loop-indicator ][ next-state]
```

*b.* The `time` parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The off state is 100 ms, thus,

```
off-state-description = "off" [ :100 ][ loop-indicator ][ next-state]
```

*c.* The `loop-indicator` parameter is defined as:

```
loop-indicator = ":l"
```

It is used to stop looping between states. It indicates that the loop stops after three times. Once the loop is completed, the pattern goes to the next state (which is state 3).

The `off-state-description` parameter is now:

```
off-state-description = "off" [ :100 ] [ :1 ][ next-state]
```

*d.* The `next-state` parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the `next-state` parameter is the on state, which is designated as `s1`.

The `off-state-description` parameter is now:

```
off-state-description = "off" [ :100 ] [: 1 ][ :s1]
```

*e.* You can now complete the second `state-description` parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                  [ ",s2=" state-description
                    [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [100] [ s2 ])
                  [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                    [ ",s3=" state-description ] ]
```

**7.** Let's define the third and last state. Since the third state describes an on tone, `off-state-description` is not required for this state.

*a.* You now have to complete the `on-state-description` parameter for the first state.

The `on-state-description` parameter is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

*b.* `frequency-selection` is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

*c.* The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ time] [ loop-indicator ] [
next-state ]
```

*d.* The `time` parameter is the number of milliseconds to perform the action of the state. Since the third state is a continuous tone, the `time` parameter is not required, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ loop-indicator ] [ next-
state ]
```

*e.* The `loop-indicator` parameter is used to stop looping between states. Since the third state is a continuous tone and does not use loops, this parameter is not required.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ next-state ]
```

*f.* The `next-state` parameter is the next tone state to use when the time has elapsed. This value is not present if the `time` parameter is not present. You have already discarded the `time` parameter, so the `next-state` parameter is not required.

The `on-state-description` parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ]
```

*g.* You can now complete the third `state-description` parameter and the `states-section` parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ]
```

**8.** Now that you have the three main categories completed, you can finish the pattern:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," ",s1=" ( "on" [ ":f1" ]
[ ":f2" ] [ :100 ] [ :s2 ])
                 [ ",s2=" "off" [ :100 ] [ :1 ][ :s1]
                   [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ] ]
```

If you remove all the brackets and quotation marks, which are not to be included, the pattern is:

```
Pattern = f1=350:-17,f2=440:-
17,l=3,s1=on:f1:f2:100:s2,s2=off:100:1:s1,s3=on:f1:f2
```

Ther pattern could also be defined as follows:

```
Pattern = f1=350:-17,f2=440:-
17,s1=on:f1:f2:100:s2,s2=off:100:s3,s3=on:f1:f2:100:s4,s4=off:100:s5,s5=on:f
1:f2:100:s6,s6=off:100:s7,s7=on:f1:f2
```

CHAPTER 13

# Transparent Address Sharing

This chapter explains how to properly configure the Transparent Address Sharing service for a cable or DSL modem.

| | |
|---|---|
| **Standards Supported** | • RFC 1027 – Using ARP to Implement Transparent Subnet Gateways (section 2.14.3.3.8 Processing of ARP messages). |

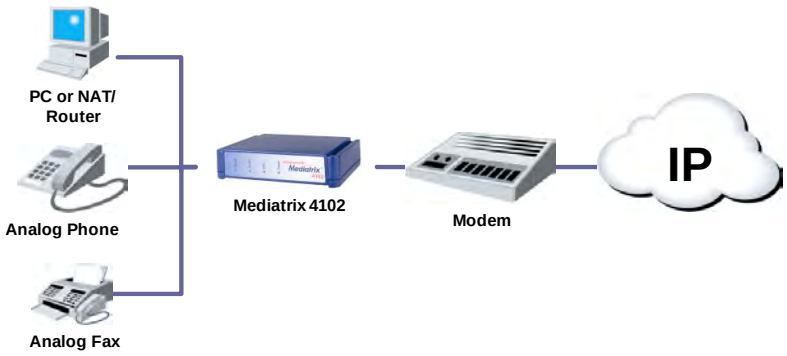
## What is Transparent Address Sharing?

When in Transparent Address Sharing (TAS) mode, the Mediatrix 4102 shares a Wide Area Network (WAN) connection with a single PC or a network of IP equipment. The unit routes IP packets between the LAN and the network providing the public address (WAN).

The Media5 patent on transparent IP address sharing allows both WAN and LAN interfaces to be used with a single IP address from the service provider in a user-friendly way, without the configuration complexities of an integrated NAT.

The Mediatrix 4102 thus connects up to two analog phones or fax machines to a broadband access equipment, allowing Service Providers to offer IP telephony services to residential users.

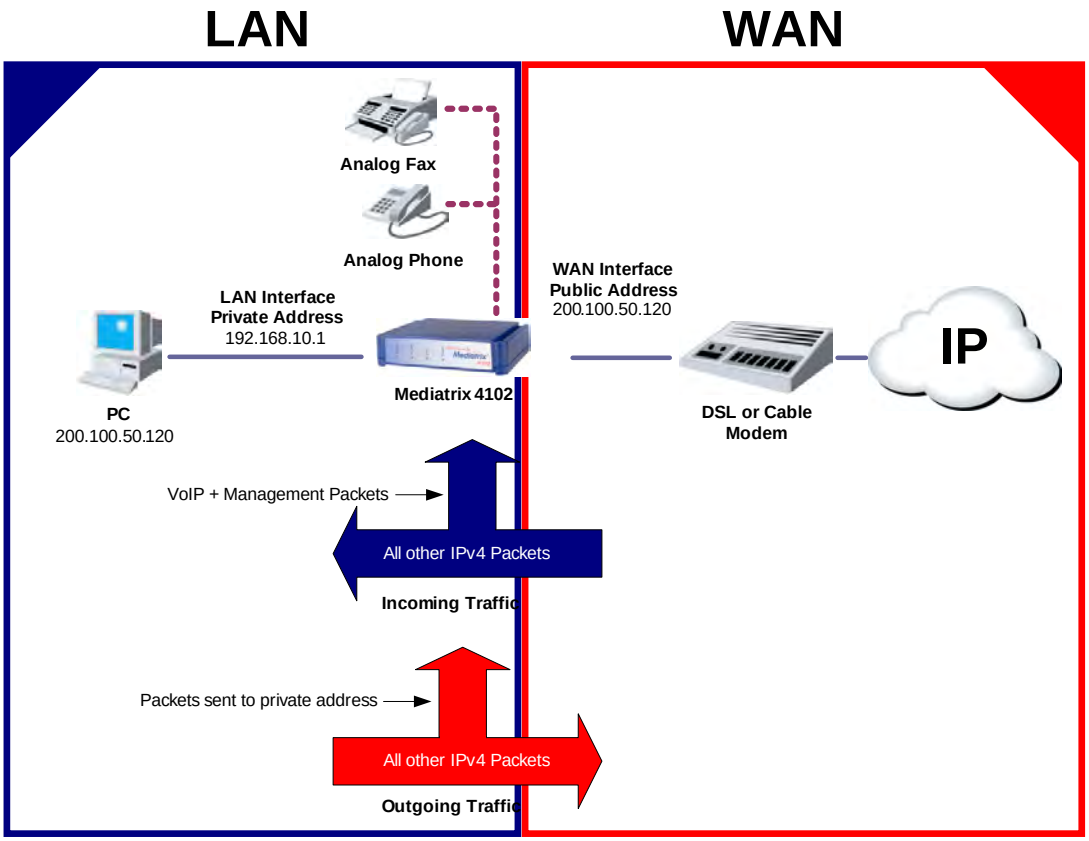
**Figure 85:** Mediatrix 4102 Residential Application Scenario




In a residential VoIP deployment, the WAN interface of the Mediatrix 4102 gets assigned a public IPv4 address by the ISP, either by a DHCP negotiation, by establishing a point-to-point link (PPPoE), or by some other mechanism depending on the type of link.

The device on the LAN (e.g., the PC) gets assigned the same public IPv4 address as the Mediatrix 4102. The subnet mask given is the same as the one assigned by the ISP, or if it is not available (such as for a PPPoE connection), it uses the predefined subnet classes.

**Figure 86:** Transparent Address Sharing




The LAN interface of the Mediatrix 4102 is configured with a private IPv4 address. This address allows the device on the LAN to communicate with the Mediatrix 4102 as this would be otherwise impossible because both devices share the same public IPv4 address. The Mediatrix 4102 performs transparent routing by forwarding to the WAN any packet sent to any IPv4 address that is included in the public subnet.

Each packet received from the WAN is forwarded directly to the device on the LAN, except if it belongs to the hosted application, in this case, VoIP. In the other direction, each IPv4 packet received from the LAN is forwarded to the WAN, except for packets sent explicitly to the private address assigned to the Mediatrix 4102. The Mediatrix 4102 itself can initiate a communication with the device on the LAN, by using its private IPv4 address as the source address.

## Router Mode

The router mode separates the two external interfaces of the Mediatrix 4102 (the *LAN* and *WAN* connectors). The Mediatrix 4102 has two distinct network interfaces with one IP address for each of them. These interfaces are called LAN (*LAN* connector) and WAN (*WAN* connector). The Mediatrix 4102 performs IP routing both ways.

The router mode is a requirement for the TAS to properly work. When enabling TAS, you also enable the Mediatrix 4102 in router mode.

The router mode is also a requirement for the Bandwidth Control feature ("WAN Upstream Bandwidth Control" on page 208).

In non-router mode, the interfaces are switched and the Mediatrix 4102 only has one network interface. Both external network interfaces generally have the same behaviour.

## Cable vs DSL Modem

Most of the Mediatrix 4102 settings are the same no matter what the modem you are using. However, there are a few differences.

**Table 127:** Cable vs DSL Modem

| Cable Modem | DSL Modem |
|---|---|
| You must configure the Mediatrix 4102 to use a DHCP server to get its IP information as per "DHCP Server Configuration" on page 168.<br>However, some locations may require to manually enter static IP information instead. The easiest way to do so is to use the web interface. See "Chapter 2 - Web Interface – Introduction" on page 25 for more details. | You must configure the PPPoE service as per "PPPoE Service" on page 202.<br>However, some DSL modems may require that you configure the Mediatrix 4102 to use a DHCP server to get its IP information as per "DHCP Server Configuration" on page 168. |

## Multicast and IGMP

The Mediatrix 4102 does not support the Internet Group Management Protocol (IGMP), i.e., the PC connected to the *LAN* connector of the Mediatrix 4102 cannot register to IGMP services.

Multicast is communication between a single sender and multiple receivers on a network. IGMP is an Internet protocol that provides a way for an Internet computer to report its multicast group membership to adjacent routers. Multicasting allows one computer on the Internet to send content to multiple other computers that have identified themselves as interested in receiving the originating computer's content.

## Configuration Steps

The following are the steps to follow to properly setup the TAS service.

▶ **To setup the TAS service:**

1. Define the PPPoE parameters as described in "PPPoE Service" on page 202. **This only applies if you are using a DSL modem.**
2. Define the WAN IP address configuration source as described in "WAN Information Configuration Source" on page 204.
3. Configure the TAS mechanism as described in "Configuring TAS" on page 205.
4. Optionally, modify port allocation settings as described in "Ports Settings" on page 210.
5. Restart the Mediatrix 4102 so that the changes may take effect.

# PPPoE Service

| Standards Supported | |
|---|---|
| | • RFC 1332 – IP Control Protocol (IPCP)<br>• RFC 1661 – Point to Point Protocol (PPP)<br>• RFC 1334 – Password Authentication Protocol (PAP)<br>• RFC 1994 – Challenge Handshake Authentication Protocol (CHAP)<br>• RFC 2516 – PPP over Ethernet (PPPoE)<br>• RFC 1471 – PPP Link control Protocol MIB (PPP-LCP-MIB) (with the exception of the LQR MIB)<br>• RFC 1472 – PPP Security Protocols MIB (PPP-SEC-MIB)<br>• RFC 1473 – PPP IP Network Control Protocol MIB (PPP-IP-NCP-MIB)<br>• RFC 1877 – PPP IPCP Extentions for Name Server Address – with the exception of sections 1.2 and 1.4 |

The Mediatrix 4102 uses the PPPoE protocol to interact with a DSL broadband modem. It can discover a PPP access concentrator (AC) and establish a PPP session with it.

**Note:** This section applies only if you are using a DSL modem. If you are using a cable modem, go directly to "WAN Information Configuration Source" on page 204.

The PPPoE service is required to properly use the TAS service with DSL modems. You must perform the following tasks to configure the PPPoE service:

- Enable the PPPoE service.
- Set a user name and password.

## Enabling the PPPoE Service

You must configure and enable the service to properly connect to an access concentrator.

You can also use the web interface to enable the PPPoE service. See "WAN Page" on page 35 for more details.

**To configure the PPPoE service:**

1. In the *ipAddressConfig* folder, set the *localHostSelectConfigSource* variable (under the *ipAdressConfigLocalHost* group) to **static**.

   This is required to avoid conflicts with the PPPoE interface. See "Local Host" on page 158 for more details on the local host settings.

2. Set the *localHostStaticDefaultRouter* variable (under the *ipAddressConfigLocalHostStatic* group) to **0.0.0.0**.

3. In the *pppoeMIB*, set the *pppoeAcName* variable with the name of the access concentrator to which connect.

   The variable may be set with any string of characters, with a maximum of 255 characters.

### PPPoE

PPPoE (Point to Point Protocol over Ethernet) is a proposal specifying how a host personal computer interacts with a DSL broadband modem to access the growing number of Highspeed data networks. Relying on two widely accepted standards, Ethernet and the point-to-point protocol (PPP), the PPPoE implementation requires virtually no more knowledge on the part of the end user other than that required for standard Dialup Internet access. In addition, PPPoE requires no major changes in the operational model for Internet Service Providers (ISPs) and carriers. The base protocol is defined in RFC 2516.

If you leave this variable empty, the Mediatrix 4102 accepts the first offer that it receives.

4. Set the *pppoeServiceName* variable with the name of the service requested to the access concentrator.

   The variable may be set with any string of characters, with a maximum of 255 characters.

   If you leave this variable empty, the Mediatrix 4102 looks for any access concentrator.

5. Enable the PPPoE service by setting the *pppoeEnable* variable to **enable**.

**Note:** When establishing a PPPoE connection, the default router IP address is automatically overridden by the PPP connection's peer IP address.

6. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

## Setting a User Name and Password

When connecting to an access concentrator, it usually requests that the Mediatrix 4102 identifies itself with a specific user name and password, also called ID and secret pair. This information is set in the standard *PPP-SEC-MIB* as described in RFC 1472.

You can also use the web interface to enter the PPPoE user name and password. See "WAN Page" on page 35 for more details.

▶ **To configure a user name and password:**

1. In the *PPP-SEC-MIB*, locate the *pppSecuritySecretsTable*.

   This table contains the ID (user name) and secret (password) pair that the Mediatrix 4102 advertises to the access concentrator.

**Caution:** When you are in a specific row and you set the *pppSecuritySecretStatus* variable of this row to **invalid**, the row is deleted and you cannot add it back. When the last remaining row is deleted, two rows are re-created with default passwords and user names.

2. Set the Identity (user name) of the ID/Secret pair in the *pppSecuritySecretsIdentity* variable.

   You can set an identity for both security protocols (CHAP and PAP) supported. The table contains one row for each protocol. The CHAP security protocol is described in RFC 1994, while the PAP security protocol is described in RFC 1334. If you want to exclusively use one of the security protocols, you can disable the other row.

3. Set the secret (password) of the ID/Secret pair in the *pppSecuritySecretsSecret* variable.

   You can set an identity for both security protocols (CHAP and PAP) supported. The table contains one row for each protocol. The CHAP security protocol is described in RFC 1994, while the PAP security protocol is described in RFC 1334. If you want to exclusively use one of the security protocols, you can disable the other row.

   For other standard settings, please refer to the MIBs described in RFC 1471 (with the exception of the LQR MIB), RFC 1472, and RFC 1473.

4. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

# WAN Information Configuration Source

The WAN address is the public IP address attributed to the Mediatrix 4102. This address is used for incoming signalling, media and management traffic. You can assign this information to the Mediatrix 4102 through an access concentrator (DSL modem) or DHCP server (cable modem).

**To set the WAN IP address configuration source:**

1. In the *ipAddressConfig* folder, locate the *localHostWanAddressSelectConfigSource* variable (under the *ipAddressConfigLocalHost* group).

   This variable indicates the source to be used for the provisioning of the WAN address.

   **Table 128:** WAN IP Address Source Settings

| Option | Description |
|---|---|
| localAddress | The WAN address is the one that is set in the *localHostAddress* variable, whereas the *localHostStaticWanAddress* is ignored. |
| static | The Mediatrix 4102 has a static WAN address. The address is configured in the *localHostStaticWanAddress* variable. Note that this setting allows a limited NAT traversal scheme. |
| pppoe | The Mediatrix 4102 uses PPP over Ethernet in order to obtain its WAN IP address.<br>**Note**: The PPPoE service must be enabled for the WAN address to be configured. |
| automatic | If the PPPoE service is enabled, the Mediatrix 4102 uses *pppoe* as the configuration source. Otherwise it uses *localAddress*. |

2. Set the *localHostWanAddressSelectConfigSource* variable to **automatic**.

   You can query the actual configuration source used in the *localHostWanAddressConfigSource* read-only variable (in the *ipAddressStatus* folder).

   You can query the actual WAN IP address attributed to the Mediatrix 4102 in the *localHostWanAddress* read-only variable (in the *ipAddressStatus* folder).

3. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

# Configuring TAS

The following steps describe how to properly setup and enable the Mediatrix 4102 in TAS mode. By enabling TAS, you also implicitly enable the Mediatrix 4102 in router mode (see "Router Mode" on page 200 for more details).

When the TAS service is enabled, it adds a network interface to the Mediatrix 4102. This means that two IP addresses may be attributed to the unit – the *LAN* and *WAN* connectors of the Mediatrix 4102 each have an IP address.

The *LAN* and *WAN* connectors of the Mediatrix 4102 are allocated IP addresses differently depending on the scenario involved:

- **TAS disabled**

  The *LAN* and *WAN* connectors use the same IP address as set in the *localHostAddress* variable. See "Local Host" on page 158 for more details.

- **TAS enabled - cable modem (PPPoE disabled)**

  The *WAN* connector receives an IP address to access the WAN. This IP address is coming from the DHCP server and is stored in the *localHostAddress* variable. See "Local Host" on page 158 for more details.

  The IP address of the *LAN* connector (LAN interface) is configured statically, as described in "LAN Interface" on page 206.

  **Figure 87:** Cable Modem IP Addresses

  For example:

  - The Mediatrix 4102 is set up to get a WAN address via DHCP.
  - The local host address is set to the DHCP-derived address (let's say 10.40.40.53).
  - The LAN side address is also 10.40.40.53. This is consistent with the TAS feature.

  The *lanStaticAddress* variable is set by default to 192.168.10.1.

  In summary:

  *a.* WAN is good.

  IP address 10.40.40.53 on both sides, WAN and LAN.

  The web page of the Mediatrix 4102 is at 192.168.10.1.

  *b.* WAN goes down.

  The IP address 10.40.40.53 is not available on the WAN side.

  The web page of the Mediatrix 4102 is still at 192.168.10.1.

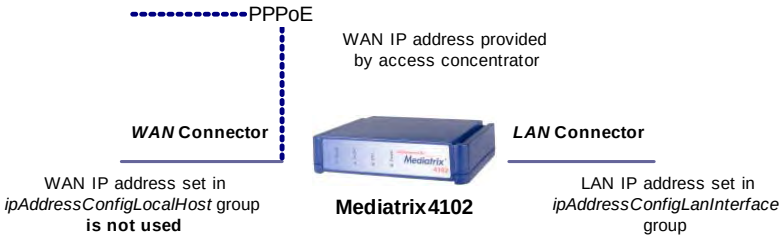
  The IP address of the *LAN* connector is the IP address of the Mediatrix 4102 plus or minus 1, i.e., the IP address of the *LAN* connector is either 192.168.10.0 or 192.168.10.2.

- **TAS enabled - DSL modem (PPPoE enabled)**

  The *WAN* connector receives an IP address to access the WAN. This IP address is coming from the access concentrator and is stored in the *localHostWanAddress* variable. In this case, the IP address in the *localHostAddress* variable is not used.

  The IP address of the *LAN* connector (LAN interface) is configured statically, as described in "LAN Interface" on page 206.

**Figure 88:** DSL Modem IP Addresses (PPPoE)




## QoS Differentiated Services Fields

If you want to differentiate the packets sent by the PC from the packets sent by the Mediatrix 4102, you must substitute a configured value for the QoS Differentiated Services fields of the packets sent from the PC (routed from the LAN to the WAN interface).

The goal is to prioritize the Mediatrix 4102's packets over the PC's packets. This allows to offer a quicker response time for the voice, for instance. However, the ISP's network must support QoS routing. It must be configured accordingly and route according to the DiffServ it encounters.

See "Differentiated Services (DS) Field" on page 365 for more details.

▶ **To configure a substitution value:**

1. In the *ipRoutingMIB*, set the substitution value in the *ipRoutingQosDiffServSubstitution* variable.

   This substitution value is used for the Differentiated Services fields of IP packets originating from the equipment connected to the LAN interface and routed to the WAN. It supersedes the TOS byte of the packets coming from the PC. The default value is **0**.

2. Set the *ipRoutingQosDiffServSubstitutionEnable* variable to **enable**.

   The *DiffServ* field of IP packets originating from the equipment connected to the LAN interface and routed to the WAN is modified.

3. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

## LAN Interface

The Mediatrix 4102's two Ethernet connectors are used as follows:

- *WAN* connector: WAN interface of the Mediatrix 4102 where you can connect your modem.
- *LAN* connector: LAN interface where you connect the PC or other IP equipment. The LAN interface is usually used to connect a PC that will have access to the WAN by sharing the Mediatrix 4102 WAN address.

**Note:** The LAN interface configuration settings are valid only when TAS is enabled.

You can also use the web interface to configure the LAN interface parameters. See "LAN Page" on page 37 for more details.

▶ **To configure the LAN interface settings:**

1. In the *ipAddressConfig* folder, locate the *ipAddressConfigLanInterface* group.

2. Set the LAN IP address of the *LAN* connector in the *lanStaticAddress* variable.

   If TAS is disabled, the *LAN* connector rather uses the same address as the *WAN* connector as set in the *localHostAddress* variable. See "Local Host" on page 158 for more details.

**Note:** Do not set the *lanStaticAddress* variable to 0.0.0.0. This could prevent the unit from properly sending a DHCP discover request.

3. Set the LAN subnet mask of the *LAN* connector in the *lanStaticNetworkMask* variable.

   If TAS is disabled, the *LAN* connector rather uses the same network mask as the *WAN* connector as set in the *localHostNetworkMask* variable. See "Local Host" on page 158 for more details.

**Note:** Media5 recommends not to set the *lanStaticNetworkMask* variable to 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host address. Since a subnet must have a network (all bits 0) and a broadcast (all bits 1) address, this leaves no room for hosts.

4. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

## MAC Address Spoofing

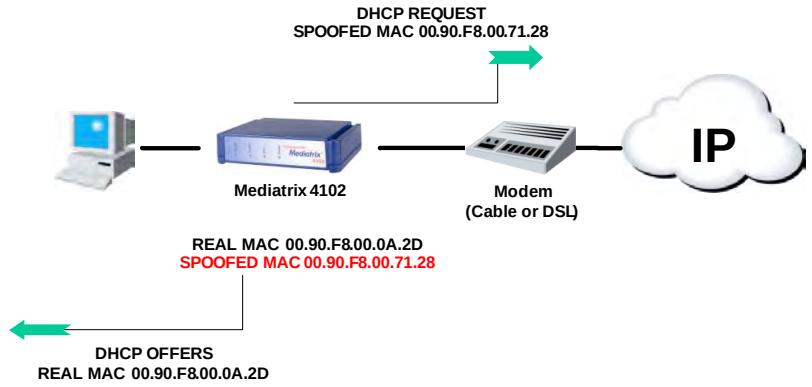
A number of ISPs control connections to their servers by monitoring the MAC address of the connecting device. If the MAC address does not match its database, it refuses the connection.

Consider the typical scenario in which a user of a Mediatrix 4102 is already subscribed to an ISP for a WAN access (i.e., the Internet). If the ISP monitors the MAC address, the user will not be able to connect to the WAN when using the Mediatrix 4102 in front of the usual device.

The workaround is to have the Mediatrix 4102 spoof its MAC address in messages destined to the WAN. The spoofed MAC address matches the ISP's database and the connection will be granted. However, the spoof is only performed on the WAN side of the Mediatrix 4102. The LAN (or private) side of the Mediatrix 4102 remains unchanged.

For example, DHCP requests sent by the Mediatrix 4102 on the WAN side would contain the spoofed MAC address, but DHCP offers returned by the Mediatrix 4102 to the device networked on the *LAN* connector will contain the real MAC address of the Mediatrix 4102.

**Figure 89:** MAC Address Spoofing




You can also use the web interface to enter the MAC address spoofing information. See "LAN Page" on page 37 for more details.

▶ **To spoof the MAC address:**

1. In the *ipRoutingMIB*, set the MAC address used to spoof the unit's actual MAC address in the *ipRoutingMacSpoofAddress* variable.

   A valid MAC address is a continuous series of 12 hexadecimal digits (without colons). An empty character string means that the spoofing is considered disabled, even though the *ipRoutingMacSpoofEnable* variable is set to **enable**.

**Note:** The following MAC addresses are not allowed:

- 000000000000
- FFFFFFFFFFFF
- 01xxxxxxxxxx, where x can be any digit or letter

You can view the current MAC address of the online device in the *LAN* connector in the *ipRoutingMacAddress* variable (*ipRoutingMacSpoof* group of the *ipRoutingMIB*).

**Note:** When dialing the ***#*1** digits on a telephone connected to the Mediatrix 4102, the real MAC address of the unit is returned, not the spoofed one. See "Special Vocal Features" on page 16 for more details.

If you are using a router that is connected to the *LAN* connector of the Mediatrix 4102, you must disable the router's MAC address spoofing feature so that the Mediatrix 4102 properly spoofs the MAC address of the PC connected to the router. In this case, the *ipRoutingMacAddress* variable does not show the PC's MAC address, but rather the router's MAC address.

2. Enable the MAC address spoofing by setting the *ipRoutingMacSpoofEnable* variable to **enable**.

   The unit's MAC address used on the WAN side is the configured MAC address.

   If you set the variable to **disable**, the unit's MAC address used on the WAN side is the actual unit's MAC address.

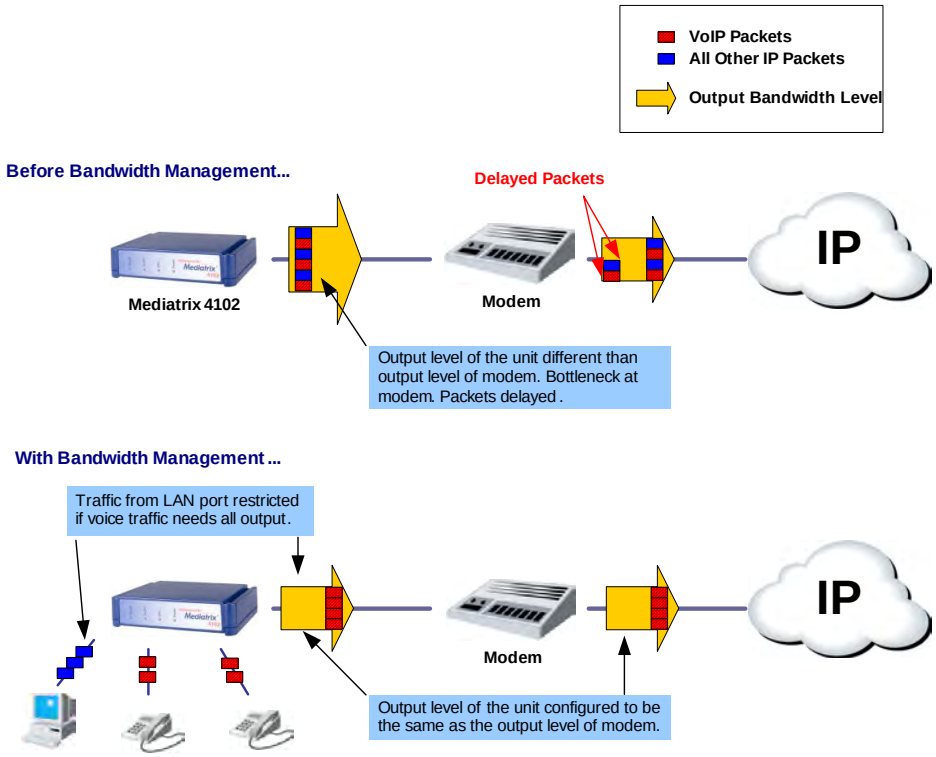
3. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

## WAN Upstream Bandwidth Control

The bandwidth management feature limits the upload bandwidth on the WAN interface. This allows to optimize the voice quality over an Ethernet link; the Mediatrix 4102 knows the available bandwidth on its WAN interface and can slow down the traffic coming from its LAN interface to use all the available bandwidth for voice traffic first.

When the bandwidth control is enabled, packets sent from the PC or IP equipment to the WAN are limited in bandwidth. You can determine the maximum bandwidth. Excess packets coming from the LAN interface are dropped. The Mediatrix 4102 uses the priorities set in "IEEE 802.1q" on page 366 to determine which packets to drop first, even when 802.1q tagging is disabled.

**Figure 90:** Bandwidth Management



▶ **To configure the WAN upstream bandwidth control:**

1. In the *ipRoutingMIB*, set the maximum outgoing amount of data transferred (throughput) to the WAN interface in the *ipRoutingWanUpstreamBandwidth* variable.

   Outgoing traffic includes both traffic generated by voice calls from the Mediatrix 4102 and traffic coming from the PC or IP equipment on the LAN side. The value is expressed in kilobits per seconds. The default value is **512** kbps.

2. Enable the WAN upstream bandwidth control by setting the *ipRoutingBandwidthControlEnable* variable to **enable**.

   The Mediatrix 4102 will limit the outgoing throughput on the WAN interface. The unit must be in router mode to properly use the bandwidth control. See "Router Mode" on page 200 for more details.

**Note:** If TAS is not enabled, the Mediatrix 4102 will still prioritize voice over the signalling and SNMP/HTTP information on a low bandwidth connection.

If you set the variable to **disable**, the bandwidth from the PC or IP equipment to the WAN is not limited.

## Enabling TAS

Enabling TAS is essential to use a broadband connection. By enabling TAS, you also implicitly enable the Mediatrix 4102 in router mode. See "Router Mode" on page 200 for more details.

▶ **To configure TAS:**

1. In the *ipRoutingMIB*, set the duration, in seconds, of the lease offered by the DHCP server in the *ipRoutingDhcpServerLeaseTime* variable (in the *ipRoutingDhcp* group).

   The TAS service of the Mediatrix 4102 contains a DHCP server. Enabling TAS also enables this DHCP server, which allocates an IP address to the PC or IP equipment located on the LAN. See "DHCP Server" on page 212 for more details.

**Note:** The DHCP server embedded in the Mediatrix 4102 allocates only one address. To connect more PCs to the Mediatrix 4102, use a router.

The *ipRoutingDhcpServerLeaseTime* variable is the lease time given to the PC connected to the Mediatrix 4102.

- If the lease time is short, the PC will react faster to address changes, but it will have to renew its lease often.
- If the lease time is long, the PC will react more slowly to address changes.

2. Enable TAS by setting the *ipRoutingEnable* variable to **enable**.

   You may want to disable the factory reset procedure of the Mediatrix 4102, even if users depress the *Default Settings* switch. See "Disabling the Factory Reset" on page 23 for more details.

3. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211

   Optionally, you may want to modify the port allocation settings.

# Ports Settings

The Mediatrix 4102 allows you to define how to dynamically allocate the ports it uses. This allows for better NAT and firewall traversal capabilities.

## UDP and TCP Ports

When needed, the TCP/IP implementation of the Mediatrix 4102 randomly selects a dynamic port amongst the free ports of the range.

Let's say for instance that the dynamic ports range is from 41000 to 42000. The Mediatrix 4102 needs to download a new version of its software. The TCP/IP implementation selects a local UDP port for the TFTP client. The port is selected in the dynamic ports range so the port has a value between 41000 and 42000.

**To set the range of the UDP and TCP ports:**

1. In the *sysConfigMIB*, set the lower boundary for the range of dynamic UDP and TCP ports in the *sysConfigMinDynamicPort* variable.

   The default lower boundary value is **31001**.

2. Set the upper boundary for the range of dynamic UDP and TCP ports in the *sysConfigMaxDynamicPort* variable.

   The default upper boundary value is **32000**.

3. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

**Note:** The smallest acceptable range – between the lower and upper boundaries – is 500.

## T.38 Base Port Range

The T.38 ports are allocated starting from the base port. T.38 uses one port for each FXS interface.

**To define a T.38 base port:**

1. In the *dataIfMIB*, set the *dataIfT38BasePort* variable with the port number you want to use as T.38 base port.

   The default T.38 base port is **6004**. In the case of the base port defined on 6004:

   - If there is currently no ongoing call and FXS connector 1 has an incoming or outgoing call, it uses the T.38 port 6004.
   - If there is currently a call on FXS connector 1 and FXS connector 2 has an incoming or outgoing call, then FXS connector 2 uses the T.38 port 6005.

2. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

### UDP and TCP Ports Ranges

The UDP and TCP ports are separated in three ranges: well-known ports (0 to 1023), registered ports (1024 to 49151) and dynamic ports (49152 to 65535). The IANA (Internet Assigned Numbers Authority, www.iana.org) assigns the well-known ports. The IANA also lists the registered ports. The dynamic ports are not under the authority of the IANA. Most TCP/IP implementations use the range 1025 to 65535 for dynamic ports instead of the range defined by the IANA.

### RTP/RTCP Base Port Range

The RTP/RTCP ports are allocated starting from the base port. The Mediatrix 4102 may use two or four RTP/RTCP ports for each FXS interface:

- It uses two ports in case of a standard call.
- It uses four ports in other types of calls such as a conference call, a call transfer, etc.

**To define a RTP/RTCP base port:**

1. In the *rtpMIB*, set the *rtpConfigBasePort* variable with the port number you want to use as RTP/RTCP base port.

   The default RTP/RTCP base port is **5004**. In the case of the base port defined on 5004:

   - If there is currently no ongoing call and FXS connector 1 has an incoming or outgoing call, it uses the RTP/RTCP ports 5004 and 5005.
   - If there is currently a standard call on FXS connector 1 and FXS connector 2 has a conference call, then FXS connector 2 uses the RTP/RTCP ports 5006, 5007, 5008, and 5009, which are the next available ports.

2. If you do not need to configure other parameters, restart the Mediatrix 4102 as per "Restarting the Mediatrix 4102" on page 211.

# Restarting the Mediatrix 4102

Once all the mandatory and optional changes are done, restart the Mediatrix 4102 so that the changes may take effect.

If you are using , the Mediatrix 4102 tries to establish a PPP connection to the access concentrator.

The IP address of the PC is set to be the exact same as the unit's WAN address. The embedded DHCP server provides the IP address to the PC.

**Figure 91:** Network Layout with Addresses Example

# DHCP Server

Enabling the TAS service also enables a DHCP server that allocates IP addresses to the PC located on the LAN.

## DHCP Server Compliance

| | |
|---|---|
| **Standards Supported** | • RFC 2131 – Dynamic Host Configuration Protocol, section 2<br>• RFC 2132 – DHCP Options and BOOTP Vendor Extensions |

The DHCP server is compliant to RFC 2131 and RFC 2132 with the following limitations:

- The pool of assignable IP addresses contains only one address. This means that only one DHCP client at a time can lease an address. All other lease requests are ignored.
- The DHCP server never accepts the IP address requested by the client (option 50) unless by coincidence.
- The DHCP server never accepts the lease time that is proposed by the client (option 51) unless by coincidence.
- The DHCP server returns only the options specified in "Supported DHCP Options" on page 212, all other parameter requests (as part of option 55) are ignored.

## Supported DHCP Options

The DHCP server embedded in the Mediatrix 4102 supports the following options.

**Table 129:** Supported DHCP Option

| Code | Description |
|---|---|
| 1 | Network Mask<br>When the router state is public, returns a subnet mask that depends on the class of the WAN address. See RFC 791, section 3.2.<br>When the router state is private, returns the network mask as configured in the MIB. |
| 3 | Router Option<br>When the router state is public, returns the first address in the subnet.<br>When the router state is private, does not return a router IP address. |
| 6 | Domain Name Server Option<br>Returns the list of DNS addresses as configured in the MIB. |
| 42 | Network Time Server Option<br>Returns the SNTP server address as configured in the MIB. If the MIB contains a FQDN, option 42 contains the resolved IP address. |
| 51 | Lease Time<br>Returns 20 seconds. |
| 52 | Option Overload<br>Always set to "3". |
| 53 | DHCP Message Type |
| 53 | Server Identifier<br>Returns the unit's LAN IP address. |

# DSL Modem Specific Information

The following sections apply only if you are using a DSL modem.

## Establishing a Connection

When the Mediatrix 4102 restarts, it establishes the connection to the access concentrator in conformance with the RFCs listed in "PPPoE Service" on page 202.

When establishing a PPP connection, the Mediatrix 4102 goes through three distinct phases:

- Discovery phase
- Authentication phase
- Network-layer protocol phase

### Discovery Phase

The Mediatrix 4102 broadcasts the value of the *pppoeServiceName* MIB variable (see "Enabling the PPPoE Service" on page 202 for more details).

The access concentrator with a matching service name answers the Mediatrix 4102.

- If no access concentrator answers, this creates a "PPPoE failure" error. The Mediatrix 4102 handles it as described in Table 130 on page 214.
- If more than one access concentrators respond to the discovery, the Mediatrix 4102 tries to establish the PPP connection with the first one that supports the requested service name.

### Authentication Phase

If the access concentrator requests authentication, the Mediatrix 4102 sends the ID/secret pair configured in the *pppSecuritySecretsTable* (see "Setting a User Name and Password" on page 203 for details). If the access concentrator rejects the authentication, this creates an "authentication failure" error. The Mediatrix 4102 handles it as described in Table 130 on page 214.

### Network-Layer Protocol Phase

The Mediatrix 4102 negotiates an IP address. The requested IP address is the one from the last successful PPPoE connection. If the Mediatrix 4102 never connected by using PPPoE (or after a factory reset), it does not request any specific IP address.

When the PPP connection is established, the access concentrator assigns an IP address to the Mediatrix 4102. This IP address may be used as the WAN IP address. See "WAN Information Configuration Source" on page 204 for details.

Primary and secondary DNS servers may be supplied by the access concentrator. If this is the case, the new DNS servers supersede the servers defined locally.

### Configuration of DNS Servers

When the PPP connection is active, the DNS servers supplied by the access concentrator supersede the locally defined servers as follows:

- If the *localHostDnsOverrideEnable* variable is set to **enable**, the servers supplied by the access concentrator do not replace the *localHostPrimaryDns* or *localHostSecondaryDns* variables. See "Static DNS" on page 160 for more details.
- If the *localHostDnsOverrideEnable* variable is set to **disable**:
  - If no server is supplied by the access concentrator, the value of the *localHostPrimaryDns* and *localHostSecondaryDns* variables applies.
  - If one server is supplied by the access concentrator, it replaces the server defined in the *localHostPrimaryDns* variable.

- If two servers are supplied by the access concentrator, they replace both the *localHostPrimaryDns* and *localHostSecondaryDns* variables.

**Note:** If the DNS servers addresses sent by the access concentrator change while the Mediatrix 4102 is establishing the connection to the access concentrator, you must restart the unit so that it uses the new addresses.

## Error Handling

The following describes the Mediatrix 4102 behaviour in case of error.

**Table 130:** Error Handling

| On this error... | The Mediatrix 4102... |
|---|---|
| Authentication failure | **1.** Waits for 10 seconds.<br>**2.** Retries to establish the connection as in "Establishing a Connection" on page 213. |
| PPPoE failure | **1.** Waits for 10 seconds.<br>**2.** Retries to establish the connection as in "Establishing a Connection" on page 213. |
| Peer not responding | **1.** Retries to establish the connection as in "Establishing a Connection" on page 213. |

### Connection Unsuccessful

If the first PPP connection fails, the Mediatrix 4102 stops its initialization until the PPP connection is opened successfully. Until the PPP connection is successfully established, the unit cannot be reached by using SNMP. However, you can access the web page through the LAN port.

### PPP Connection Loss

A network connection may abruptly shutdown for many reasons. One of the most common reasons is the maintenance of the network and its environment.

When the Mediatrix 4102 detects it has lost the PPP connection, it tries to re-establish the connection as in "Establishing a Connection" on page 213.

It is possible that the IP address assigned by the access concentrator changes after re-establishing a connection. In this case, the Mediatrix 4102 restarts because it does not support dynamic IP address change.

# Routing Mechanism

Usually, the packets go through the Mediatrix 4102.

If the PC wants to directly contact the Mediatrix 4102, it must use the unit's LAN address (*localHostAddress* variable if TAS is disabled or *lanStaticAddress* if TAS is enabled).

## Blocked Ports

The Mediatrix 4102 uses some ports for signalling, media transport and management purposes. Packets sent to these ports are blocked. Most of the ports can be configured by using a MIB variable.

**Table 131:** Blocked Ports

| Port Description | Port Number | Configurable |
|---|---|---|
| SNMP | 161 UDP | Yes |
| DHCP offer listening | 68 UDP | No |
| TFTP server for configuration downloads.<br>**Note**: This port is used only if the configuration download service is enabled, and only for the time it is required. | 69 UDP | No |
| SIP | 5060 UDP/ TCP | Yes |
| RTP/RTCP<br>**Note**: The Mediatrix 4102 uses up to four UDP ports per FXS interface. See "RTP/RTCP Base Port Range" on page 211 for details. | 5004+ UDP | Yes |
| T.38<br>**Note**: The Mediatrix 4102 uses one UDP port per FXS interface. See "T.38 Base Port Range" on page 210 for details. | 6001+ UDP | Yes |

# Using the Mediatrix 4102 with a Low Bandwidth Connection

You can use the Mediatrix 4102 with a low bandwidth connection without any visible performance issues. This is true for both DHCP and PPPoE connections, with either a DSL or cable modem. However, you must configure the Mediatrix 4102 accordingly.

## What is Considered a Low Bandwidth Connection?

Media5 considers that the process is not optimal when the transmission of one or more large Ethernet packet takes more time than the time it takes for packetization. For instance, a 1518 bytes packet takes the following transmission time according to the bandwidth:

**Table 132:** Transmission Time vs. Bandwidth

| Bandwidth (kbps) | Transmission Time (ms) |
|---|---|
| 64 | 190 |
| 128 | 95 |
| 256 | 47 |
| 512 | 24 |
| 1024 | 12 |
| 2048 | 6 |
| 4096 | 3 |

One can see that from 128 kbps or less, the delay becomes significant, considering this is for one packet only. Media5 thus recommends to follow the configuration in the next section for a bandwidth lower than 4096 kbps. This will ensure the best voice quality possible.

## Configuration for a Low Bandwidth Connection

The following steps should allow you to use a low bandwidth connection.

▶ **To use the Mediatrix 4102 with a low bandwidth connection:**

1. Configure the “Bandwidth Management” feature as described in “WAN Upstream Bandwidth Control” on page 208.

   *a.* Set the maximum bandwidth in kilobits per seconds (kbps) in the *ipRoutingWanUpstreamBandwidth* variable.

   This value should be implemented or provided by your ISP. It must be between 64 kbps and 4096 kpbs.

   *b.* Enable the “Bandwidth Management” feature by setting the *ipRoutingBandwidthControlEnable* variable to **enable**.

2. Configure the Mediatrix 4102 in TAS mode as described in this chapter. Enable the TAS mode by setting the *ipRoutingEnable* variable to **enable**. See “Enabling TAS” on page 209 for more details.

3. In the *qosIeee8021q* group of the *qosMIB*, define a 802.1q priority for the voice and fax packets in the following variables.
   - *qosVoiceIeee8021qUserPriority* for voice priority
   - *qosT38FaxIeee8021qUserPriority* for fax priority

   The Mediatrix 4102 uses 9 output queues with increasing priorities. The lowest priority queue is always used for the traffic coming from the LAN port and routed to the WAN port. You cannot change this priority level. The 8 other queues are used for the traffic the Mediatrix 4102 sends. By default, the lowest priority queue (of these 8 queues) is used for all traffic, and the other 7 are unused.

However, you can assign certain types of traffic to other queues of higher priority by configuring a 802.1q priority. It is not required to activate the packet tagging feature, only provide a priority to the protocol.

A good practice would be to always have a priority other than “0” for the voice and the fax. The signalling could also receive a higher priority. Thus, the SNMP and HTTP accesses performed from the WAN would be answered in lower priority.

See “IEEE 802.1q” on page 366 for more details on 802.1q priorities.

4. Set the value of the following variables to **enable**:
   - qosVoiceIeee8021qEnable
   - qosT38FaxIeee8021qEnable

   The corresponding user priority configuration is enabled.

5. Restart the Mediatrix 4102 so that the changes may take effect.

CHAPTER 14

# Configuration File Download

The configuration file download feature allows to update the Mediatrix 4102 configuration by transferring a configuration file via TFTP, HTTPS, or HTTP. The configuration file can either be transferred from the management server or from the configuration file download server. The main difference is the session initiator, which is respectively the management server and the Mediatrix 4102. The advantage of having the Mediatrix 4102 as the session initiator is to allow NAT traversal.

You can also manually upload a configuration file to the Mediatrix 4102 by using the End User web interface. See "Configuration File Upload Page" on page 39 for more details.

You can also set these parameters via the web interface, as described in "Configuration File Download" on page 54.

## Configuration File Download Server

The service allows to download a unique file for each Mediatrix 4102, and/or a file shared among many units. These configuration files may be encrypted or not.

You have the choice to perform the configuration file download by using the TFTP protocol, the HTTPS protocol or the HTTP protocol. You can also configure the Mediatrix 4102 to automatically update its configuration.

To download a configuration file, you may need to setup the following applications on your computer:

- TFTP server with proper root path
- SNTP server properly configured
- HTTP server with proper root path
- HTTPS server with proper root path
- Configuration source
- Configuration file name and location

### Configuring the TFTP Server

If you are to perform a configuration file download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the TFTP server host. It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

### Configuring the SNTP Server

If you are to use the automatic configuration file update feature (see "Automatic Configuration Update" on page 228 for more details) or the HTTPS protocol, you need to have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

### Configuring the HTTP Server

If you are to perform a configuration file download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

## Configuring the HTTPS Server

| | |
|---|---|
| **Standards Supported** | • RFC 2246 – The TLS Protocol Version 1.0<br>• RFC 2818 – HTTP Over TLS<br>• RFC 2459 – X.509 Digital Certificates<br>• RFC 3280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile |

If you are to perform a configuration file download that requires authentication or privacy by using the HTTP over the Transport Layer Security (TLS) protocol (HTTPS), you must install a HTTPS server running on the PC designated as the server host. It is assumed that you know how to set the root path and set the SSL/TLS security configuration. If not, refer to your HTTPS server's documentation.

**Note:** The web interface does not support the HTTPS protocol.

**Caution:** You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

The Mediatrix 4102 supports the following SSL/TLS parameters:

**Table 133:** Secure Parameters Supported

| Supported Parameter | Description |
|---|---|
| Key Exchange Mechanism | • RSA<br>• Diffie-Hellman |
| Ciphers | • AES (128 and 256 bits)<br>• 3DES (168 bits) |
| Message Digests | • SHA-1 |

**Note:** Media5 recommends to use cipher suites based on the RSA key exchange mechanism, because the Diffie-Hellman key exchange mechanism introduces a noticeable delay in the HTTPS session establishment.

### Certificates

The Mediatrix 4102 contains embedded security certificates formatted as per ITU x.509 and RFC 3280. The certificates are factory-installed. The name of X.509 certificates currently installed in the Mediatrix 4102 are listed in the *securityCertificatesMIB* under the *certificateName* variable (under the *certificateTable* group). You must have at least one matching certificate on your HTTPS server.

You can also find the following information:

**Table 134:** Certificates

| Parameter | Description |
|---|---|
| certificateSubjectCommonName | The certificate's subject name.<br>If the certificate contains a subject field, display the common name. Otherwise display the first non-empty subject alternative name from the following list:<br>• Uniform Resource Locator<br>• DNS name<br>• IP Address<br>• RFC 822 name<br>See RFC 3280 sections 4.1.2.6 and 4.2.1.7 for details. |
| certificateExpirationDate | The certificate's expiration date.<br>Display the date at which the certificate expires. The format is MM/DD/YYYY in universal time.<br>See RFC 3280 section 4.1.2.5 for details. |

When contacting a HTTPS server, the Mediatrix 4102 establishes a TLS connection by (among others):

- negotiating cipher suites
- checking the server certificates validity (dates)

The Mediatrix 4102 then checks the server's identity by validating the host name used to contact it against the information found in the server's certificate, as described in RFC 2818, section 3.1.

If any of the above does not succeed, the Mediatrix 4102 refuses the secure connection. To help detect such errors, you can increase the syslog messages level.

## Configuration File Server Settings

The Mediatrix 4102 must know the IP address and port number of its configuration file server. This server contains the configuration file the Mediatrix 4102 will download. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself in static variables.

### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

**To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable (under the *ipAddressConfigFileFetching* group).

   This variable defines whether the Mediatrix 4102 must ask for its configuration file server settings through a DHCP server or not.

2. Set the *configFileFetchingConfigSource* variable to **dhcp**.

   You can query the configuration file server's IP address and port number assigned by the DHCP server in the following read-only variables (in the *ipAddressStatus* folder):

   - configFileFetchingHost
   - configFileFetchingPort

3. Set how you want to define the configuration server information in the DHCP server:

**Table 135:** Configuration File Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *configFileFetchingDhcpSiteSpecificCode* variable to **0**. Set the configuration file server IP address in the DHCP server inside the vendor specific sub-option 201 (hexadecimal 0xC9). |
| site specific code | The *configFileFetchingDhcpSiteSpecificCode* variable to any value between 128 and 254. Set the configuration file server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *configFileFetchingDhcpSiteSpecificCode* variable in the unit's configuration). |

See "Vendor and Site Specific DHCP Options" on page 169 for more details.

### Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable.

   This variable defines whether the Mediatrix 4102 must ask for its configuration file server settings through a DHCP server or not.

2. Set the *configFileFetchingSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 136:** Configuration File Server Static Information

| Variable | Description |
|---|---|
| configFileFetchingStaticHost | Static configuration file server IP address or domain name to use when downloading a configuration file. This is the current address of the PC that hosts the configuration files.<br>**Default Value**: 192.168.0.10 |
| configFileFetchingStaticPort | Static configuration file server IP port number to use when downloading a configuration file.<br>**Default Value**: 69 |

The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP/HTTPS server to perform the configuration file download, you must change the port value to 80.

## Setting up the Configuration File Download

When performing a configuration file download, you can download two different files:

- A generic configuration file that should be used to update a large number of units with the same configuration.
- A specific configuration file that contains the configuration for a single unit, for instance the telephone numbers of its lines.

When both the generic and specific configuration files are downloaded, settings from the specific configuration file always override the settings from the generic configuration file. These files must be located in the same directory.

### ▶ To setup the configuration file download:

1. In the *configFileFetchingMIB*, set the *configFileFetchingFileLocation* variable with the path, on the remote server, of the directory where the configuration files are located.

   The path is case sensitive hence it must be entered properly.

   The path is relative to the root path of the transfer server (*configFileFetchingHost*). Use the "/" character when defining the path to indicate sub-directories.

   Let's consider the following example:

   - The directory that contains the configuration file is called: **Config_File**.
   - This directory is under **C:/Root/Download**.

   **Table 137:** Path Configurations Example

   | Root Path | Corresponding Path Name |
   |---|---|
   | c:/root/download | Config_File |
   | c:/ | root/download/Config_File |
   | c:/root | download/Config_File |

   The following are some tips to help your download process:

   - Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
   - If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
   - Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
   - Cut and paste the path and/or name of the directory that contains the extracted files into the configuration file path of the Mediatrix 4102 (you may have to convert "\" into "/ ") to eliminate typographical errors.

   Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

2. Set the *configFileFetchingFileName* variable with the name of the generic configuration file to download.

**Caution:** The generic configuration file must be in XML format, no matter what its file extension.

The file name is case sensitive hence it must be entered properly.

This file should be used to update a large number of units with the same configuration.

If you leave the variable empty, the Mediatrix 4102 does not download the generic configuration file.

3. Set the *configFileFetchingSpecificFileName* variable with the name of the specific configuration file to download.

**Caution:** The specific configuration file must be in XML format, no matter what its file extension.

The file name is case sensitive hence it must be entered properly.

This file should be used to update the configuration of a single unit.

This variable may contain macros that are substituted by actual values when downloading the configuration file. Supported macros are:

- %mac%: the MAC address of the unit
- %product%: the product name of the unit

- %%: the character “%”

For instance:

- The “%mac%.xml” value for a Mediatrix 4102 with MAC address “0090F12345AB” will be “0090F12345AB.xml”.
- The value “Hello%%Hi” will result in “Hello%Hi”.
- The value “%%%mac%%%mac%.xml” will result in “%0090F12345AB%mac%.xml”.

  From left to right: the first macro encountered is first substituted, the second macro encountered is then substituted, etc.

When the character “%” is not part of a macro, it is not replaced. The following are examples:

- The value “%mac.xml” stays “%mac.xml”
- The value “Hello%Hi” stays “Hello%Hi”
- The value “%moc%.xml” stays “%moc%.xml”

If the variable is empty (after macro substitution), the Mediatrix 4102 does not download the specific configuration file.

## Configuration Update Status

If valid configuration files are successfully downloaded, then the Mediatrix 4102 automatically restarts to apply all the new settings. If the Mediatrix 4102 does not restart, this could mean the download failed or that the configuration in the file is the same as the configuration in the unit.

You can validate the status of the configuration update in various ways.

### MIB Variable

You can query the status of the last configuration file download in the *sysAdminDownloadConfigFileStatus* variable:

- idle: No configuration file download has been performed yet.
- fail: The last configuration file download failed.
- success: The last configuration file download succeeded.
- inProgress: A configuration file download is in progress.
- listening: The unit is listening and waiting for a configuration file to be sent by the management server.

### Syslog Messages

A lot of information is transmitted as system log (syslog) messages. The following are some of the syslog messages sent by the unit:

**Table 138:** Configuration File Download Syslog Messages

| Level | Message | Event |
|---|---|---|
| Informational | `The specific configuration update succeeded.` | The configuration update with the specific configuration file has been successful. |
| Error | `The specific configuration update failed.` | The configuration update with the specific configuration file experienced an error and has not been completed. |
| Informational | `The configuration file “XXX” was successfully fetched.` | A configuration file was successfully fetched. |
| Informational | `The unit configuration is not updated. The parameter values defined in the fetched configuration files are identical to the actual unit configuration.` | The parameter values defined in the fetched configuration files are identical to the actual unit configuration. |

**Table 138:** Configuration File Download Syslog Messages (Continued)

| Level | Message | Event |
|---|---|---|
| Informational | `The generic file \"%s\" parameter values are not applied. They are either identical to the unit configuration or overwritten by the specific file.` | The generic configuration file parameter values are either identical to the unit configuration or overwritten by the specific configuration file. |
| Warning | `None of the parameter values defined in the configuration file \"%s\" was successfully applied.` | No parameter value from a fetched configuration file was successfully applied (e.g., because of bad OIDs). |
| Informational | `Parameter values defined in the configuration file \"%s\" were successfully applied.` | A fetched configuration file was successfully applied. |
| Informational | `The unit is restarting to complete the configuration update.` | All necessary fetched configuration files were successfully applied. |

You can view these messages in the End User web interface. See "System Log Page" on page 42 for more details.

## Configuration Files Encryption

You can secure the exchange of configuration files between the server and the Mediatrix 4102. A privacy key allows the unit to decrypt a previously encrypted configuration file. This applies to files downloaded via TFTP, HTTPS, or HTTP, but NOT on updates performed from the web interface.

To encrypt a configuration file (generic or specific), you must use the MxCryptFile application. MxCryptFile is a command line tool that encrypts files before sending them to the Mediatrix 4102 unit. Contact your sales representative for more details.

### Configuration File Decryption on the Mediatrix 4102

The following describes how to decrypt a previously encrypted generic or specific configuration file. You must have one key for the generic configuration file and another key for the specific configuration file.

▶ **To decrypt a configuration file:**

1. In the *configFileFetchingMIB,* set the proper decryption variable with the secret key used to decrypt the configuration file.

**Table 139:** Decryption Variables

| Configuration File | Variable |
|---|---|
| Generic | configFilePrivacyGenericSecret |
| Specific | configFilePrivacySpecificSecret |

The key is encoded in hexadecimal notation. You can thus use characters in the range 0-9, A-F. All other characters are not supported.

Each character encodes 4 bits of the key. For instance, a 32-bit key requires 8 characters.

- If you enter too many bits, the key is truncated to the first 448 bits.
- If you do not enter enough bits, the key is padded with zeros.

For instance, a 32-bit key could look like the following: A36CB299.

This key must match the key used for the encryption of the relevant configuration file.

If the variable is empty, the configuration file is not decrypted.

2. Set the *configFilePrivacyEnable* variable to **enable**.

   The Mediatrix 4102 will be able to decrypt the next encrypted generic or specific configuration file. If this variable is set to **disable**, the configuration file is not decrypted by the unit and the configuration update fails.

## Configuration Download via TFTP

The following steps explain how to download configuration files by using the TFTP protocol.

**Note:** The configuration download via TFTP can only traverse NATs of types “Full Cone” or “Restricted Cone”. If the NAT you are using is of type “Port Restricted Cone” or “Symmetric”, the file transfer will not work.

**To download configuration files via TFTP:**

1. Set the configuration file server host and port as defined in “Configuration File Server Settings” on page 221.
2. Place the configuration files to download on the computer hosting the TFTP server.

   These files must be in a directory under the TFTP root path.
3. If not already done, set the configuration file path as described in “Setting up the Configuration File Download” on page 222.
4. In the *configFileFetchingMIB,* set the *configFileTransferProtocol* variable to **tftp**.
5. In the *groupAdminMIB*, set the *groupSetAdmin* variable to **ForceLock**.

   All activities in progress on the Mediatrix 4102 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is “locked”). The configuration file download may take place.
6. In the *sysAdminMIB*, initiate the configuration file download via TFTP by setting the *sysConfigCommand* variable to **updateConfiguration**.

   The Mediatrix 4102 immediately downloads the configuration files. It is the initiator of the TFTP sessions.

### NAT Variations

NAT treatment of UDP varies among implementations. The four treatments are:

- Full Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host by sending a packet to the mapped external address.
- Restricted Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.
- Port Restricted Cone: Similar to a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.
- Symmetric: All requests from the same internal IP address and port, to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

For more details on NAT treatments, refer to RFC 3489.

## Configuration Download via HTTP/HTTPS

The following steps explain how to download the configuration files by using the HTTP or HTTPS protocol. If you are using HTTPS, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 220 for more details.

▶ **To download the configuration files via HTTP or HTTPS:**

1. Set the configuration file server host and port as defined in "Configuration File Server Settings" on page 221.

> **Caution:** When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *configFileFetchingPort* read-only variable (in the *ipAddressStatus* folder).
>
> If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 221 for more details.

2. Place the configuration files to download on the computer hosting the HTTP or HTTPS server.

   These files must be in a directory under the root path.

3. If not already done, set the configuration file path as described in "Setting up the Configuration File Download" on page 222.

4. In the *configFileFetchingMIB,* set the *configFileTransferProtocol* variable to **http** or **https**.

   Your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.

5. If your HTTP or HTTPS server requires authentication when downloading the configuration file, set the following:
   - The user name in the *configFileTransferUsername* variable.
   - The password in the *configFileTransferPassword* variable.

6. In the *groupAdminMIB*, set the *groupSetAdmin* variable to **ForceLock**.

   All activities in progress on the Mediatrix 4102 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is "locked"). The configuration file download may take place.

7. In the *sysAdminMIB*, initiate the configuration file download via HTTP or HTTPS by setting the *sysConfigCommand* variable to **updateConfiguration**.

   The Mediatrix 4102 immediately downloads the configuration files. It is the initiator of the HTTP/HTTPS sessions.

### User Agent Header of HTTP Requests

The *User-Agent* header field of an HTTP request contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

```
User-Agent: Softphone Beta1.5
```

You can customize the information that the Mediatrix 4102 sends when establishing a communication.

▶ **To customize the HTTP User Agent header of HTTP requests:**

1. In the *interopMIB*, set the *mxInteropHttpUAHeaderConfig* variable with the proper macro.

   The following macros are replaced by their representation:
   - **%version%**: Version of the application.
   - **%mac%**: Unit MAC address (lowercase).
   - **%rev%**: Hardware revision number.
   - **%product%**: Product name.

- **%%**: A '%' sign.

2. Restart the Mediatrix 4102 so that the changes may take effect.

## Automatic Configuration Update

You can configure the Mediatrix 4102 to automatically update its configuration. This update can be done:

- Every time the Mediatrix 4102 restarts.
- At a specific time interval you can define.

### Automatic Update on Restart

The Mediatrix 4102 may download new configuration files each time it restarts.

**To set the automatic update every time the Mediatrix 4102 restarts:**

1. Set the configuration file server host and port as defined in "Configuration File Server Settings" on page 221.

**Caution:** When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 221 for more details.

2. Place the configuration files to download on the computer hosting the HTTP, HTTPS, or TFTP server.

   These files must be in a directory under the root path.

3. If not already done, set the configuration file path as described in "Setting up the Configuration File Download" on page 222.

4. In the *configFileFetchingMIB,* set the *configFileTransferProtocol* variable to either **http**, **https**, or **tftp**.

   If you are using the HTTPS protocol, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 220 for more details.

   If you are using the HTTP or HTTPS protocol to download the configuration, be aware that your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.

5. If you are using the HTTP or HTTPS protocol to download the configuration and your HTTP or HTTPS server requires authentication, set the following:
   - The user name in the *configFileTransferUsername* variable.
   - The password in the *configFileTransferPassword* variable.

   The Mediatrix 4102 supports basic and digest HTTP authentication, as described in RFC 2617.

6. Set the *configFileAutoUpdateOnRestartEnable* variable to **enable** (in the *configFileAutomaticUpdate* group).

7. In the *sysConfigMIB*, set the *sysConfigDownloadConfigFile* variable to **automaticInitiateFileDownload**.

   The automatic configuration update will be performed each time the Mediatrix 4102 restarts.

   The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

## Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4102 to download new configuration files at a specific day and/or time.

▶ **To set the automatic update at a specific time interval:**

1. Set the configuration file server host and port as defined in "Configuration File Server Settings" on page 221.

> **Caution:** When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).
>
> If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Configuration File Server Settings" on page 221 for more details.

2. Place the configuration files to download on the computer hosting the HTTP, HTTPS, or TFTP server.

   These files must be in a directory under the root path.

3. If not already done, set the configuration file path as described in "Setting up the Configuration File Download" on page 222.

4. In the *configFileFetchingMIB,* set the *configFileTransferProtocol* variable to either **http**, **https**, or **tftp**.

   If you are using HTTPS, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 220 for more details.

   If you are using the HTTP or HTTPS protocol to download the configuration, be aware that your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.

5. If you are using the HTTP or HTTPS protocol to download the configuration and your HTTP or HTTPS server requires authentication, set the following:
   - The user name in the *configFileTransferUsername* variable.
   - The password in the *configFileTransferPassword* variable.

   The Mediatrix 4102 supports basic and digest HTTP authentication, as described in RFC 2617.

6. Define the time base for automatic configuration updates in the *configFileAutoUpdateTimeUnit* variable (in the *configFileAutomaticUpdate* group).

   You have the following choices:

**Table 140:** Time Unit Parameters

| Parameter | Description |
|---|---|
| minutes | Updates the unit's configuration every *x* minutes. You can specify the *x* value in the variable *configFileAutoUpdatePeriod* (see Step 7). |
| hours | Updates the unit's configuration every *x* hours. You can specify the *x* value in the variable *configFileAutoUpdatePeriod* (see Step 7). |
| days | Updates the unit's configuration every *x* days. You can specify the *x* value in the variable *configFileAutoUpdatePeriod* (see Step 7).<br>You can also define the time of day when to perform the update in the *configFileAutoUpdateTimeRange* variable (see Step 8). |

7. Set the waiting period between each configuration update in the *configFileAutoUpdatePeriod* variable.

   The time unit for the period is specified by the *configFileAutoUpdateTimeUnit* variable (see Step 6). Available values are from 1 to 48.

8. If you have selected **days** in Step 6, set the time of the day when to initiate a configuration update in the *configFileAutoUpdateTimeRange* variable.

   The time of the day is based on the *sntpTimeZoneString* variable setting (see "Chapter 21 - SNTP Settings" on page 317 for more details).

   You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

   If a time range is specified, the unit will download the configuration files at a random time within the interval specified.

   The format should be one of the following:

   ```
   hh[:mm[:ss]]
   hh[:mm[:ss]] - hh[:mm[:ss]]
   ```

   Where:

   ```
   hh: Hours.
   mm: Minutes.
   ss: Seconds.
   ```

   The configuration files are downloaded at the first occurrence of this value and thereafter with a period defined by the *configFileAutoUpdatePeriod* variable. Let's say for instance the automatic unit configuration update is set with the time of day at 14h00 and the update period at every 2 days.

   - If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.
   - If the time range is set to '14:00 - 15:00' and the automatic unit configuration update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

9. Set the *configFileAutoUpdatePeriodicEnable* variable to **enable**.

10. In the *sysConfigMIB*, set the *sysConfigDownloadConfigFile* variable to **automaticInitiateFileDownload**.

    The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

    If one of the telephones/faxes is off-hook, the Mediatrix 4102 will perform the update 5 minutes after both ports are detected on-hook.

## Error Handling

The following configuration file fetching service error sources are divided in three types depending on the transfer protocol: common errors (Table 35), TFTP errors (Table 36) and HTTP/HTTPS errors (Table 37). The error cause and the unit behaviour are also described.

**Table 141:** Configuration File Fetching Error Handling

| Error Type | Cause | Behaviour |
|---|---|---|
| **Common Error Handling** | | |
| Invalid file format | The file format is not valid. | Send a syslog **warning** message including the file location/name with the transfer server address:<br>`The fetched configuration file "XXX", from server "XXX", has an invalid format.`<br>No recorded settings applied. |

**Table 141:** Configuration File Fetching Error Handling (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Empty file | Committing an empty file. | Send a syslog **warning** message including the file location/name with the transfer server address:<br>`The fetched configuration file “XXX”, from server “XXX”, is empty.` |
| Invalid file content | The file contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126. | Send a syslog **warning** message including the file location/name, the transfer server address and the invalid character (ASCII code):<br>`The fetched configuration file “XXX”, from server “XXX”, has an invalid character “ASCII code XXX”.`<br>No recorded settings applied. |
| Invalid transfer server address | The server address is not valid. | Send a syslog **warning** message including the transfer server address:<br>`No configuration file is fetched because the server host “XXX” is invalid.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File size too big | Downloading a file with a size exceeding 512000 bytes. | Send a syslog **warning** message including the file location/name, the transfer server address, the file size and the maximum allowed size:<br>`The fetched configuration file “XXX”, from server “XXX”, has a size “XXX bytes” that exceeds the maximum allowed size “XXX bytes”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Invalid encryption | The configuration file cannot be decrypted. A badly encrypted file is detected if the header or the padding is invalid. | Send a syslog **warning** message including the file location/name and the transfer server address:<br>`The fetched configuration file \”%s\”, from server \”%s\”, can not be decrypted.` |
| **TFTP-Specific Error Handling** | | |
| File not found | Received error code 1 (file not found) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file “XXX” was not found on the TFTP server “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Access violation | Received error code 2 (access violation) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file “XXX” was not fetched. There was a TFTP access violation with server “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

**Table 141:** Configuration File Fetching Error Handling (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Connection timeout | No answer from the TFTP server. The time elapsed since the TFTP request was sent exceeds 32 seconds. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`The configuration file “XXX” was not fetched. The TFTP connection with server “XXX” timed out.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Transfer error | Received a TFTP error (other than error code 1 and 2) from the TFTP server. | Send a syslog **warning** message including the file name and location with the TFTP server address:<br>`Error in the TFTP transfer of the configuration file “XXX” from host “XXX” and port number XXX.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File size too big | Downloading a file with a size exceeding 512000 bytes. | Abort the transfer by sending error code 3 (disk full or allocation exceeded) to the TFTP client. |
| **HTTP/HTTPS-Specific Error Handling** | | |
| Access unauthorized | Received a 401 Unauthorized from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The access to configuration file “XXX” is unauthorized on HTTP server “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| File not found | Received a 404 Not Found from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file “XXX” was not found on the HTTP server “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Session timeout | No answer from the HTTP server. The time elapsed since the HTTP request was sent exceeds 15 seconds. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file “XXX” was not fetched. The HTTP session with server “XXX” timed out.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |
| Session closed by peer | The HTTP server closed the session. | Send a syslog **warning** message including the file location/name with the HTTP server address:<br>`The configuration file “XXX” HTTP transfer session was closed by peer: host “XXX”.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

**Table 141:** Configuration File Fetching Error Handling (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Transfer error | Received an HTTP error (other than 401 and 404) from the HTTP server. | Send a syslog **warning** message including the file location/name with the HTTP server address and port:<br>`Error in the HTTP transfer of the configuration file "XXX" from host "XXX" and port number XXX.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail*. |

# Management Server

You can set the Mediatrix 4102 so that it asks the management server to send it a configuration file.

**Note:** Downloading a configuration file from the management server can only be performed through the TFTP protocol.

## Management Server Configuration

To download a configuration file from the management server, you must setup the management server information as per "Chapter 27 - Management Server Configuration" on page 363.

## Downloading from the Management Server

Once the management server has been properly set up, you can define the configuration file download.

In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Downloading a Configuration File*.

**To download the configuration file from the management server:**

1. Place the configuration file on the computer hosting the management server.
2. In the *sysConfigMIB*, request a configuration file download by setting the *sysConfigDownloadConfigFile* variable to **requestFileDownload**.
3. Set the *sysConfigDownloadConfigMode* variable to **request**.

   The Mediatrix 4102 sends a notification, *msTrapConfigInformation,* to the management server, via SNMP traps, to request the configuration file.

   The management server then initiates the TFTP session and pushes the file into the unit.

   If the management server is the Unit Manager Network from Media5, the following steps are automatically performed. If you are using another management server, you may have to perform them manually.

   *a.* The Unit Manager Network sets the *sysConfigDownloadConfigMode* variable to **record**.

   *b.* The Unit Manager Network sends the configuration file to the Mediatrix 4102.

   *c.* Once the configuration file has been sent, the Unit Manager Network sets the *sysConfigDownloadConfigFile* variable to **noFileDownload**.

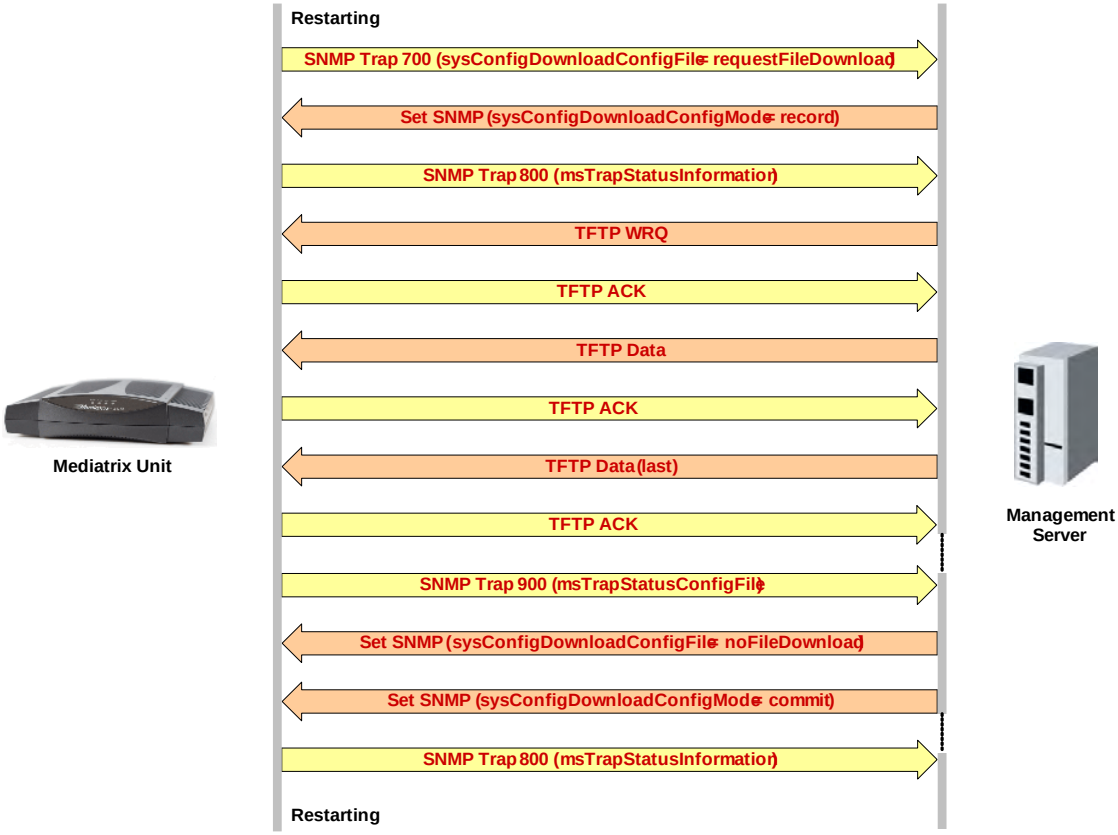
   *d.* The Unit Manager Network sets the *sysConfigDownloadConfigMode* variable to **commit**.

   If a valid configuration file is successfully downloaded, then the Mediatrix 4102 automatically restarts to apply all the new settings. If the Mediatrix 4102 does not restart, this could mean the download failed. In this case, you can query the status of the last configuration file download in the *sysAdminDownloadConfigFileStatus* variable:

   - idle: No configuration file download has been performed yet.

- fail: The last configuration file download failed.
- success: The last configuration file download succeeded.
- inProgress: A configuration file download is in progress.
- listening: The unit is listening and waiting for a configuration file to be sent by the management server.

**Figure 92:** Configuration Sequence Update Using the Management Server




## Error Handling

The following are possible error sources when updating the unit configuration using the management server. The error cause and the unit behaviour are also described.

**Table 142:** Configuration File Error Handling with the Management Server

| Error Type | Cause | Behaviour |
|---|---|---|
| Empty file | Committing an empty file. | Send a syslog **warning** message including the file name and the TFTP client address:<br>`The configuration file “XXX” pushed to the unit by the TFTP client “XXX” is empty.` |
| Invalid file content | Committing a file that contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126. | Send a syslog **warning** message including the file name, the TFTP client address and the invalid character (ASCII code):<br>`The configuration file “XXX” pushed to the unit by the TFTP client “XXX” has an invalid character “ASCII code XXX”.`<br>No recorded settings applied. |

**Table 142:** Configuration File Error Handling with the Management Server (Continued)

| Error Type | Cause | Behaviour |
|---|---|---|
| Invalid file format | Committing a file with an invalid format. | Send a syslog **warning** message including the file name and the TFTP client address:<br>`The configuration file "XXX" pushed to the unit by the TFTP client "XXX" has an invalid format.`<br>No recorded settings applied. |
| File size too big | Downloading a file with a size exceeding 512000 bytes. | Send a syslog **warning** message including the file name, the TFTP client address, the file size and the maximum allowed size:<br>`The configuration file "XXX" from the TFTP client "XXX" is not downloaded because its size "XXX bytes" exceeds the maximum allowed size "XXX bytes".`<br>Send error code 3 (disk full or allocation exceeded) to the TFTP client.<br>Set *sysAdminDownloadConfigFileStatus* to *fail* and send *msTrapStatusConfigFile*. |
| TFTP transfer error | Received a TFTP error from the TFTP client. | Send a syslog **warning** message including the file name and the TFTP client address:<br>`Error in the TFTP transfer of the configuration file "XXX" from the TFTP client "XXX".`<br>Set *sysAdminDownloadConfigFileStatus* to *fail* and send *msTrapStatusConfigFile*. |
| TFTP transfer aborted | The transfer was aborted while in progress by changing the value of *sysConfigDownload ConfigMode* or *sysConfigDownloadConfigFile*. | Send a syslog warning message including the file name and the TFTP client address:<br>`The TFTP transfer of the configuration file "XXX" from the TFTP client "XXX" was aborted.`<br>Set *sysAdminDownloadConfigFileStatus* to *fail* and send *msTrapStatusConfigFile*. |
| File pulling not allowed | A TFTP client is trying to read a file from the unit. | Send a syslog informational message including the file name and the TFTP client address:<br>`The TFTP client "XXX" is trying to pull the file "XXX" from the unit. This is not allowed.`<br>Send error code 2 (access violation) to the TFTP client. |

## Syslog Messages

A syslog message is sent whenever it is impossible for the management server to download a configuration file or when it is impossible to apply the new settings to the unit.

**Table 143:** Syslog Messages Using the Management Server

| Level | Message | Event |
|---|---|---|
| Warning | `The notification "XXX" could not be sent to msHost "XXX" and msTrapPort XXX.` | A SNMP trap could not be sent to the management server. The syslog warning message includes the SNMP trap number, the management server address and port. |

**Table 143:** Syslog Messages Using the Management Server (Continued)

| Level | Message | Event |
|---|---|---|
| Informational | `Parameter values defined in the configuration file were successfully committed. Restarting the unit...` | A downloaded configuration file was successfully committed. |
| Warning | `None of the parameter values defined in the configuration file was successfully committed.` | No parameter value from the downloaded configuration file was successfully applied (e.g., because of bad OIDs). |

# Configuration File Example

The configuration file format uses XML (eXtensible Markup Language). The following is the accepted format:

```
<MX_Config_File FileId="MX_MIBFILE" MIBVersionNumber="" VersionNumber="1.0">
   <Object Prefix="" Suffix="" Value=""/>
   <Object Prefix="" Suffix="" Value=""/>
</MX_Config_File>
```

The following is an example of a configuration file:

```
<MX_Config_File FileId="MX_MIBFILE" MIBVersionNumber="1.0" VersionNumber="1.0">
   <Object Prefix="1.3.6.1.4.1.4935.15.1.8.1" Suffix="0" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.1.10.1" Suffix="0" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.1.10.10.1" Suffix="0" Value="192.168.0.10"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.1.20.1" Suffix="0" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.3.1.30.1" Suffix="0" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.3.1.30.3" Suffix="0" Value="ConfigFile.xml"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.15.1.5" Suffix="0" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.35.1.20.1.1" Suffix="3" Value="0"/>
   <Object Prefix="1.3.6.1.4.1.4935.15.35.1.20.1.1" Suffix="4" Value="0"/>
</MX_Config_File>
```

## Supported Characters

When creating and/or editing a configuration file, the following ASCII codes are supported:

| | | | | | |
|---|---|---|---|---|---|
| 10 | LF, line feed | 62 | >, greater than | 94 | ^, caret |
| 13 | CR, carriage return | 63 | ?, question mark | 95 | _, underscore |
| 32 | space | 64 | @, commercial at | 96 | `, back quote |
| 33 | !, exclamation mark | 65 | A | 97 | a |
| 34 | ", double quote | 66 | B | 98 | b |
| 35 | #, hash | 67 | C | 99 | c |
| 36 | $, dollar | 68 | D | 100 | d |
| 37 | %, percent | 69 | E | 101 | e |
| 38 | &, ampersand | 70 | F | 102 | f |
| 39 | ', quote | 71 | G | 103 | g |
| 40 | (, open parenthesis | 72 | H | 104 | h |
| 41 | ), close parenthesis | 73 | I | 105 | i |
| 42 | *, asterisk | 74 | J | 106 | j |
| 43 | +, plus | 75 | K | 107 | k |
| 44 | ,, comma | 76 | L | 108 | l |
| 45 | -, minus | 77 | M | 109 | m |
| 46 | ., full stop | 78 | N | 110 | n |
| 47 | /, oblique stroke | 79 | O | 111 | o |
| 48 | 0, zero | 80 | P | 112 | p |
| 49 | 1 | 81 | Q | 113 | q |
| 50 | 2 | 82 | R | 114 | r |
| 51 | 3 | 83 | S | 115 | s |
| 52 | 4 | 84 | T | 116 | t |
| 53 | 5 | 85 | U | 117 | u |
| 54 | 6 | 86 | V | 118 | v |
| 55 | 7 | 87 | W | 119 | w |
| 56 | 8 | 88 | X | 120 | x |
| 57 | 9 | 89 | Y | 121 | y |
| 58 | :, colon | 90 | Z | 122 | z |
| 59 | ;, semicolon | 91 | [, open square bracket | 123 | {, open curly bracket |
| 60 | <, less than | 92 | \, backslash | 124 | \|, vertical bar |
| 61 | =, equals | 93 | ], close square bracket | 125 | }, close curly bracket |
| | | | | 126 | ~, tilde |

All other ASCII codes will result in an invalid configuration file.

CHAPTER 15

# Software Download

This chapter describes how to download a software version available on the designated software server into the Mediatrix 4102.

You have the choice to perform the software download by using the TFTP, HTTPS, or HTTP protocol. You can also configure the Mediatrix 4102 to automatically update its software version.

**Note:** You can only perform a software download from the WAN interface of the Mediatrix 4102. Software downloads from the LAN side are not supported.

You can also set these parameters via the web interface, as described in "Firmware Download" on page 65.

## Before Downloading

To download a software, you may need to setup the following applications on your computer:

- TFTP server with proper root path
- MIB browser (with the current Mediatrix 4102 MIB tree)

  You can use the MIB browser built in the Media5's Unit Manager Network. See "Unit Manager Network – Element Management System" on page xxiv for more details.
- Software upgrade zip file
- SNTP server properly configured
- HTTP server with proper root path
- HTTPS server with proper root path
- Syslog daemon (optional)

### Configuring the TFTP Server

If you are to perform a software download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the software file server. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4102 in the same subnet.

It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

### Configuring the SNTP Server

If you are to use the automatic software update feature (see "Automatic Software Update" on page 249 for more details) or the HTTPS protocol, you must have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

### Configuring the HTTP Server

If you are to perform a software download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4102 in the same subnet.

It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

## Configuring the HTTPS Server

| | |
|---|---|
| **Standards Supported** | • RFC 2246 – The TLS Protocol Version 1.0<br>• RFC 2818 – HTTP Over TLS<br>• RFC 2459 – X.509 Digital Certificates<br>• RFC 3280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile |

If you are to perform a software download that requires authentication or privacy by using the HTTP over the Transport Layer Security (TLS) protocol (HTTPS), you must install a HTTPS server running on the PC designated as the server host. It is assumed that you know how to set the root path and set the SSL/TLS security configuration. If not, refer to your HTTPS server's documentation.

**Caution:** You must have a time server SNTP that is accessible and properly configured, or the automatic software update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

The Mediatrix 4102 supports the following SSL/TLS parameters:

**Table 144:** Secure Parameters Supported

| Supported Parameter | Description |
|---|---|
| Key Exchange Mechanism | • RSA<br>• Diffie-Hellman |
| Ciphers | • AES (128 and 256 bits)<br>• 3DES (168 bits) |
| Message Digests | • SHA-1 |

**Note:** Media5 recommends to use cipher suites based on the RSA key exchange mechanism, because the Diffie-Hellman key exchange mechanism introduces a noticeable delay in the HTTPS session establishment.

### Certificates

The Mediatrix 4102 contains embedded security certificates formatted as per ITU x.509 and RFC 3280. The certificates are factory-installed. The name of X.509 certificates currently installed in the Mediatrix 4102 are listed in the *securityCertificatesMIB* under the *certificateName* variable (under the *certificateTable* group). You must have at least one matching certificate on your HTTPS server.

You can also find the following information:

**Table 145:** Certificates

| Parameter | Description |
|---|---|
| certificateSubjectCommonName | The certificate's subject name.<br>If the certificate contains a subject field, display the common name. Otherwise display the first non-empty subject alternative name from the following list:<br>• Uniform Resource Locator<br>• DNS name<br>• IP Address<br>• RFC 822 name<br>See RFC 3280 sections 4.1.2.6 and 4.2.1.7 for details. |

**Table 145:** Certificates (Continued)

| Parameter | Description |
|---|---|
| certificateExpirationDate | The certificate's expiration date.<br>Display the date at which the certificate expires. The format is MM/DD/YYYY in universal time.<br>See RFC 3280 section 4.1.2.5 for details. |

When contacting a HTTPS server, the Mediatrix 4102 establishes a TLS connection by (among others):

- negotiating cipher suites
- checking the server certificates validity (dates)

The Mediatrix 4102 then checks the server's identity by validating the host name used to contact it against the information found in the server's certificate, as described in RFC 2818, section 3.1.

If any of the above does not succeed, the Mediatrix 4102 refuses the secure connection. To help detect such errors, you can increase the syslog messages level.

# Software Servers Configuration

The Mediatrix 4102 must know the IP address and port number of its Primary and Secondary software servers. These servers contain the files required for the software update. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself in static variables.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

## DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

**To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable (under the *ipAddressConfigImage* group).

   This variable defines whether the Mediatrix 4102 must ask for its Image server settings through a DHCP server or not.

2. Set the *imageSelectConfigSource* variable to **dhcp**.

   You can query the Image server's IP address and port number assigned by the DHCP server in the following read-only variables (in the *ipAddressStatus* folder):

   - imagePrimaryHost
   - imagePrimaryPort
   - imageSecondaryHost
   - imageSecondaryPort

3. Set how you want to define the Primary Image server information in the DHCP server.

**Table 146:** Primary Image Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *imageDhcpPrimarySiteSpecificCode* variable to **0**. Set the Primary image server IP address in the DHCP server inside the vendor specific sub-option 117 (hexadecimal 0x75). |

**Table 146:** Primary Image Server DHCP Information (Continued)

| To use a... | Set... |
|---|---|
| site specific code | The *imageDhcpPrimarySiteSpecificCode* variable to any value between 128 and 254. Set the Primary image server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *imageDhcpPrimarySiteSpecificCode* variable in the unit's configuration). |

See "Vendor and Site Specific DHCP Options" on page 169 for more details.

4. Set how you want to define the Secondary Image server information in the DHCP server.

**Table 147:** Secondary Image Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *imageDhcpSecondarySiteSpecificCode* variable to **0**. Set the Secondary image server IP address in the DHCP server inside the vendor specific sub-option 118 (hexadecimal 0x76). |
| site specific code | The *imageDhcpSecondarySiteSpecificCode* variable to any value between 128 and 254. Set the Secondary image server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *imageDhcpPrimarySiteSpecificCode* variable in the unit's configuration). |

See "Vendor and Site Specific DHCP Options" on page 169 for more details.

## Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable.

   This variable defines whether the Mediatrix 4102 must ask for its Image server settings through a DHCP server or not.

2. Set the *imageSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 148:** Image Static Information

| Variable | Description |
|---|---|
| imageStaticPrimaryHost | Static primary image server IP address or domain name. This is the current address of the PC that hosts the files required for the download (extracted from the zip file).<br>**Default Value**: 192.168.0.10 |
| imageStaticPrimaryPort | Static primary image server IP port number.<br>**Default Value**: 69 |
| imageStaticSecondary Host | Static secondary image server IP address or domain name. This is the current address of the PC that hosts the files required for the download (extracted from the zip file).<br>**Default Value**: 192.168.0.10 |
| imageStaticSecondaryPort | Static secondary image server IP port number.<br>**Default Value**: 69 |

The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value (69) applies to a TFTP server. It may be different for other servers. If you are using an HTTP/HTTPS server, you must change the port value to 80.

# Download Procedure

The following describes how to download a software version into the Mediatrix 4102.

**Note:** Configuration settings are not lost when upgrading the software to a newer version. However, configuration settings may be lost if you upload an older firmware to the device. See "Software Downgrade" on page 252 for more details.

You have the choice to perform the software download by using the TFTP, or HTTP, or HTTPS protocol. You can also configure the Mediatrix 4102 to automatically update its software version.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

**Note:** You can only perform a software download from the WAN interface of the Mediatrix 4102. Software downloads from the LAN side are not supported.

## Extracting the Zip File

The zip file contains the software information required for the download.

Extract the contents of the zip file on the PC designated as the software file server. Be sure to use the defined folder name. This creates a directory that contains the files required for the Mediatrix 4102 to properly update its software.

The directory name must be the same as the name defined in the *imageLocation* or *imageSelectionFileLocation* variable of the *imageMIB*. See "Setting up the Image Path" on page 243 for more details.

Media5 suggests that a folder, named identically to the software build, be available and used for the files related to that build only. Each folder should include only one delivery to ensure accuracy.

This directory must be located under the root path as defined in the TFTP/HTTP/HTTPS server or the software download will not proceed.

## Setting up the Image Path

When performing a software download, you must configure the path, on the remote image server, of the directory where you extracted the files required for the download. This applies to both the manual or automatic download procedure, using the HTTP, HTTPS, or TFTP protocol.

The directory must be located under the root path, as defined in the TFTP, or HTTP, or HTTPS server, or the software download will not proceed. See "Before Downloading" on page 239 for more details.

The Mediatrix 4102 first downloads a file called "setup.inf". This file contains the list of all the other files to download, depending on the product. The "setup.inf" file and all the other files must be in the same directory. If any of the files is missing, the procedure will not work properly.

**To setup the Image path:**

1. In the *imageMIB*, select where to get the image location in the *imageLocationProvisionSource* variable.

   You have the following choices:

**Table 149:** Image Location Parameters

| Parameter | Description |
|---|---|
| static | Uses the directory specified in the *imageLocation* variable (see Step 2). |
| remoteFile | The image location is defined in a file called “mediatrix4102targetimage.inf” . The location of this file is defined in the *imageSelectionFileLocation* variable.<br>This is useful if you are using automatic updates with multiple units (see Step 3). |

2. If you have set the *imageLocationProvisionSource* variable to **static** (see Step 1), configure the path in the *imageLocation* variable.

   This is the location of the “setup.inf” file that contains the list of the files to download into the Mediatrix 4102. The “setup.inf” file and all the other files must be in the same directory. In other words, this is the path from the root TFTP/HTTP/HTTPS folder down to the files extracted from the zip file.

   Note that the path must contain a maximum of 63 characters.

3. If you have set the *imageLocationProvisionSource* variable to **remoteFile** (see Step 1):

   *a.* Create a text file and write the path and/or name of the directory that contains the files required for download. Save this file as “mediatrix4102targetimage.inf” under the server root path.

**Note:** If you leave the file empty, the Mediatrix 4102 will look for the software download information in the root directory of the image server.

   *b.* Configure the path of the “mediatrix4102targetimage.inf” file in the *imageSelectionFileLocation* variable.

   Note that the selection file name is in lower case. Some web servers are case sensitive. The path must contain a maximum of 63 characters.

   This is useful if you are using automatic updates with multiple units. If you want the units to download a new version, you only have to change the path once in the “mediatrix4102targetimage.inf” file. If you were to use the *imageLocation* variable, you would have to change the path in every unit.

Let’s consider the following example:

- The directory that contains the files required for download is called: **SIP_v5.0.1.1_MX-S5001-01**.
- This directory is under **C:/Root/Download**.

**Table 150:** Path Configurations Example

| Root Path | Corresponding Path Name |
|---|---|
| c:/root/download | SIP_v5.0.1.1_MX-S5001-01 |
| c:/ | root/download/SIP_v5.0.1.1_MX-S5001-01 |
| c:/root | download/SIP_v5.0.1.1_MX-S5001-01 |

The following are some tips to help your download process:

- Use the “/” character when defining the path to indicate sub-directories. For instance, *root/download*.

  If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the “/” character and produce an error. In this case, use the “\” character.
- Use basic directory names, without spaces or special characters such as “~”, “@”, etc., which

may cause problems.

- Cut and paste the path and/or name of the directory that contains the extracted files into the image path of the Mediatrix 4102 (you may have to convert "\" into "/") to eliminate typographical errors.

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

## Software Download Status

You can validate the status of the software download in various ways.

### Syslog Messages

If you are using a Syslog daemon, you will receive messages that inform you of the software update status. The following are the syslog messages the Mediatrix 4102 sends:

**Table 151:** Software Update Syslog Messages

| Level | Message | Event |
|---|---|---|
| **General Messages** | | |
| Informational | `The software update succeeded.` | The software update has been successful. |
| Error | `The software update failed.` | The software update experienced an error and has not been completed. |
| Error | `The software update failed (xxx).` | An error occurs when updating the software, internal error code provided. |
| Warning | `Primary image server not specified, cannot download file: xxx` | This error occurs when an image download is initiated and no domain name or address is specified for the primary image server. |
| Warning | `Secondary image server not specified, cannot download file: xxx.` | When a request involving the primary server fails, the secondary server is tried.<br>This error occurs when there is no address or domain name specified for the secondary image server. |
| Error | `Cannot resolve address of image server: xxx.` | A DNS request failed to resolve the domain name of the image server (primary or secondary). |
| Error | `Target image at location: xxx from host: xxx is invalid or corrupted.` | For periodic and automatic updates, the target image to download is first compared with the installed image. This error occurs when this comparison failed because of corruption in the target image files. |
| Informational | `Image download transfer initiated.` | When manual, periodic or "at restart" image download is initiated. |
| Warning | `The file: xxx from host: xxx exceeds the size limit.` | The selection file or "setup.inf" file received exceeds 10000 bytes. |
| Informational | `Target image at location: xxx from host xxx is identical to currently installed image. Transfer aborted.` | For periodic and automatic updates, the target image to download is first compared with the installed image. This message occurs when this comparison determined that the target image is identical to the installed image. |
| Error | `Image does not support hardware (error %d)` | The software download failed because the software image is not compatible with the hardware. |

**Table 151:** Software Update Syslog Messages (Continued)

| Level | Message | Event |
|---|---|---|
| **HTTP-Specific Messages** | | |
| Warning | `HTTP image transfer of file: xxx from host: xxx was closed by peer.` | The HTTP transfer was closed by the peer. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was closed due to unsupported or malformed response from the host.` | In the HTTP response, one of the following error occurred:<br>• The protocol version is not 1.0 or 1.1.<br>• Some field or line is not properly formatted.<br>• The trailing <crlf> is not present at the end of the header.<br>• Unsupported kind of response. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because of a malformed or incompatible request.` | When receiving HTTP response #400 or #403. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because of a server error.` | When receiving HTTP response #500 or #501. |
| Warning | `HTTP image transfer of file: xxx from host: xxx was refused because service is unavailable.` | When receiving HTTP response #503. |
| **TFTP-Specific Messages** | | |
| Warning | `Image transfer of file: xxx from host: xxx and port: xxx was closed due to unexpected error` | Unexpected error, either internal or on a TFTP or HTTP connection. |
| Warning | `Image transfer of file: xxx from host: xxx port: xxx was closed after timeout` | When not receiving TFTP packets for 32 seconds or not receiving a HTTP packet for 15 seconds. |
| Warning | `Image transfer. File: xxx not found on host: xxx` | When receiving TFTP error “NOT FOUND” or HTTP response #404. |
| Warning | `Image transfer. Access to file: xxx on host: xxx is unauthorized` | When receiving TFTP error “ACCESS” or HTTP response #401. |

If the local syslog messages are enabled (see “Local Syslog” on page 374 for more details), you can view these messages on the web interface.

## LED States

When the Mediatrix 4102 initiates a software download, the LEDs located on the front panel indicate the status of the process.

**Table 152:** LED States in Software Download

| Event | LED State |
|---|---|
| Image downloading and writing | *Power*, *LAN*, *In Use* and *Ready* LEDs blink alternately at 1 Hz with 1/4 ON duty cycle.<br>**Warning**: Do not turn the Mediatrix 4102 off while in this state. |

**Table 152:** LED States in Software Download (Continued)

| Event | LED State |
|---|---|
| Image download failed | *Power*, *LAN*, *In Use* and *Ready* LEDs blink at the same time at 2 Hz with 50% ON duty cycle for 4 seconds. |

See "LED Indicators" on page 16 for a detailed description of the LED patterns related to the software download process.

### MIB Variable

You can validate the result of the last software update by checking the state of the *sysAdminLastDownloadSoftware* MIB variable.

## Download via TFTP

The following steps explain how to download a software by using the TFTP protocol.

In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Downloading a Software Version*.

▶ **To download a software via TFTP:**

1. If not already done, setup the Image server used to download the software (see "Before Downloading" on page 239).
2. Be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.
3. If not already done, configure the Image path as described in "Setting up the Image Path" on page 243.
4. If not already done, configure the image hosts and ports as defined in "Software Servers Configuration" on page 241.
5. Set the TFTP root path in your TFTP server.

   It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.
6. Set the *imageTransferProtocol* variable to **tftp**.
7. Set the *groupSetAdmin* variable (in the *groupAdminMIB*) to **ForceLock**.

   All activities in progress on the Mediatrix 4102 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is "locked"). The software upgrade may take place.

   The Mediatrix 4102 lines will be unlocked after successfully downloading the software and restarting. If, for any reason, the software download is not successful, you must manually unlock the lines as per "Lines Administrative State" on page 255.
8. Initiate the download by setting the *sysAdminCommand* variable (in the *sysAdminMIB*) to **downloadSoftware**.

   This starts the download process.

**Caution:** Never shutdown the Mediatrix 4102 manually while in the download process, because the image may be partially written and the Mediatrix 4102 is unable to restart.

   The software download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

   If Transparent Address Sharing is enabled during the software download, the PC connected to the Mediatrix 4102 may experience momentary loss of Internet connectivity.
9. Update the MIB browser with the MIB version coming with the software version.

## Download via HTTP/HTTPS

The following steps explain how to download a software by using the HTTP protocol.

The following steps explain how to download a software by using the HTTP or HTTPS protocol. If you are using HTTPS, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 240 for more details.

▶ **To download a software via HTTP or HTTPS:**

1. If not already done, setup the Image server used to download the software (see "Before Downloading" on page 239).
2. If not already done, configure the Image path as described in "Setting up the Image Path" on page 243.
3. If not already done, configure the image hosts and ports as defined in "Software Servers Configuration" on page 241.

**Caution:** When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Software Servers Configuration" on page 241 for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to **http** or **https**.

   Your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.
5. If your HTTP or HTTPS server requires authentication, set the following:
   - The user name in the *imageTransferUsername* variable.
   - The password in the *imageTransferPassword* variable.
6. Set the *groupSetAdmin* variable (in the *groupAdminMIB*) to **ForceLock**.

   All activities in progress on the Mediatrix 4102 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is "locked"). The software upgrade may take place.

   The Mediatrix 4102 lines will be unlocked after successfully downloading the software and restarting.

   If, for any reason, the software download is not successful, you must manually unlock the lines as per "Lines Administrative State" on page 255.
7. Initiate the download by setting the *sysAdminCommand* variable (in the *sysAdminMIB*) to **downloadSoftware**.

   This starts the download process.

**Caution:** Never shutdown the Mediatrix 4102 manually while in the download process, because the image may be partially written and the Mediatrix 4102 is unable to restart.

   The software download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

   If Transparent Address Sharing is enabled during the software download, the PC connected to the Mediatrix 4102 may experience momentary loss of Internet connectivity.
8. Update the MIB browser with the MIB version coming with the software version.

### User Agent Header of HTTP Requests

The *User-Agent* header field of an HTTP request contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

```
User-Agent: Softphone Beta1.5
```

You can customize the information that the Mediatrix 4102 sends when establishing a communication.

▶ **To customize the HTTP User Agent header of HTTP requests:**

1. In the *interopMIB*, set the *mxInteropHttpUAHeaderConfig* variable with the proper macro.

   The following macros are replaced by their representation:

   - **%version%**: Version of the application.
   - **%mac%**: Unit MAC address (lowercase).
   - **%rev%**: Hardware revision number.
   - **%product%**: Product name.
   - **%%**: A '%' sign.

2. Restart the Mediatrix 4102 so that the changes may take effect.

## Automatic Software Update

You can configure the Mediatrix 4102 to automatically update its software version. This update can be done:

- Every time the Mediatrix 4102 restarts.
- At a specific time interval you can define.

### Automatic Update on Restart

The Mediatrix 4102 may download a new software version each time it restarts.

▶ **To set the automatic update every time the Mediatrix 4102 restarts:**

1. If not already done, setup the Image server used to download the software (see "Before Downloading" on page 239).
2. If not already done, configure the Image path as described in "Setting up the Image Path" on page 243.
3. If not already done, configure the image hosts and ports as defined in "Software Servers Configuration" on page 241.

**Caution:** When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Software Servers Configuration" on page 241 for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to either **http**, **https**, or **tftp**.

   If you are using the HTTPS protocol, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 240 for more details.

   If you are using the HTTP or HTTPS protocol to download the software, be aware that your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.

5. If you are using the HTTP or HTTPS protocol and your HTTP or HTTPS server requires authentication, set the following:
   - The user name in the *imageTransferUsername* variable.

- The password in the *imageTransferPassword* variable.

6. Set the *imageAutoUpdateOnRestartEnable* variable to **enable**.
7. Set the *imageAutoUpdateEnable* variable to **enable**.

   The automatic software update will be performed each time the Mediatrix 4102 restarts.

## Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4102 to download a software version at a specific day and/or time.

**To set the automatic update at a specific time interval:**

1. If not already done, setup the Image server used to download the software (see "Before Downloading" on page 239).
2. If not already done, configure the Image path as described in "Setting up the Image Path" on page 243.
3. If not already done, configure the image hosts and ports as defined in "Software Servers Configuration" on page 241.

**Caution:** When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See "Software Servers Configuration" on page 241 for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to either **http**, **https**, or **tftp**.

   If you are using the HTTPS protocol, the Mediatrix 4102 must contain the proper certificate. See "Configuring the HTTPS Server" on page 240 for more details.

   If you are using the HTTP or HTTPS protocol to download the software, be aware that your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.
5. If you are using the HTTP or HTTPS protocol and your HTTP or HTTPS server requires authentication, set the following:
   - The user name in the *imageTransferUsername* variable.
   - The password in the *imageTransferPassword* variable.
6. Define the time base for automatic software updates in the *imageAutoUpdateTimeUnit* variable (in the *imageAutomaticUpdate* group).

   You have the following choices:

**Table 153:** Time Unit Parameters

| Parameter | Description |
|---|---|
| minutes | Updates the software every *x* minutes. You can specify the *x* value in the variable *imageAutoUpdatePeriod* (see Step 7). |
| hours | Updates the software every *x* hours. You can specify the *x* value in the variable *imageAutoUpdatePeriod* (see Step 7). |
| days | Updates the software every *x* days. You can specify the *x* value in the variable *imageAutoUpdatePeriod* (see Step 7).<br>You can also define the time of day when to perform the update in the *imageAutoUpdateTimeRange* variable (see Step 8). |

7. Set the waiting period between each software update in the *imageAutoUpdatePeriod* variable.

   The time unit for the period is specified by the *imageAutoUpdateTimeUnit* variable (see Step 6). Available values are from 1 to 48.

8. If you have selected **days** in Step 6, set the time of the day when to initiate a software update in the *imageAutoUpdateTimeRange* variable.

   The time of the day is based on the *sntpTimeZoneString* variable setting (see "Chapter 21 - SNTP Settings" on page 317 for more details).

   You must have a time server SNTP that is accessible and properly configured, or the automatic software update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to "Chapter 21 - SNTP Settings" on page 317 for more details on how to configure the Mediatrix 4102 for a SNTP server.

   If a time range is specified, the unit will initiate the image software download at a random time within the interval specified.

   The format should be one of the following:

   ```
   hh[:mm[:ss]]
   hh[:mm[:ss]] - hh[:mm[:ss]]
   ```

   Where:

   ```
   hh: Hours.
   mm: Minutes.
   ss: Seconds.
   ```

   The image software download is initiated at the first occurrence of this value and thereafter with a period defined by *imageAutoUpdatePeriod*. Let's say for instance the automatic update is set with the time of day at 14h00 and the update period at every 2 days.

   - If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.
   - If the time range is set to '14:00 - 15:00' and the automatic update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

9. Set the *imageAutoUpdateEnable* variable to **enable**.

   If one of the telephones/faxes is off-hook, the Mediatrix 4102 will perform the download five minutes after both ports are detected on-hook.

## Spanning Tree Protocol (STP)

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. When a software download occurs, the *LAN* and *WAN* connectors of the Mediatrix 4102 may switch off. This shutdown may trigger these network switches to shutdown the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent software download.

To prevent this, the Mediatrix 4102 supports the STP. However, this management has a potential time cost. It may appear from time to time that software downloads take more time. This is normal.

The following is an example where the STP management impacts the download duration.

- The software download procedure does not use any DHCP and DNS services.
- The primary image server is down (or not properly configured).
- The secondary image server is up and running well.

In this case, the Mediatrix 4102 tries to contact the primary image server. As it is not available, the Mediatrix 4102 retries for two minutes. It contacts the secondary server after that period and starts the software download.

**Note:** When using the Mediatrix 4102, Media5 recommends to disable the Spanning Tree Protocol on the network to which the unit is connected. See also "DHCP Options Waiting Time" on page 167.

# Software Downgrade

It is possible to downgrade a Mediatrix 4102 from the current version (for instance, v5.0rx.x) to an older version (for instance, v4.4rx.x).

**Note:** If you perform a default reset on the Mediatrix 4102, you must download the current version into the unit before performing the software downgrade procedure.

**To perform a software downgrade:**

1. Create, in a common folder under the TFTP root path, the current (for instance, v5.0) and older (for instance, v4.4) applications folders.
2. Re-update the Mediatrix 4102 with the current application.

   The Mediatrix 4102 runs the current software version (v5.0rx.x).
3. Perform the software downgrade to the older application (v4.4rx.x) as described in "Download Procedure" on page 243.

# Emergency Software Procedure

If the software download is suddenly interrupted, it may not be complete. Without any protection against this situation, the Mediatrix 4102 is not functional.

A transfer may be interrupted for the following reasons:

- An electrical shortage.
- The user of the Mediatrix 4102 can accidentally power off the unit.

Depending on the moment when the software download has been interrupted, the emergency software procedure (also called rescue application) can automatically start a new software download to repair the software if it has been corrupted by the interruption. However, there is a small but critical time frame during which unrecoverable errors could happen. This is why it is very important that the unit is not turned off during software downloads.

## Using the Emergency Software

When the emergency software procedure starts, the following steps apply:

1. The Mediatrix 4102 tries to initiate the software download with the primary software server.
2. If the software download fails with the primary software server, the Mediatrix 4102 tries to initiate the software download with the secondary software server.
3. If the primary and the secondary servers cannot be reached, the Mediatrix 4102 tries two default servers: 192.168.0.10 and then 192.168.0.2.

   If, for some reason, it is impossible to rescue the unit by using the primary and secondary servers, setting up a server at one of these addresses within the correct subnet will provide an ultimate way to rescue the unit. However, if these addresses cannot be reached from the unit's subnet, the default gateway must provide appropriate routing to them.
4. If the software download also fails with the two default servers, the Mediatrix 4102 idles for one minute.
5. After this one minute, the Mediatrix 4102 tries to initiate the software download again.
6. If the software download fails again with the primary, secondary, and default software servers, the Mediatrix 4102 idles for two minutes before attempting to initiate the software download.

**7.** If the emergency software download still fails, the Mediatrix 4102 tries to initiate the software download again by doubling the delay between each attempt up to a maximum of 16 minutes:

- first attempt: 1 minute delay
- second attempt: 2 minutes delay
- third attempt: 4 minutes delay
- fourth attempt: 8 minutes delay
- fifth attempt: 16 minutes delay
- sixth attempt: 16 minutes delay
- etc.

This procedure continues until the software download completes successfully. The software download can fail if the software server cannot be reached or if the software directory is not found on the software server.

CHAPTER 16

# Line Configuration

This chapter describes the features available on the lines connected to the Mediatrix 4102.

For information on voice codecs, see "Chapter 17 - Voice Transmissions" on page 263.

For information on data codecs, see "Chapter 18 - Fax Transmission" on page 279.

## Lines Administrative State

You can independently set the administrative state of each analog line of your Mediatrix 4102. This state determines how the Mediatrix 4102 processes calls.

For instance, you must properly unlock the two analog lines of the Mediatrix 4102 to properly make and receive calls on all of them.

The administrative states may be applied in two ways:

- Temporary: The administrative state is applied immediately, but it is not kept after the Mediatrix 4102 restarts.
- Permanent: When the Mediatrix 4102 restarts, it reads a MIB variable to determine the administrative state defined for each analog line.

### Temporary Administrative State

You can set the administrative state of a line that will be kept until the Mediatrix 4102 restarts. Once the unit restarts, it uses the permanent state defined for each line. See "Permanent Administrative State" on page 256 for more details.

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

You can also set these parameters via the web interface, as described in "Interface Management" on page 48.

**To set a temporary administrative state:**

1. In the *ifAdminMIB*, locate the *ifAdminSetAdmin* variable.

   This variable temporary locks/unlocks the selected line of the Mediatrix 4102. This state is kept until the unit restarts. It offers the following settings:

**Table 154:** Temporary Lock Settings

| Setting | Description |
|---|---|
| unlock | Registers the line to the SIP server. |
| lock | Cancels the line registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated. |
| forcelock | Cancels the line registration to the SIP server. All active calls in progress are terminated immediately. No new calls may be initiated. |

### Permanent Administrative State

The permanent administrative state is applied every time the Mediatrix 4102 restarts.

▶ **To set a permanent administrative state:**

1. In the *ifAdminMIB*, locate the *ifAdminInitialAdminState* variable.

   This variable indicates the administrative state the current analog line will have after the Mediatrix 4102 restarts. It offers the following settings:

**Table 155:** Permanent Lock Settings

| Setting | Description |
|---|---|
| unlocked | Registers the line to the SIP server. |
| locked | The analog line is unavailable for normal operation. It cannot be used to make and/or receive calls. |

## Unregistered Line Behaviour

You can specify whether a line should remain enabled or not when not registered. This is useful if you want your users to be able to make calls even if the line is not registered with the SIP server.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To specify unregistered line behaviour:**

1. In the *sipExperimentalMIB*, locate the *sipUnregisteredPortBehavior* variable.

   The following values are available:

**Table 156:** Unregistered Line Behaviour

| Value | Description |
|---|---|
| disablePort | When the line is not registered, it is disabled. The user cannot make or receive calls. Picking up the handset yields a fast busy tone, and incoming INVITEs receive a "403 Forbidden" response. |
| enablePort | When the line is not registered, it is still enabled. The user can receive and initiate outgoing calls. Note that because the line is not registered to a registrar, its public address is not available to the outside world; the line will most likely be unreachable except through direct IP calling. |

## Flash Hook Detection

The flash hook can be described as quickly depressing and releasing the plunger in or the actual handset-cradle to create a signal indicating a change in the current telephone session. Services such as picking up a call waiting, second call, call on hold, and conference are triggered by the use of the flash hook.

A flash hook is detected when the hook switch is pressed for a shorter time than would be required to be interpreted as a hang-up.

Using the "flash" button that is present on many standard telephone handsets can also trigger a flash hook.

The Mediatrix 4102 allows you to set the minimum and maximum time within which pressing and releasing the plunger is actually considered a flash hook.

▶ **To set flash hook parameters:**

1. In the *fxsMIB*, set the following variables:

**Table 157:** Flash Hook Parameters

| Variable | Description |
|---|---|
| fxsFlashHookDetectionDelayMin | Minimum time in ms the hook switch must remain pressed to perform a flash hook.<br>**Default Value**: 100 |
| fxsFlashHookDetectionDelayMax | Maximum time in ms the hook switch can remain pressed to perform a flash hook.<br>**Default Value**: 1200 |

2. Restart the Mediatrix 4102 so that the changes may take effect.

# Source Line Selection

The source line selection feature defines a list of callers that have the right to use a specific FXS line to make a call. This feature can be used to map an FXS line to a specific FXO line of a gateway such as the Mediatrix 1204. See "Examples of Source Line Selection Use" on page 257 for more details.

▶ **To set the line selection:**

1. In the *lineSelectionMIB*, define the list of telephone numbers that can use this line to make calls in the *lineSelectionDigitMap* variable.

   Call sources that match this digit map can use this line. This string must follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). This digit map will not have any effect unless the feature's status is "enabled".

   Because this variable is located in a table, you can define different digit maps for each line of the Mediatrix 4102.

2. Enable the line selection feature by setting the *lineSelectionEnable* variable to **enable**.

   The source of the call is compared to all the source line selection digit maps defined in the previous step. The result of this comparison is a list of lines that can take the call, but are not necessarily available to do so.

   Because this variable is located in a table, you can enable/disable the feature on a per-line basis.

## Examples of Source Line Selection Use

### FXS to FXO Line Mapping

You can map an FXS line to a specific FXO line of a gateway such as the Mediatrix 1204. In this case, a call made from this FXS line will always use the same FXO line. To achieve that, the Mediatrix 4102 and Mediatrix 1204 configurations would be something similar to the following:

```
Mediatrix 4102: IP address 192.168.0.1
sipUAMainUsername = 6661111 (FXS line #1)
sipUAMainUsername = 6662222 (FXS line #2)


Mediatrix 1204: IP address 192.168.0.2
lineSelectionDigitMap (FXO line #1) = 6661111
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6661111
lineSelectionDigitMap (FXO line #2) = 6662222
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222
```

```
lineSelectionDigitMap (FXO line #3) = 6663333
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222
lineSelectionDigitMap (FXO line #4) = 6664444
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222
```

With such a configuration, a call made from line #2 of a Mediatrix 4102 is processed on line #2 of the Mediatrix 1204. On the other hand, if a caller from the SCN calls line #3 of the Mediatrix 1204, the call is automatically redirected to line #3 of the Mediatrix 4102.

### Reserving an FXS Line

You can reserve an FXS line for specific individuals. For instance, these individuals could be the management team members of a company.

If the telephone numbers of the management team are 221 and 222 and you want to reserve an FXS line for their exclusive use, configure the Mediatrix 4102 as follows:

```
lineSelectionDigitMap (FXS line #1) = (221|222)
lineSelectionDigitMap (FXS line #2) = xxx
```

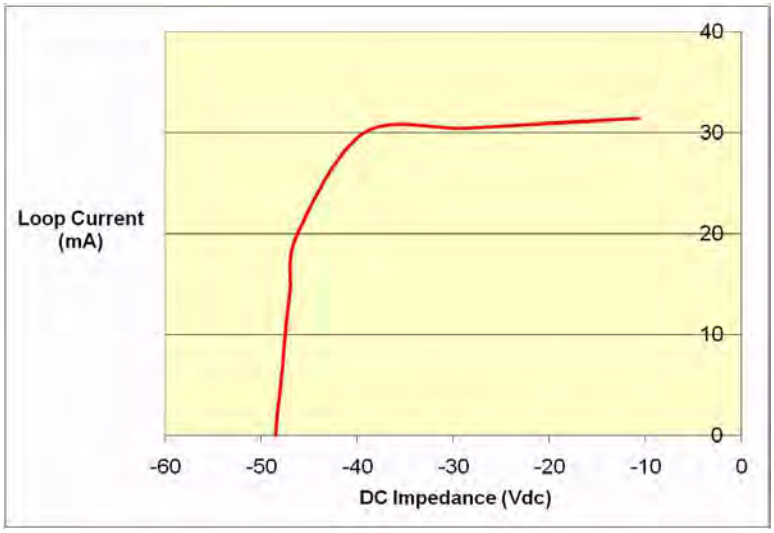
The management team can thus use all FXS lines, while others can only use lines 2,3 and 4.

# Loop Current

When one of its analog lines goes off-hook, the Mediatrix 4102 controls the line in a fixed loop current mode. The value of the loop current can be modified through the MIB.

Note that the actual measured current may be different than the value you set, because it varies depending on the DC impedance. This is illustrated in Figure 93 for a loop current of 32 mA.

**Figure 93:** Loop Current vs Impedance – 32 mA




▶ **To set the loop current:**

1. In the *fxsMIB,* set the *fxsLoopCurrent* variable to the value you want to use.

   The loop current is in mA. The range of available values is from 20 mA to 32 mA.

2. If applicable, configure the Mediatrix 4102 to suppress loop current on its lines when they cannot be used or when the IP connection is lost by setting the *fxsLoopCurrentDropEnable* variable to **enable**.

   The loop current is interrupted on the port as soon as it enters an unusable state, or if the unit's IP network connection is lost.

If you set the value to **disable**, the loop current is unaffected by the port usability.

3. Restart the Mediatrix 4102 so that the changes may take effect.

When a remote end-user goes on-hook, the Mediatrix 4102 signals the far end disconnect by performing a current loop drop (< 1 mA) on the analog line. This current loop drop, also referred to as “Power Denial” mode, is typically used for disconnect supervision on analog lines. The Mediatrix 4102 maintains a current drop for one second (this value cannot be configured), then a busy tone is generated to indicate the user to hang up.

# Callee Hang-up Supervision

This feature determines whether call clearing occurs as soon as the called user is the first to hang up a received call or after a user-defined delay.

This feature allows to emulate the behaviour of some SCNs that delay the ending of a call when the callee hangs up first.

When the feature is activated, hanging up on a received call will not terminate the call right away. Instead, the connection will remain active for a user-defined amount of time. If the callee picks up the phone before the expiration of that delay, he is still in communication with the caller (assuming the caller has not already hung up).

**Note:** This feature is effective only when the user is the called party. When the user acts as the calling party, the call ends as soon as the user hangs up. It is also effective only with a single active call. Callee hang-up supervision has no effect when the callee hangs up with a call on hold, wants to perform a call transfer, or if he is the initiator of a three-way conference call.

▶ **To enable the callee hang-up supervision:**

1. In the *fxsMIB,* set the *fxsCalleeHangupDelay* variable with the amount of time, in seconds, the Mediatrix 4102 waits after the called user hangs up before signalling the end of the call.

   The default value is **60** seconds.

2. Enable the callee hang-up supervision feature by setting the *fxsCalleeHangupSupervision* variable to **enable**.

   When the user is the first to hang up on a received call, the Mediatrix 4102 waits for the amount of time set in the *fxsCalleeHangupDelay* variable before signalling the end of the call.

# Line Reversal

Two options are available to determine how the line polarity is used to signal the beginning and end of a call. They are used because of an inability by some customer's CPE to react to busy tone. When one of the options is activated, it replaces the default behaviour of the Mediatrix 4102, which is to briefly remove power from the line to signal that the remote party has hung up.

▶ **To enable line reversal:**

1. In the *fxsMIB,* set the *fxsPolarityAndDenialBehavior* variable with the proper parameter.

**Table 158:** Line Reversal Parameters

| Parameter | Description |
|---|---|
| noReversal | Polarity reversal is not used. This is the default value. A short power drop is made at the end of a call when the call is disconnected by the remote party. The drop duration can be configured in the variable *fxsPowerDropOnDisconnectDuration* (Step 2). |
| reversalOnIdle | Activates the Reversal on Idle feature. When a line is in the idle, line lockout, or blocked state, the polarity of the line is reversed. When the line is seized to originate a call, or the ring current is applied to terminate a call, the line polarity is returned to normal. On release, the line polarity is returned to the reversed condition. |
| reversalOnEstablished | Activates the Reversal on Established option. The polarity remains normal (voltage on the phone line stays positive) when you pick up the phone, dial, and when the device is ringing. As soon as the other party picks up the phone, the unit reverses the line polarity. When the unit receives a hang-up signal or the caller hangs up the phone, the voltage returns to its positive state. |

2. Set the *fxsPowerDropOnDisconnectDuration* variable with the power drop duration, in milliseconds, that is made at the end of a call when the call is disconnected by the remote party.

   This variable only has an effect when the variable *fxsPolarityAndDenialBehavior* is set to **noReversal**.

# Blanking of an Anonymous Caller ID

You can instruct the Mediatrix 4102 to blank out the name portion of the received caller ID whenever “anonymous” (case insensitive) is used to identify the originator of the call before the caller ID data is sent to the telephone. If the variable is not activated, the anonymous string is passed on to the telephone and displayed to identify the calling party.

▶ **To enable anonymous caller ID blanking:**

1. In the *fxsMIB,* set the *fxsBlankAnonymousCallerId* variable to **enable**.

# Calling Number Transformation

It is possible to modify the value of the Caller ID before it is sent on the line. The Caller ID must match the criteria specified in the *fxsCallingNumberCriteria* variable (which must contain a regular expression). The transformation specified in the *fxsCallingNumberTransformation* variable is then applied.

▶ **To** modify the value of the Caller ID**:**

1. In the *fxsMIB,* set the *fxsCallingNumberCriteria* variable with the expression that the calling number must match in order to apply the transformation.

   The criteria must be a regular expression. See "Regular Expressions" on page 261 for more details.

   If you leave this variable empty, the feature is disabled.

2. Set the *fxsCallingNumberTransformation* variable with the transformation to apply on the calling number if it matches the expression specified in *fxsCallingNumberCriteria*.

   You can use the "\0" to "\9" groups in the string. They are replaced by the corresponding group in the regular expresion specified in *fxsCallingNumberCriteria*. See "Groups" on page 262 for more details.

## Regular Expressions

| **Standards Supported** | • IEEE Std 1003.1-2001: IEEE Standard for Information Technology---Portable Operating System Interface (POSIX®) |
|---|---|

A regular expression is a string used to find and replace strings in other large strings. The Mediatrix 4102 uses regular expressions to enter a value in the *fxsCallingNumberCriteria* variable, often by using wildcard characters. These characters provide additional flexibility and decrease the need for multiple entries in configuring number ranges.

The expression cannot begin by "^", it is implicit in the expression. The following table shows some of the wildcard characters that are supported:

**Table 159:** Regular Expressions Wildcards

| Character | Description |
|---|---|
| . | Single-digit place holder. For instance, `555 ....` matches any dialed number beginning with 555, plus at least four additional digits. Note that the number may be longer and still match. |
| * | Repeats the previous digit 0, 1, or more times. For instance, in the pattern:<br>`1888*1`<br>the pattern matches:<br>`1881, 18881, 188881, 1888881`<br>**Note:** If you are trying to handle the asterisk (*) as part of a dialed number, you must use `\*`. |
| [ ] | Range of digits.<br>• A consecutive range is indicated with a hyphen (-), for instance, `[5-7]`.<br>• A nonconsecutive range is indicated without a delimiter, for instance, `[58]`.<br>• Both can be used in combination, for instance `[5-79]`, which is the same as `[5679]`.<br>You may place a (^) symbol right after the opening bracket to indicate that the specified range is an exclude list. For instance, `[^01]` specifies the same range as `[2-9]`.<br>**Note:** The Mediatrix 4102 only supports single-digit ranges. You cannot specify the range of numbers between 99 and 102 by using `[99-102]`. |
| ( ) | Indicates a pattern (also called group), for instance, `555(2525)`. It is used when replacing a number in a mapping. See "Groups" on page 262 for more details. |

**Table 159:** Regular Expressions Wildcards (Continued)

| Character | Description |
|---|---|
| ? | Matches 0 or 1 occurrence of the previous item. For instance, `123?4` matches both `124` and `1234`. |
| + | Repeats the previous digit one or more time. For instance `12+345` matches `12345`, `122345`, etc. (but not `1345`). If you use the + at the end of a number, it repeats the last number one or more times. For instance: `12345+` matches, `12345`, `123455`, `1234555`, etc. |

The matching criterion implicitly matches from the beginning of the string, but not necessarily up to the end. For instance, `123` will match the criterion `1`, but it will not match the criterion `2`.

If you want to match the whole string, you must end the criterion with "$". For instance, `123` will not match the criterion `1$` and will match the criterion `123$`.

## Groups

A group is placed within parenthesis. It is used when replacing a string in a mapping. You can use up to nine groups (defined by "\1" to "\9") and matching is not case sensitive. "\0" represents the whole string. Lets say for instance you have the following string:

```
9(123(45)6)
```

The following describes how the groups are replaced:

**Table 160:** Groups Replacement Example

| Replacement | Result |
|---|---|
| \0 | 9123456 |
| \1 | 123456 |
| \2 | 45 |
| \3 | |

CHAPTER 17

# Voice Transmissions

This chapter describes the various codecs the Mediatrix 4102 supports for transmitting audio signals.

You can also set some of these parameters via the web interface, as described in "Voice & Fax Codecs" on page 93.

## Codec Descriptions

The two lines of the Mediatrix 4102 can simultaneously use the same codec (for instance, G.711 PCMA), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

**Table 161:** Codecs Comparison

| | Compression | Voice Quality |
|---|---|---|
| **G.711** | None | Excellent |
| **G.726** | Medium | Fair |
| **G.729a/ab** | High | Fair/Good |

### G.711 PCMA and PCMU

Specified in ITU-T Recommendation G.711. The audio data is encoded as 8 bits per sample, after logarithmic scaling. PCMU denotes µ-law scaling, PCMA A-law scaling.

**Table 162:** G.711 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. See "Packetization Time" on page 266 for more details. |
| Voice Activity Detection (VAD) | Can be enabled or disabled. When enabled, two levels of detection are available: transparent or conservative. See "G.711 and G.726 VAD" on page 275 for more details. |
| Comfort noise | Supports white and custom comfort noise as defined in *RFC 3389*. See "Comfort Noise" on page 277 for more details. |

### Analog Modem

The Mediatrix 4102 can send modem transmissions in clear channel (G.711). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

## G.726

Specified in ITU-T Recommendation G.726: 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM). It describes the algorithm recommended for conversion of a single 64 kbit/s A-law or U-law PCM channel encoded at 8000 samples/sec to and from a 40, 32, 24, or 16 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique.

**Table 163:** G.726 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. The preferred is 20 ms. See "Packetization Time" on page 266 for more details. |
| Voice Activity Detection (VAD) | Uses the G.711 VAD settings. Can be enabled or disabled. When enabled, two levels of detection are available: transparent or conservative. See "G.711 and G.726 VAD" on page 275 for more details. |
| Comfort noise | Uses the G.711 comfort noise settings. Supports white and custom comfort noise as defined in *RFC 3389*. See "Comfort Noise" on page 277 for more details. |

### Analog Modem

The Mediatrix 4102 can send modem transmissions in clear channel (G.726). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

## G.729

Specified in ITU-T Recommendation G.729, coding of speech at 8 kbit/s using conjugate structure-algebraic code excited linear prediction (CS-ACELP). For all data rates, the sampling frequency (and RTP timestamp clock rate) is 8000 Hz.

A voice activity detector (VAD) and comfort noise generator (CNG) algorithm in Annex B of G.729 is recommended for digital simultaneous voice and data applications; they can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B comfort noise frame occupies 2 octets.

The Mediatrix 4102 supports G.729A and G.729AB for encoding and G.729, G.729A and G.729AB for decoding.

**Table 164:** G.729 Features

| Feature | Description |
|---|---|
| Packetization time | Range of 10 ms to 100 ms with increment of 10 ms. See "Packetization Time" on page 266 for more details. |
| Voice Activity Detection (VAD) | The Mediatrix 4102 supports the annex B. Annex B is the built-in support of VAD in G.729. See "G.729 VAD" on page 276 for more details. |

# Preferred Codec

The preferred codec is the codec you want to favour during negotiation.

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

**To set a preferred codec:**

1. In the *voiceIfMIB,* locate the *voiceIfCodecPreferred* variable (*voiceIfCodecTable*).
   This variable sets the preferred codec for this line.
2. Choose the codec you want to use from one of the available configurations:
   - pcmu
   - pcma
   - g729
   - g726-16kbps
   - g726-24kbps
   - g726-32kbps
   - g726-40kbps

   The default value is **pcmu**.

# Enabling Individual Codecs

Enabling individual codecs allows you to define codecs that can be considered during negotiation. If codecs are disabled, they are not considered.

**To enable voice codecs:**

1. In the *voiceIfMIB,* choose the codec you want to use (*voiceIfCodecTable*).
   You have the choice between the following codecs:

**Table 165:** Enabling Voice Codecs

| Codec | Variable | Set to... |
|---|---|---|
| PCMU (G.711 u-Law) | voiceIfCodecPcmuEnable | enable |
| PCMA (G.711 a-Law) | voiceIfCodecPcmaEnable | enable |
| G.726 at 16 kbps | voiceIfCodecG72616kbpsEnable | enable |
| G.726 at 24 kbps | voiceIfCodecG72624kbpsEnable | enable |
| G.726 at 32 kbps | voiceIfCodecG72632kbpsEnable | enable |
| G.726 at 40 kbps | voiceIfCodecG72640kbpsEnable | enable |
| G.729.A | voiceIfCodecG729Enable | enable |

2. If you have enabled one or more of the G.726 codecs, set the G.726 actual RTP dynamic payload type used in an initial offer in one or more of the following variables:
   - voiceIfCodecG72616kbpsPayloadType: The default value is 97.
   - voiceIfCodecG72624kbpsPayloadType: The default value is 98.
   - voiceIfCodecG72632kbpsPayloadType: The default value is 99.
   - voiceIfCodecG72640kbpsPayloadType: The default value is 100.

The payload types available are as per RFC 3551. The values range from 96 to 127.

**Note:** When selecting the dynamic payload type, make sure that the value is not already used by another dynamic codec. If a value between 96 and 127 is refused, this means it is already used by another dynamic codec.

**Note:** If you set the *voiceIfDtmfTransport* variable to **outOfBandUsingSignalingProtocol** ("DTMF Transport Type" on page 268), you cannot configure a dynamic payload type to 111 because it is already used by the DTMF out-of-band using signalling protocol.

3. Restart the Mediatrix 4102 so that the changes may take effect.

# Packetization Time

The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

▶ **To set the packetization time:**

1. In the *voiceIfMIB,* set the packetization time of the codec(s) as required (*voiceIfCodecTable*). Available values vary from one codec to another.

**Table 166:** Packetization Time Settings

| Variable | Definition | Values (ms) |
|---|---|---|
| **PCMU (G.711 u-Law)** | | |
| voiceIfCodecPcmuMinPTime | Shortest packetization period allowed for the PCMU codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecPcmuMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecPcmuMaxPTime | Longest packetization period allowed for the PCMU codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecPcmuMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |
| **PCMA (G.711 a-Law)** | | |
| voiceIfCodecPcmaMinPTime | Shortest packetization period allowed for the PCMA codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecPcmaMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecPcmaMaxPTime | Longest packetization period allowed for the PCMA codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecPcmaMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |

**Table 166:** Packetization Time Settings (Continued)

| Variable | Definition | Values (ms) |
|---|---|---|
| **G.726** | | |
| voiceIfCodecG72616kbpsMinPTime | Shortest packetization period allowed for the G.726-16kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecG72616kbpsMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecG72616kbpsMaxPTime | Longest packetization period allowed for the G.726-16kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecG72616kbpsMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |
| voiceIfCodecG72624kbpsMinPTime | Shortest packetization period allowed for the G.726-24kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecG72624kbpsMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecG72624kbpsMaxPTime | Longest packetization period allowed for the G.726-24kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecG72624kbpsMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |
| voiceIfCodecG72632kbpsMinPTime | Shortest packetization period allowed for the G.726-32kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecG72632kbpsMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecG72632kbpsMaxPTime | Longest packetization period allowed for the G.726-32kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecG72632kbpsMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |
| voiceIfCodecG72640kbpsMinPTime | Shortest packetization period allowed for the G.726-40kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecG72640kbpsMaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecG72640kbpsMaxPTime | Longest packetization period allowed for the G.726-40kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecG72640kbpsMinPTime* variable.<br>**Default Value**: 100 | 10-100, with increments of 10 |

**Table 166:** Packetization Time Settings (Continued)

| Variable | Definition | Values (ms) |
|---|---|---|
| **G.729** | | |
| voiceIfCodecG729MinPTime | Shortest packetization period allowed for the G.729 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the *voiceIfCodecG729MaxPTime* variable.<br>**Default Value**: 10 | 10-100, with increments of 10 |
| voiceIfCodecG729MaxPTime | Longest packetization period allowed for the G.729 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the *voiceIfCodecG729MinPTime* vartiable.<br>**Default Value**: 100 | 10-100, with increments of 10 |

**Note:** The packetization time is not negotiated between endpoints, so a minimum and a maximum don't make much sense. The selected value is the default RTP value (20 ms for G.711, G.726, and G.729.AB) if it is included in the range delimited by the minimum and maximum. Otherwise, it is the minimum.

# DTMF Transport Type

| | |
|---|---|
| **Standards Supported** | • draft-choudhuri-sip-info-digit-00.txt<br>• ITU-T Recommendation Q.24 : Multifrequency push-button signal reception<br>• RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals |

You can define how to transport the DTMFs.

▶ **To set the DTMF transport type:**

1. In the *voiceIfMIB,* set the DTMF transport type in the *voiceIfDtmfTransport* variable (*voiceIfDtmfTransportTable* group).

   The following choices are available:

**Table 167:** DTMF Transport Type Parameters

| Transport Parameter | Description |
|---|---|
| inBand | The DTMFs are transmitted like the voice in the RTP stream. |

## DTMF out-of-band

Certain compression codecs such as G.729 effectively distort voice because they lose information from the incoming voice stream during the compression and decompression phases. For normal speech this is insignificant and becomes unimportant. In the case of pure tones (such as DTMF) this distortion means the receiver may no longer recognize the tones. The solution is to send this information as a separate packet to the other endpoint, which then plays the DTMF sequence back by re-generating the true tones. Such a mechanism is known as out-of-band DTMF.The Mediatrix 4102 receives and sends out-of-band DTMFs as per ITU Q.24. DTMFs supported are 0-9, A-D, *, #.

**Table 167:** DTMF Transport Type Parameters (Continued)

| Transport Parameter | Description |
|---|---|
| outOfBandUsingRtp | The DTMFs are transmitted as per RFC 2833 (see "DTMF Payload Type" on page 270 for additional information). |
| outOfBandUsingSignalingProtocol | The DTMFs are transmitted as per *draft-choudhuri-sip-info-digit-00.txt* (see "DTMF Transport Using SIP INFO" on page 269 and "DTMF Transport over the SIP Protocol" on page 271 for more details).<br>**Note**: This feature and the Hook Flash processing feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See "Hook Flash Processing" on page 352 for more details. |
| signalingProtocolDependent | The signalling protocol has the control to select the DTMF transport mode. The SDP body includes both RFC 2833 and *draft-choudhuri-sip-info-digit-00.txt* in that order of preference. |

## DTMF Transport Using SIP INFO

| | |
|---|---|
| **Standards Supported** | • RFC 2976: The SIP INFO Method<br>• draft-choudhuri-sip-info-digit-00.txt |

You can use the SIP INFO method to collect and transport DTMFs. The collection process is regarded as being an unsolicited one-character timer-less digit collection.

When the feature is enabled:

- The Mediatrix 4102 sends a separate SIP INFO method every time a digit is entered during the call.
- The Mediatrix 4102 plays each DTMF sent in a separate message upon receiving a valid SIP INFO message.

**To enable the DTMF transport using the SIP INFO feature:**

1. In the *voiceIfMIB*, set the DTMF transport type in the *voiceIfDtmfTransport* variable (*voiceIfDtmfTransportTable* group) according to the transport type you want to use.

   There are three methods to transport DTMF events:

   - in-band
   - out-of-band using RTP (RFC 2833)
   - out-of-band using SIP INFO

**Table 168:** Transport Type Setting

| To | Set the variable to: |
|---|---|
| Transport DTMF events in-band. | inBand |
| Transport DTMF events out-of-band by exclusively using RTP (RFC 2833). | outOfBandUsingRtp |
| Transport DTMF events out-of-band by exclusively using the SIP INFO method. | outOfBandUsingSignalingProtocol |
| Offer the choice to transport DTMF events out-of-band by using either RTP or the SIP INFO method. | signalingProtocolDependent |

If you set the *voiceIfDtmfTransport* variable to **signalingProtocolDependent**, the remote party must select one of the two transport types. Transporting DTMF by using RTP has priority over the SIP INFO method.

## DTMF Payload Type

| Standards Supported | • RFC 1890 – RTP Profile for Audio and Video Conferences with Minimal Control |
|---|---|

When selecting the *outOfBandUsingRtp* DTMF transport mode (see "DTMF Transport Type" on page 268 for more details), you can determine the actual RTP dynamic payload type used for the "telephone-event" in an initial offer. The payload types available are as per RFC 1890.

▶ **To define the DTMF payload type:**

1. In the *voiceIfMIB*, set the DTMF transport type in the *voiceIfDtmfTransport* variable (*voiceIfDtmfTransportTable* group) to **outOfBandUsingRtp**.
2. Set the payload type in the *voiceIfDtmfPayloadType* variable.

   Available values range from 96 to 127.

## DTMF – RFC 2833 Events

| Standards Supported | • RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals |
|---|---|

You can define which events will be relayed via RFC 2833. This could be very useful in a Remote Line Extension scenario, as described in "Remote Line Extension" on page 355.

▶ **To define the DTMF enforce default events:**

1. In the *voiceIfMIB*, set the DTMF enforce default events in the *voiceIfDtmfEnforceDefaultEvents* variable.

**Table 169:** DTMF Enforce Default Events

| Parameter | Description |
|---|---|
| enable | Conformance is enforced and support for RFC 2833 implies the support of basic telephony-events. When setting the variable *voiceIfDtmfTransport* to **outOfBandUsingRtp** ("DTMF Transport Type" on page 268), or the variable *telephonyAttributesHookFlashProcessing* to **outOfBandUsingRtp** ("Hook Flash Processing" on page 352), the Mediatrix 4102 will advertise the support for events 0-15; it will assume support for events 0-15 when support for RFC 2833 is received in an announcement. |
| disable | This creates a deliberate deviance to RFC 2833 as support of basic events is not automatic. The variables *voiceIfDtmfTransport* and *telephonyAttributesHookFlashProcessing* then act independently to specify which events will be relayed via RFC 2833. If Hook Flash relay is enabled by itself, support of event 16 alone will be advertised; if both Hook Flash and DTMF relay are activated, events 0-16 are supported. |

## DTMF Transport over the SIP Protocol

| **Standards Supported** | • draft-choudhuri-sip-info-digit-00.txt |
|---|---|

You can select the method used to transport DTMFs out-of-band over the SIP protocol.

This feature is effective only if the *voiceIfDtmfTransport* variable is set to **outOfBandUsingSignalingProtocol** (see "DTMF Transport Type" on page 268 for more details).

▶ **To select the DTMF transport method over the SIP protocol:**

1. In the *voiceIfMIB*, set the DTMF transport type in the *voiceIfDtmfTransport* variable to **outOfBandUsingSignalingProtocol**.
2. In the *sipInteropMIB*, set the DTMF transport type in the *sipInteropDtmfTransportMethod* variable (*sipInteropDtmfTransportBySipProtocol* group).

   The following methods are available:

   **Table 170:** DTMF Out-of-Band Transport Methods

| Method | Description |
|---|---|
| draftChoudhuriSipInfoDigit00 | Transmits DTMFs by using the method defined in *draft-choudhuri-sip-info-digit-00*. Only the unsolicited-digit part is supported. |
| infoDtmfRelay | Transmits DTMFs by using a custom method. This custom method requires no SDP negotiation and assumes that the other peer uses the same method.<br>It uses a SIP INFO message with a content of type *application/dtmf-relay*. The body of the message contains the DTMF transmitted and the duration of the DTMF:<br>`Signal= 1`<br>`Duration= 160`<br>When transmitting, the duration is the one set in the *sipInteropDtmfTransportDuration* variable (see Step 3 below).<br>When receiving, the duration of the DTMF received is not used. The value is the one set in the *analogScnGwDtmfDuration* variable (see Step 4 below).<br>DTMFs are transmitted one at a time. It is thus not compatible with the PIN dialing feature (see "PIN Dialing" on page 354 for more details).<br>Available digits are "0123456789ABCD*#". The Mediatrix 4102 also supports the ",;p" characters when receiving DTMFs. |

3. Set the DTMF duration sent in the INFO message when using the **infoDtmfRelay** method to transmit DTMFs in the *sipInteropDtmfTransportDuration* variable.

   This value is expressed in milliseconds (ms). The default value is **100** ms.
4. In the *analogScnGwMIB*, set the DTMF duration when using the **infoDtmfRelay** method to receive DTMFs in the *analogScnGwDtmfDuration* variable.

   This is the duration, in milliseconds (ms), a DTMF is played when dialing the destination phone number.
5. Set an inter-digit dial delay in the *analogScnGwInterDigitDial Delay* variable.

   This is the delay, in milliseconds (ms), between two DTMFs when dialing the destination phone number. This is useful when the Mediatrix 4102 receives DTMFs out-of-band faster than it can signal them.
6. Restart the Mediatrix 4102 so that the changes may take effect.

# DTMF Detection

The default DTMF detection parameters of the Mediatrix 4102 may sometimes not be enough to properly detect the DTMFs. This section describes how to set additional DTMF detection parameters.

## DTMF Frequencies

The DTMF keypad is laid out in a 4x4 matrix, with each row representing a low frequency, and each column representing a high frequency. For example, pressing a single key (such as '1') sends a sinusoidal tone of the two frequencies (697 Hz and 1209 Hz). When the unit is configured to send DTMFs out-of-band, its DSP detects these DTMFs, removes them from the RTP stream, and sends them out-of-band.

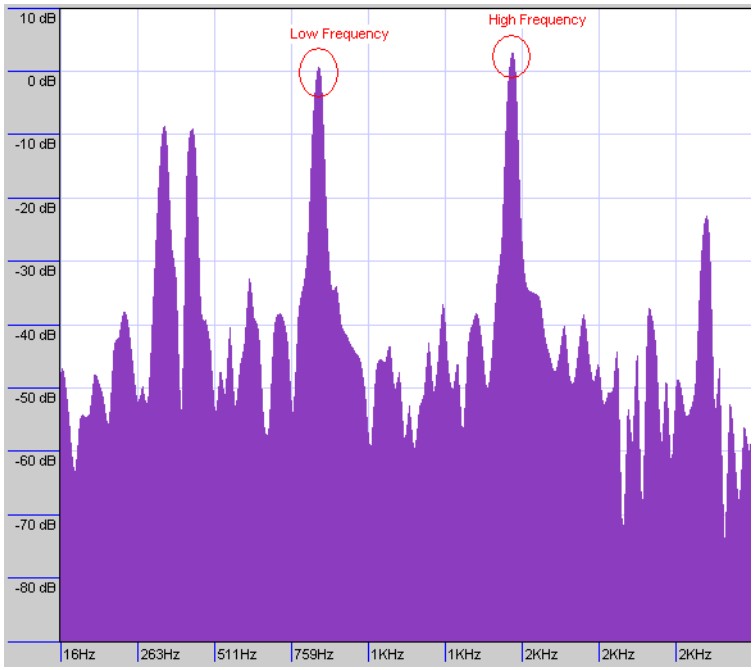
**Table 171:** DTMF Keypad Frequencies

| Low/High (Hz) | 1209 | 1336 | 1477 | 1633 |
|---|---|---|---|---|
| **697** | 1 | 2 | 3 | A |
| **770** | 4 | 5 | 6 | B |
| **852** | 7 | 8 | 9 | C |
| **941** | * | 0 | # | D |

## DTMF Detection Configuration

Below is a frequency spectrum analysis of a DTMF (9) with the Frequency in Hertz on the x axis and the Power in dBm on the y axis. The low and high frequencies of the DTMF are in red and you can clearly see that they are the most powerful frequencies in the signal.

**Figure 94:** DTMF Detection Example



To detect this DTMF, the DSP relies on several parameters. The following table lists the default values that the Mediatrix 4102 uses. You can override any one of these values.

**Table 172:** DTMF Detection Default Parameters

| Parameter | Value |
|---|---|
| MinPowerThreshold | -30 dBm0 |
| MaxPowerThreshold | 1 dBm0 |
| BreakPowerThreshold | 32 dBm0 |
| PositiveTwist | 6 dBm0 |
| NegativeTwist | 9 dBm0 |
| RiseTimeCriteria | confirmSnr |

▶ **To configure the DTMF detection:**

1. In the *voiceIfMIB*, set the maximum absolute power threshold (dBm0) for the low and high frequencies in a DTMF in the *voiceIfDtmfDetectionUnitMaxPowerThreshold* variable.

   The high AND low DTMF frequencies MUST be lower than this threshold otherwise the DTMF is not detected.

   Raising this value increases the sensitivity of the DTMF detection. Raising this value too high may also cause false detections of DTMFs.

2. Set the minimum absolute power threshold (dBm0) for the low and high frequencies in a DTMF in the *voiceIfDtmfDetectionUnitMinPowerThreshold* variable.

   The high AND low DTMF frequencies MUST be higher than this threshold otherwise the DTMF is not detected.

   You could, for instance use one of the following settings:

   - -15 dBm0: This configuration detects even more false DTMFs in the voice pattern.
   - -20 dBm0: This configuration detects more DTMFs in the voice pattern.
   - -35 dBm0: his configuration detects less DTMFs in the voice pattern.
   - -40 dBm0: This configuration detects even less DTMFs in the voice pattern.

   Raising this value reduces the sensitivity to DTMF detection.

3. Set the break absolute power threshold (dBm0) for on-off transition of a DTMF in the *voiceIfDtmfDetectionUnitBreakPowerThreshold* variable.

   While a DTMF has been positively detected, the DTMF will be considered OFF as soon as the high OR low frequency in the DTMF gets below this threshold.

4. Set the *voiceIfDtmfDetectionUnitPositiveTwist* variable.

   When the high-group frequency of a DTMF is more powerful than the low-group frequency, the difference between the high-group frequency absolute power and the low-group frequency absolute power must be smaller than or equal to the value set in this variable. Otherwise, the DTMF is not detected.

   Raising this value increases the sensitivity of DTMF detection. Raising this value too high may also cause false detections of DTMFs.

5. Set the *voiceIfDtmfDetectionUnitNegativeTwist* variable.

   When the low frequency of a DTMF is more powerful than the high frequency, the difference between the low frequency absolute power and the high frequency absolute power MUST be smaller than or equal to the value set in this variable. Otherwise, the DTMF is not detected.

   Raising this value increases the sensitivity of DTMF detection. Raising this value too high may also cause false detections of DTMFs.

6. Define how the Rise Time criteria should be configured for DTMF detection in the *voiceIfDtmfDetectionRiseTimeCriteria* variable.

**Table 173:** RiseTimeCriteria Parameters

| Parameter | Description |
|---|---|
| checkSr | Enable the Step Rise criteria and disable the Confirm DTMF SNR criteria. |
| confirmSnr | Enable the Confirm DTMF SNR criteria and disable the Step Rise criteria. |

The Step Rise criteria compares the current frame energy to the high frequency power of the previous frame. If the current frame energy is high enough, then it passes the test, further validating the DTMF.

Disabling the Step Rise criteria may result in deteriorated talk-off performance, but increases the detection of malformed DTMF.

The Confirm DTMF SNR criteria is an additional Signal-to-noise ratio test performed before a confirmed DTMF report is sent to finally validate the DTMF.

# Adaptative Jitter Buffer

The jitter buffer allows better protection against packet loss, but increases the voice delay. If the network to which the Mediatrix 4102 is connected suffers from a high level of congestion, the jitter buffer protection level should be higher. If the network to which the Mediatrix 4102 is connected suffers from a low level of congestion, the jitter buffer protection level should be lower.

**Note:** Do not put **0** as values for the *voiceIfTargetJitterBufferLength* and *voiceIfMaxJitterBufferLength* variables.

**To set Jitter Buffer variables:**

1. In the *voiceIfMIB*, locate the *voiceIfTable* group.
2. Define the jitter buffer length in the *voiceIfTargetJitterBufferLength* variable.

   The adaptive jitter buffer attempts to hold packets to the target holding time. This is the minimum delay the jitter buffer adds to the system. The target jitter buffer length is in ms and must be equal to or smaller than the maximum jitter buffer.

   Values range from 0 ms to 135 ms. The default value is 30 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

   It is best not to set target jitter values below the default value. Setting a target jitter buffer below 5 ms could cause an error. Jitter buffer adaptation behaviour varies from one codec to another. See "About Changing Jitter Buffer Values" on page 275 for more details.
3. Define the maximum jitter buffer length in the *voiceIfMaxJitterBufferLength* variable.

   This is the maximum jitter the adaptive jitter buffer can handle. The jitter buffer length is in ms and must be equal to or greater than the target jitter buffer.

   Values range from 0 ms to 135 ms. The default value is 125 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

   The maximum jitter buffer value should be equal to the minimum jitter buffer value + 4 times the ptime value. Let's say for instance that:

   - Minimum jitter buffer value is 30 ms
   - Ptime value is 20 ms

     The maximum jitter buffer value should be: 30 + 4x20 = 110 ms

   See "About Changing Jitter Buffer Values" on page 275 for more details.
4. Restart the Mediatrix 4102 so that the changes may take effect.

### About Changing Jitter Buffer Values

Media5 recommends to avoid changing the target and maximum jitter buffer values unless experiencing or strongly expecting one of the following symptoms:

- If the voice is scattered, try to increase the maximum jitter buffer value.
- If the delay in the voice path (end to end) is too long, you can lower the target jitter value, but ONLY if the end-to-end delay measured matches the target jitter value.

  For instance, if the target jitter value is 50 ms, the maximum jitter is 135 ms and the delay measured is 130 ms, it would serve nothing to reduce the target jitter. However, if the target jitter value is 100 ms and the measured delay is between 100 ms and 110 ms, then you can lower the target jitter from 100 ms to 30 ms.

# Voice Activity Detection

The Voice Activity Detection (VAD) defines how the Mediatrix 4102 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, the VAD may affect packets that are not really silent (for instance, cut sounds that are too low). The VAD can thus slightly affect the voice quality.

## G.711 and G.726 VAD

The G.711/G.726 VAD is generic – when enabling VAD, G.711/G.726 sends speech frames only during periods of audio activity. During silence periods, G.711/G.726 does not send speech frames, but it may send Comfort Noise (CN) packets (payload 13) containing information about background noise.

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

**To enable G.711 and G.726 VAD:**

1. In the *voiceIfMIB*, locate the *voiceIfTable* group.
2. Define the sensitivity of the VAD algorithm to silence periods in the *voiceIfG711VoiceActivityDetectionEnable* variable.

   The following settings are available:

   **Table 174:** G.711/G.726 VAD Settings

| Setting | Description |
|---|---|
| Disable | VAD is not used. |
| Transparent | VAD is enabled. It has low sensitivity to silence periods. |
| Conservative | VAD is enabled. It has normal sensitivity to silence periods. |

   The difference between transparent and conservative is how “aggressive” the algorithm considers something as an inactive voice and how “fast” it stops the voice stream. A setting of conservative is a little bit more aggressive to react to silence compared to a setting of transparent.

   The default value is **conservative**.
3. Restart the Mediatrix 4102 so that the changes may take effect.

### G.729 VAD

G.729 has a built-in VAD in its Annex B version. It is recommended for digital simultaneous voice and data applications and can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B frame occupies 2 octets. The CN packets are sent in accordance with annex B of G.729.

▶ **To enable G.729 VAD:**

1. In the *voiceIfMIB*, locate the *voiceIfTable* group.
2. Define the *voiceIfG729VoiceActivityDetectionEnable* variable.

   The following settings are available:

**Table 175:** G.729 VAD Settings

| Setting | Description |
|---|---|
| disable | G.729 uses annex A only. The Mediatrix 4102 does not send G.729 Annex B comfort noise frames. |
| enable | G.729 annex A is used with annex B. The Mediatrix 4102 sends G.729 Annex B comfort noise frames during silence periods. |

See "Enabling Individual Codecs" on page 265 for more details.

## Echo Cancellation

Echo cancellation eliminates the echo effect caused by signal reflections. An echo is a signal that has been reflected or otherwise returned with enough magnitude and delay to be perceived. The echo cancellation is usually an active process in which echo signals are measured and cancelled or eliminated by combining an inverted signal with the echo signal.

You can disable the echo cancellation feature of the Mediatrix 4102, which may be useful to ensure the success of some modem transmissions.

▶ **To enable or disable echo cancellation:**

1. In the *voiceIfMIB*, set the *voiceIfEchoCancellationEnable* variable to the proper value.

**Table 176:** Echo Cancellation Parameters

| Parameter | Description |
|---|---|
| disable | The DSP does not use echo cancellation on the related port. |
| enable | The DSP proceeds to cancel signals that are recognized as echo when appropriate. This is the default value. |

### Signal Limiter

If you are experiencing echo issues, you may use the Signal Limiter. When enabled, the Signal Limiter attenuates or removes peaks in the voice signal.

**Caution:** This may reduce the presence of echo but may impair DTMF detection because of the signal distorsion.

▶ **To use the Signal Limiter:**

1. In the *voiceIfMIB*, set the *voiceIfSignalLimiterLevel* variable to the proper value.

**Table 177:** Signal Limiter Parameters

| Parameter | Description |
|---|---|
| 0 | Disable signal limiter |
| 1 | Most aggressive |
| ... | |
| 5 | Least aggressive |

# Comfort Noise

| **Standards Supported** | • RFC 3389: Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN) |
|---|---|

Comfort Noise (CN) defines how the Mediatrix 4102 processes silence periods information it receives.

**Note:** Comfort noise only applies to the G.711 and G.726 codecs. G.729 CNG is not configurable because it is part of the codec.

During silence periods, the Mediatrix 4102 may receive CN packets containing information about background noise. When enabling Comfort Noise Generation (CNG), those packets are used to generate local comfort noise.

▶ **To enable Comfort Noise:**

1. In the *voiceIfMIB*, locate the *voiceIfTable* group.
2. Define the type of comfort noise in the *voiceIfG711ComfortNoiseGenerationEnable* variable.
   The following settings are available:

**Table 178:** Comfort Noise Settings

| Setting | Description |
|---|---|
| disable | CNG disabled. |
| whiteNoise | CNG enabled – white noise. |
| customNoise | CNG enabled – custom noise. More elaborated background noise that sounds better than white comfort noise. |

3. Restart the Mediatrix 4102 so that the changes may take effect.

# User Gain

The user gain allows you to modify the input and output sound level of the Mediatrix 4102.

**Caution:** Use these settings with great care. Media5 recommends not to modify the user gain variables unless absolutely necessary because default calibrations may not be valid anymore.

Modifying user gains may cause problems with DTMF detection and voice quality – using a high user gain may cause sound saturation (the sound is distorted). Furthermore, some fax or modem tones may not be recognized anymore. The user gains directly affect the fax communication quality and may even prevent a fax to be sent.

You can compensate with the user gain if there is no available configuration for the country in which the Mediatrix 4102 is located. Because the user gain is in dB, you can easily adjust the loss plan (e.g., if you need an additional 1 dB for analog to digital, simply put 1 for user gain input).

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

**To set user gain variables:**

1. In the *voiceIfMIB*, locate the *voiceIfTable* group.
2. Define the following variables:
   - *voiceIfUserInputGainOffset*: User input gain offset in dB (from analog to digital).
   - *voiceIfUserOutputGainOffset*: User output gain offset in dB (from digital to analog).

   Values range from -30 dB to +20 dB. However, going above +6 dB may introduce clipping/distortion depending on the country selected. Under -24 dB, you will not have much signal either.
3. Restart the Mediatrix 4102 so that the changes may take effect.

CHAPTER 18

# Fax Transmission

This chapter describes how to perform fax transmissions in clear channel and T.38 with the Mediatrix 4102.

You can also set some of these parameters via the web interface, as described in "Voice & Fax Codecs" on page 93.

## Introduction

The Mediatrix 4102 handles G3 fax transmissions at speeds up to 14.4 kbps. Automatic fax mode detection is standard on all lines. Real-Time Fax Over UDP with the T.38 protocol stack is also available.

The quality of T.38 fax transmissions depends upon the system configuration, type of call control system used, type of Mediatrix units deployed, as well as the model of fax machines used. Should some of these conditions be unsatisfactory, performance of T.38 fax transmissions may vary and be reduced below expectations.

A fax call works much like a regular voice call, with the following differences:

1. The fax codec may be re-negotiated by using a re-INVITE.
2. The goal of the re-INVITE is to allow both user agents to agree on a fax codec, which is either:
   a. Clear channel (PCMU/PCMA or G.726) without Echo Cancellation nor Silence Suppression (automatically disabled).
   b. T.38.
3. Upon fax termination, if the call is not BYE, the previous voice codec is recovered with another re-INVITE.

All lines of the Mediatrix 4102 can simultaneously use the same codec (for instance, T.38), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

## Fax Calling Tone Detection

You can enable the fax calling tone (CNG tone) detection.

▶ **To enable fax calling tone detection:**

1. In the *dataIfMIB*, set the *dataIfCngToneDetectionEnable* variable to **enable**.

   Upon recognition of the CNG tone, the Mediatrix 4102 switches the communication from voice mode to fax mode and the CNG is transferred by using the preferred fax codec.

   This option allows for quicker fax detection, but it also increases the risk of false detection.

   If you do not want the Mediatrix 4102 to detect the fax calling tone, set the variable to **disable**. In this case, the CNG tone does not trigger a transition from voice to data and the CNG is transferred in the voice channel.

   With this option, faxes are detected later, but the risk of false detection is reduced.

## CED Fax Tone Detection

You can define the behaviour of the Mediatrix 4102 upon reception of a CED fax tone from the network.

▶ **To enable CED fax calling tone detection:**

1. In the *dataIfMIB*, set the *dataIfCedFaxToneEnable* variable with the proper value.

**Table 179:** CED Fax Tone Detection Parameters

| Parameter | Description |
|---|---|
| enable | Upon reception of a CED tone following a sent CNG tone, the unit switches the communication from voice mode to fax mode and the CED is transferred by using the preferred fax codec. |
| disable | The CED tone does not trigger a transition from voice to data and the CED is transferred in the voice channel. |

This configuration has no effect if the *dataIfCngToneDetectionEnable* variable is set to **enable**.

## Analog CED Detection Behaviour

You can define the behaviour of the Mediatrix 4102 upon reception of a CED fax tone from the analog port.

▶ **To define the analog CED detection behaviour:**

1. In the *dataIfMIB*, set the *dataIfAnalogCedDetectionBehavior* variable with the proper value.

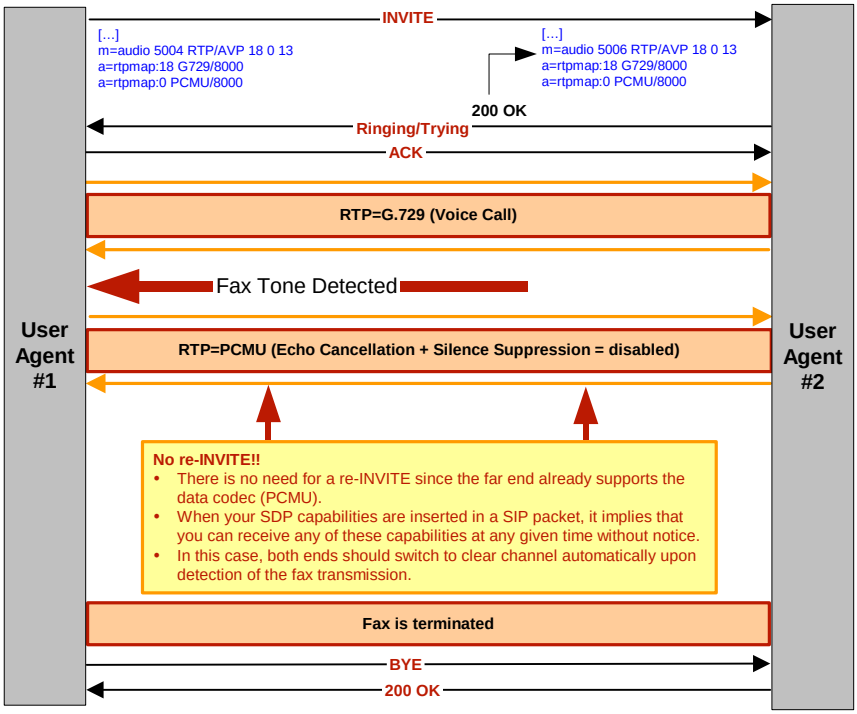
**Table 180:** Analog CED Detection Behaviour Parameters

| Parameter | Description |
|---|---|
| passthrough | The CED tone does not trigger a transition from voice to data and is transferred in the voice channel. Use this setting when any kind of analog device (i.e.: telephone, fax or modem) can be connected to this port. |
| faxmode | Upon reception of a CED tone, the unit switches the communication from voice mode to fax mode and the CED is transferred by using the preferred fax codec. Only a fax can then be connected to this port. |

# Clear Channel Fax

The Mediatrix 4102 can send faxes in clear channel. The following is a clear channel fax call flow:

**Figure 95:** Clear Channel Fax Call Flow




▶ **To set a preferred clear channel fax transmission codec:**

1. Set the clear channel codec to use upon detecting a fax tone in the *dataIfClearChannelCodecPreferred* variable.

   This variable is used to decide which of the following codecs is preferred, even for voice transmissions:

   - PCMU
   - PCMA
   - G.726 at 32 kbs
   - G.726 at 40 kbs

**Note:** In clear channel, G.726 at 16 kbs and 24 kbs are not available for fax transmission.

**Note:** If you want to set the G.726 codec at 32 kbs or at 40 kbs as the preferred clear channel codec, you must also select the corresponding G.726 codec as the preferred voice codec as described in "Preferred Codec" on page 265. Otherwise, the Mediatrix 4102 will fail to switch to the G.726 codec for clear channel faxes because G.726 is not negotiated.

   - noPreferredCodec

     When *noPreferredCodec* is selected and no data-capable codecs are negociated, data transmission may fail.

It has an impact only if a codec other than PCMU, or PCMA or G.726 is chosen in the *voiceIfCodecPreferred* variable (see "Preferred Codec" on page 265). For instance, if G.729 is the preferred voice codec, then PCMU, and PCMA and G.726 are ordered following the *dataIfClearChannelCodecPreferred* setting.

Clear channel faxes use the negotiated codec, regardless of the setting applied to *dataIfClearChannelCodecPreferred*.

This variable increases the relative priority of the selected codec vs other data-capable codecs. However, the priority of the preferred clear channel codec remains lower than the *voiceIfCodecPreferred* variable (see "Preferred Codec" on page 265).

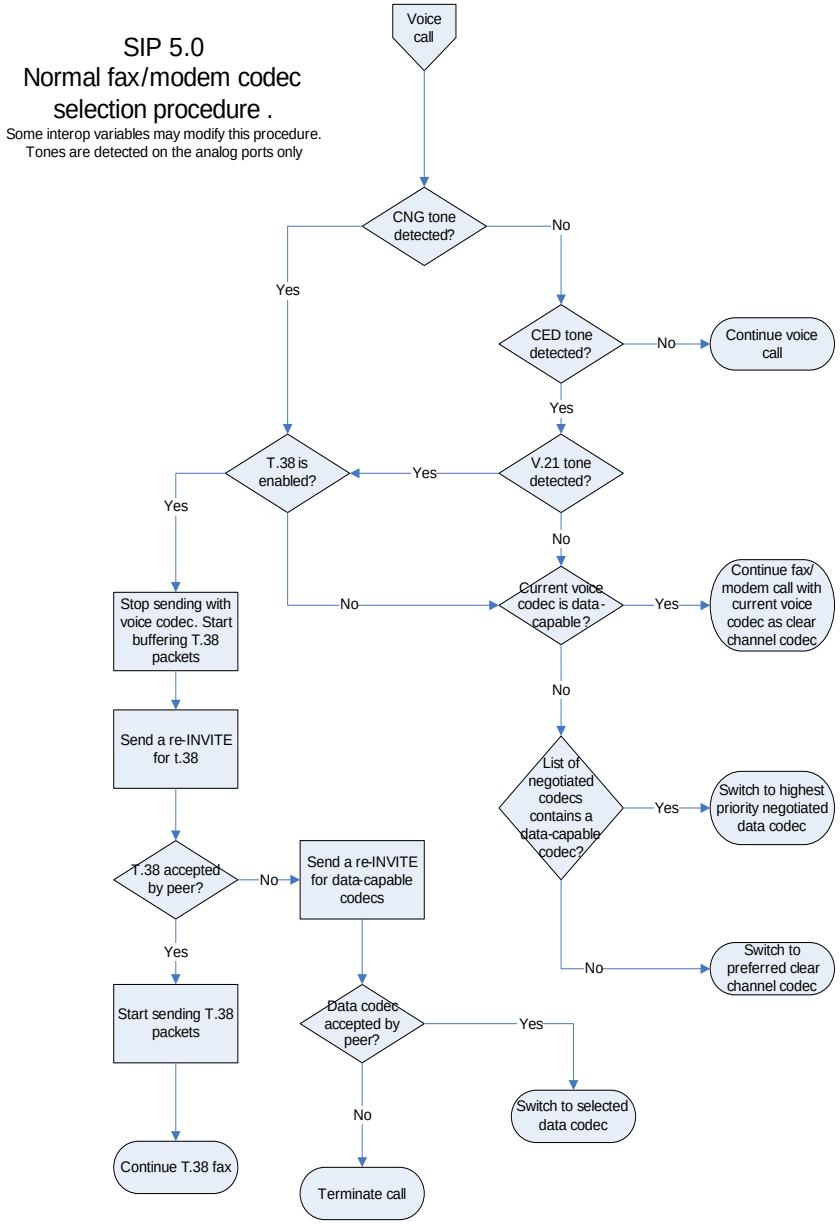
Moreover, when no data-capable codec is part of the list of negotiated codecs, this variable indicates which codec to use when fax or modem tones are detected. However, if the negotiated voice codec is data-capable, the voice codec will be used for data instead of the preferred data codec. See "Data Codec Selection Procedure" on page 283 for more details.

Media5 suggests to use *pcma* if you are located in Europe and *pcmu* anywhere else. However, you should check first which codec is supported in your telephone network.

## Data Codec Selection Procedure

The Mediatrix 4102 follows a procedure when selecting data codec. This procedure is the default behaviour of the Mediatrix 4102. Some interop variables may modify this procedure. Tones are detected on the analog ports only.

**Figure 96:** Data Codec Selection Procedure



# T.38 Fax

| **Standards Supported** | • Based on *draft-ietf-sipping-realtimefax-01.txt*<br>• Recommendation ITU T.38 version 0 |
|---|---|

T.38 fax relay is a real-time fax transmission; that is, two fax machines communicating with each other as if there were a direct phone line between the two. T.38 is called a fax relay, which means that instead of sending inband fax signals, which implies a loss of signal quality, it sends those fax signals out-of-band in a T.38 payload, so that the remote end can reproduce the signal locally.

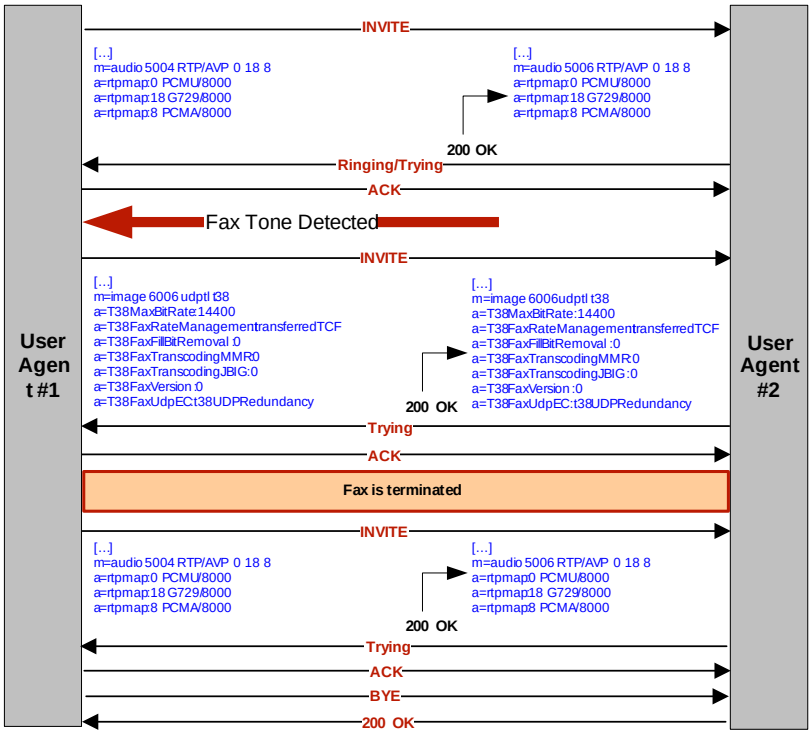
The Mediatrix 4102 can send faxes in T.38 mode over UDP or TCP. T.38 is used for fax if both units are T.38 capable; otherwise, transmission in clear channel over G.711 as defined is used (if *G.711 µ-law* and/or *G.711 A-law* are enabled). If no clear channel codecs are enabled and the other endpoint is not T.38 capable, the fax transmission fails.

**Caution:** The Mediatrix 4102 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4102 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message. See “Fax Issues” on page 385 for more details.

In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

The following is a T.38 fax call flow:

**Figure 97:** T.38 Fax Call Flow




### ▶ To set T.38 fax transmission:

1. Enable T.38 by setting the *dataIfCodecT38Enable* variable to **enable**.

2. Set the number of redundancy packets sent with the current packet in the *dataIfCodecT38ProtectionLevel* variable.

   This is the standard redundancy offered by T.38. Please see step ***3*** for additional reliability options for T.38.

   Available values range from 1 to 5, 3 being the default value.

3. For additional reliability, define the number of times T.38 packets are retransmitted in the *dataIfT38FinalFramesRedundancy* variable.

   This only applies to the T.38 packets where the PrimaryUDPTL contains the following T.38 data type:

   - HDLC_SIG_END,
   - HDLC_FCS_OK_SIG_END,
   - HDLC_FCS_BAD_SIG_END and
   - T4_NON_ECM_SIG_END

4. Restart the Mediatrix 4102 so that the changes may take effect.

## T.38 No-Signal

You can set the Mediatrix 4102 to send no-signal packets during a T.38 fax transmission. The Mediatrix 4102 sends no-signal packets if no meaningful data have been sent for a user-specified period of time.

▶ **To send T.38 no-signal:**

1. Set the period, in seconds, at which no-signal packets are sent during a T.38 transmission in the *dataIfT38NoSignalTimeout* variable.

   No-signal packets are sent out if there are no valid data to send.

2. Enable the sending of T.38 no-signal packets by setting the *dataIfT38NoSignalEnable* variable to **enable**.

### Behaviour when Switching from Voice to T.38

You can define the behaviour when the unit starts sending T.38 no-signal packets when being Re-Invited to switch from voice to T.38.

▶ **To define the behaviour:**

1. In the *sipInteropMIB*, set the *sipInteropT38NoSignalBehavior* variable to the proper value.

**Figure 98:** Behaviour when Switching from Voice to T.38 Parameters

| Parameter | Description |
|---|---|
| receivingReInvite | The unit starts sending T.38 no-signal packets when receiving the Re-Invite. |
| receivingAck | The unit waits until the ACK to its 200 OK has been received before sending T.38 no-signal packets. |

This configuration has no effect if the *dataIfT38NoSignalEnable* variable is set to **disable** (see "T.38 No-Signal" on page 285 for more details).

## T.38 INVITE Rejected with 606

You can define the behavior of the Mediatrix 4102 when receiving a 606 SIP error response to an INVITE for T.38 fax.

### ▶ To define the behavior

1. In the *sipInteropMIB*, set the *sipInteropBehaviorOnT38InviteRejectedWith606* variable to the proper value.

**Figure 99:** Behaviour when T.38 INVITE is Rejected with 606 Parameters

| Parameter | Description |
|---|---|
| dropCall | The call is dropped by sending a BYE. |
| usePreviousMediaNegotiation | No re-INVITE is sent and the audio codec from the last successful negotiation is used. For the remainder of the call, T.38 is disabled and fax detection may trigger a switch to a clear channel codec that was available in the last successful negotiation. |

CHAPTER 19

# SIP Protocol Features

This chapter defines how to configure the SIP-specific features to properly use the SIP signalling programs and information defined in Media5' SIP stack.

## User Agents

A user agent is a logical entity that can act as both client and server for the duration of a dialog. Each line (also known as endpoint) of the Mediatrix 4102 is a user agent.

You can set information for each user agent such as its telephone number and friendly name. This information is used to dynamically create the *To*, *From* and *Contact* headers used in the request the user agent sends. These headers make up the caller ID information that is displayed on telephones/faxes equipped with a proper LCD display. See "Caller ID Information" on page 189 for more details.

Most of the variables related to the user agents are located in tables. You can display and define the information for all lines. You can also use these tables to create/edit five user names and passwords per line. This means that:

- Rows 1-5 of the table are reserved for line 1.
- Rows 6-10 of the table are reserved for line 2.

If you want to enter a user name for the second line, you must do so in the sixth row of the table.

Before changing a parameter value, build its corresponding table with your MIB browser's table functionality. Depending on the MIB browser you are using, the tables may not appear the same way.

In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

You can also set these parameters via the web interface, as described in "SIP User Agent" on page 79.

**To set basic user agent information:**

1. In the *sipMIB*, set the user agent port number in the *sipPort* variable.

   The default value is 0. If *sipPort* is set to 0, the default SIP port is used.

2. Define whether or not to override the default proxy home domain used by entering a domain in the *sipDomain* variable.

   This value replaces the home domain proxy host as defined in "Proxy Server" on page 179. It is used by the address of record in the *To* and *From* headers.

3. In the *sipUAIfTable* group, set a main user name in the *sipUAMainUsername* variable.

   The main user name uniquely identifies this endpoint in the domain, such as a telephone number. It is used to create the *Contact* and *From* headers. The *From* header carries the permanent location (IP address, home domain) where the endpoint is located. The *Contact* header carries the current location (IP address) where the endpoint can be reached. Contact headers are used in two ways:

   - First, contacts are registered to the registrar. This enables callers to be redirected to the endpoint's current location.
   - Second, a contact header is sent along with any request the user agent sends (e.g., INVITE), and is used by the target user agent as a return address for later requests to this endpoint.

   You cannot set this field to an empty value. Furthermore, it is reset to 333000X during a factory reset, the X digit being the port number.

4. Set a display name in the *sipUADisplayName* variable.

   This is a friendly name for the user agent. It contains a descriptive version of the URI and is intended to be displayed to a user interface.

5. Define a list of other accepted user names in the *sipUAOtherAcceptedUsernames* variable.

   This is a list of user names that the endpoint recognizes as its own, but does not register in contacts sent to the registrar. The endpoint only registers the user name in *sipUAMainUsername*.

   You can use this variable to add variations on the main user name. For instance, let's say that the main user name is a telephone number, 555-1111. Variations could be to prefix the local area or country code, such as 819-555-1111.

   To include more than one user name, separate them with a "," character, such as: user1, user2, 5552222, 18195552222.

6. Restart the Mediatrix 4102 so that the changes may take effect.

## Home Domain Override

You can override the home domain configuration. The address of record in the REGISTER uses this string instead of the SIP domain as set in the *sipDomain* variable (see "User Agents" on page 287) or home domain proxy host (variable *sipHomeDomainProxyStaticHost* variable (see "Proxy Server" on page 179 for details).

▶ **To override the home domain configuration:**

1. In the *sipInteropMIB*, set the *sipInteropRegisterHomeDomainHostOverride* variable with the proper IP address or domain name.

## SIP User Agent Header

| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol, section 20.41 (User-Agent) |
|---|---|

The *User-Agent* header field contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

```
User-Agent: Softphone Beta1.5
```

You can specify whether or not the Mediatrix 4102 sends this information when establishing a communication and customize the header value.

▶ **To enable sending the SIP User Agent header:**

1. In the *sipInteropMIB*, set the *sipInteropSendUAHeaderEnable* variable to **enable**.
2. Set the *sipInteropUAHeaderConfig* variable with the proper macro.

   The following macros are replaced by their representation:

   - **%version%**: Version of the application.
   - **%mac%**: Unit MAC address (lowercase).
   - **%rev%**: Hardware revision number.
   - **%product%**: Product name.
   - **%%**: A '%' sign.

3. Restart the Mediatrix 4102 so that the changes may take effect.

# Session Timers

The session timer extension allows to detect the premature end of a call caused by a network problem or a peer's failure by resending an INVITE at every *n* seconds.

A successful response (200 OK) to this INVITE indicates that the peer is still alive and reachable. A timeout to this INVITE may mean that there are problems in the signalling path or that the peer is no longer available. In that case, the call is shut down by using normal SIP means.

▶ **To set Session Timer information:**

1. In the *sipUAIfTable* group, set the session timer maximum expiration delay in the *sipUAMaximumSessionExpirationDelay* variable.

   This is the suggested maximum time, in seconds, for the periodical session refreshes. It must be equal to or greater than the minimum value. This value is reflected in the *Session-Expires* header.

2. Set the session timer minimum expiration delay in the *sipUAMinimumSessionExpirationDelay* variable.

   This is the minimum value, in seconds, for the periodical session refreshes. It must be equal to or smaller than the maximum value. This value is reflected in the *Min-SE* header.

   The *Min-SE* value is a threshold under which proxies and user agents on the signalling path are not allowed to go.

▶ **To disable the Session Timer service:**

1. Set the *sipUAMaximumSessionExpirationDelay* variable to **0**.

   Increasing the maximum helps to reduce network traffic, but also makes "dead" calls longer to detect.

## Session Timer Version

| | |
|---|---|
| **Standards Supported** | • draft-ietf-sip-session-timer-08.txt<br>• draft-ietf-sip-session-timer-04.txt (expired) |

You can select the version of the session timer draft that the Mediatrix 4102 uses. Session timer versions other than those provisioned may not work because of backward compatibility issues between the versions.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

The Mediatrix 4102 supports the following session timer versions:

**Table 181:** Session Timer Versions Supported

| Version | Description |
|---|---|
| sessionTimer04 | The Mediatrix 4102 uses the session timer extension as described in the now expired *draft-ietf-sip-session-timer-04.txt*. Its use is deprecated. You should use this setting for backwards compatibility issues only. |
| sessionTimer08 | The Mediatrix 4102 uses the session timer extension as described in the more recent *draft-ietf-sip-session-timer-08.txt*. This draft version contains several enhancements over the previous ones, including the use of the *Min-SE* header. Use this setting if you do not need to interoperate with session timer v4-enabled parties. |

▶ **To set the version of session timers supported:**

1. In the *sipInteropMIB*, set the *sipInteropSessionTimersVersion* variable with the proper version.
   - sessionTimer04
   - sessionTimer08

## Background Information

The following explains how the session timers are used.

### SDP in Session Timer reINVITEs

The reINVITE is sent with the last SDP that was negotiated. Receiving a session timer reINVITE should not modify the connection characteristics.

### Relation Between Minimum and Maximum Values

A user agent that receives a *Session-Expires* header whose value is smaller than the minimum it is willing to accept replies a “422 Timer too low” to the INVITE and terminates the call. The phone does not ring.

It is up to the caller to decide what to do when it receives a 422 to its INVITE. Mediatrix units will automatically retry the INVITE, with a *Session-Expires* value equal to the minimum value that the user agent server was ready to accept (located in the *Min-SE* header). This means that the maximum value as set in the Mediatrix unit might not be followed. This has the advantageous effect of establishing the call even if the two endpoints have conflicting values. Mediatrix units will also keep retrying as long as they get 422 answers with different *Min-SE* values.

### Who Refreshes the Session?

Re-sending a session timer INVITE is referred to as refreshing the session. Normally, the user agent server that receives the INVITE has the last word on who refreshes. Mediatrix units always let the user agent client (caller) perform the refreshes if the caller supports session timers. In the case where the caller does not support session timers, the Mediatrix unit assumes the role of the refresher.

# Authentication

| | |
|---|---|
| **Standards Supported** | Basic and Digest authentication as per RFC 3261 |

Authentication information allows you to add some level of security to the Mediatrix 4102 lines by setting user names and passwords. You can add two types of authentication information:

- line-specific authentication
- unit authentication

When a realm requests authentication, the line-specific authentication is tried first, and then the unit authentication if required.

You can also set these parameters via the web interface, as described in “SIP Authentication” on page 85.

## Line-Specific Authentication

You can define up to five user names and five passwords for each line of the Mediatrix 4102. A line can thus register with five different realms. Keep in mind that:

- Rows 1-5 of the table where you define the user names and passwords are reserved for line 1.
- Rows 6-10 of the table where you define the user names and passwords are reserved for line 2.
- etc.

For instance, to enter a user name for the second line, you must do so in the sixth row of the table.

▶ **To set line-specific authentication:**

1. In the *sipUAIfAuthenticationTable* group, set the following information:

**Table 182:** Line-Specific Authentication

| Variable | Description |
|---|---|
| sipUAAuthRealm | When authentication informations are required from users, the realm identifies who requested the information. |
| sipUAAuthUsername | A string that uniquely identifies this endpoint in the realm, used for authentication purposes. The user name always maps to a password. |
| sipUAAuthPassword | User password. |
| sipUAAuthValidateRealm | When set to disable, the current user credentials are valid for any realm. When set to enable, the user credentials are used only for a specific realm set in the sipUAAuthRealm variable. |

## Unit Authentication

You can define up to five user names and five passwords for the Mediatrix 4102. These user names and passwords apply to all lines of the unit.

In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Authentication*.

▶ **To set unit authentication:**

1. In the *sipUnitAuthenticationTable* group, set the following information:

**Table 183:** Unit-Specific Authentication

| Variable | Description |
|---|---|
| sipUnitAuthRealm | When authentication informations are required from users, the realm identifies who requested the information. |
| sipUnitAuthUsername | A string that uniquely identifies this endpoint in the realm, used for authentication purposes. The user name always maps to a password. |
| sipUnitAuthPassword | User password. |
| sipUnitAuthValidateRealm | When set to disable, the current unit credentials are valid for any realm. When set to enable, the unit credentials are used only for a specific realm set in the *sipUnitAuthRealm* variable. |

## Authentication Request Protection

When the Mediatrix 4102 sends an authentication request, you can configure it so that it tries to apply the authentication with integrity protection when this feature is supported by the SIP server.

▶ **To set the quality of protection:**

1. In the *sipInteropMIB*, specify the quality of protection the SIP User Agent should apply to its authentication request in the *sipInteropAuthenticationQop* variable.

   The following values are supported:

**Table 184:** Quality of Protection

| Parameter | Description |
|---|---|
| auth | The SIP User Agent applies authentication only. This is the default value. |

**Table 184:** Quality of Protection (Continued)

| Parameter | Description |
|---|---|
| auth-int | The SIP User Agent applies authentication with integrity protection (see RFC 2617). |

2. Restart the Mediatrix 4102 so that the changes may take effect.

# SIP Trusted Sources

You can configure the Mediatrix 4102 so that it only accepts SIP messages coming from one of six trusted IP addresses. All other SIP messages are ignored. This source address validation takes place at the IP level and does not depend on the SIP header or body.

▶ **To configure the SIP trusted sources feature:**

1. In the *sipMIB*, set the *sipTrustedSourcesIpAddress* variable with the IP address of a trusted source of SIP messages.

   You can enter up to six addresses.

2. Enable the SIP trusted sources feature by setting the *sipTrustedSourcesEnable* to **enable**.

   If you set the variable to **disable**, the Mediatrix 4102 does not validate the source address of SIP messages at the IP level.

# NAT Traversal

The Mediatrix 4102 may be used in a private domain that is not directly connected to the IP network. For instance, this may be the case for ITSP (Internet Telephony Service Provider) clients that have a small private network. This private network is connected to the public IP network through the NAT (Name Address Translation) technology.

Currently only one Mediatrix unit can be deployed behind a standard NAT.

You can configure the Mediatrix 4102 with the public IP address of the NAT system, which allows to reach the unit. SIP packets sent by the Mediatrix 4102 contain the NAT address configured as SIP contact. If the NAT service is not activated, the real IP address of the Mediatrix 4102 is used.

This method is recommended when the public IP address of the NAT system is static or does not change regularly since it would cause downtime until it is changed manually.

## Network Address Translation

Network Address Translation (NAT) is a method of connecting multiple computers to the Internet (or any other IP network) by using one IP address. This allows home users and small businesses to cheaply and efficiently connect their network to the Internet. The basic purpose of NAT is to multiplex traffic from the internal network and present it to the Internet as if it was coming from a single computer having only one IP address.

NAT automatically provides firewall-style protection without any special set-up because it only allows connections originating on the inside network. This means, for instance, that an internal client can connect to an outside FTP server, but an outside client cannot connect to an internal FTP server because it would have to originate the connection, and NAT does not allow that.

### Mediatrix 4102 Configuration

This section describes how to activate the NAT service of the Mediatrix 4102.

▶ **To activate the NAT service:**

1. In the *ipAddressConfig* folder, set the *localHostWanAddressSelectConfigSource* variable to **static**.

**Note:** If you want to do NAT traversal, you cannot use a PPPoE connection.

2. Enter the public IP address of the NAT system in the *localHostStaticWanAddress* variable.

   This is the public IP address used as Contact address by outgoing SIP packets crossing a NAT system.

### NAT System Configuration

You must configure the NAT system to some degree. The configuration required depends on the type of NAT system you are using, but this usually involves port forwarding configuration.

## SIP Transport Type

| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol |
|---|---|

You can globally set the transport type for all the lines of the Mediatrix 4102 to either UDP (User Datagram Protocol) or TCP (Transmission Control Protocol). The Mediatrix 4102 will include its supported transports in its registrations.

Please note that RFC 3261 states the implementations must be able to handle messages up to the maximum datagram packet size. For UDP, this size is 65,535 bytes, including IP and UDP headers. However, the maximum datagram packet size the Mediatrix 4102 supports for a SIP request or response is 5120 bytes excluding the IP and UDP headers. This should be enough, as a packet is rarely bigger than 2500 bytes.

You can also set these parameters via the web interface, as described in "SIP Transport Type" on page 83.

▶ **To set the transport type:**

1. In the *sipMIB*, set the priority order of the transport in the *sipTransportQValue* variable.

   A qvalue parameter is added to each contact. This only applies if the transport-specific registration is enabled.

   The qvalue gives each transport a weight, indicating the degree of preference for that transport. A higher value means higher preference.

   The format of the qvalue string must follow the RFC 3261 ABNF (a floating point value between 0.000 and 1.000). If you specify an empty string, no qvalue is set in the contacts.

   Because this variable is located in a table, you can have a different value for each line.

2. Enable the transport by setting the *sipTransportEnable* variable to **enable**.

   The UDP and TCP transport types are located in a table with two rows – one for each type. You can enable/disable a type for the unit.

   If the TCP transport type is not used, Media5 strongly suggests to disable it.

3. Enable the transport registration by setting the *sipTransportRegistrationEnable* variable to **enable**.

   The Mediatrix 4102 includes its supported transports in its registrations. It registers with one contact for each transport that is currently enabled. Each of these contacts contains a "transport" parameter.

This is especially useful for a system where there are no SRV records configured to use a predefined transport order for receiving requests. When sending a request, the unit either follows the SRV configuration, or, if not available, any transport parameter received from a redirection or from a configured SIP URL. See "Chapter 11 - DNS SRV Configuration" on page 185 for more details.

**Note:** If the Mediatrix 4102 has the following configuration:

- the *sipTransportRegistrationEnable* variable is set to **disable**
- the UDP transport type is disabled
- the TCP transport type is enabled

The Mediatrix 4102 will not work properly unless the SIP server uses the TCP transport type by default.

This is also true if the Mediatrix 4102 has the TCP transport disabled and the UDP transport enabled. In this case, the Mediatrix 4102 will not work properly unless the SIP server uses the UDP transport protocol by default.

4. Restart the Mediatrix 4102 so that the changes may take effect.

## Transport Parameter

You can define whether the Mediatrix 4102 must include its supported transport in all SIP messages that have the *Contact* header, except for the REGISTER message. See "SIP Transport Type" on page 293 for details on how to include transport parameters in the REGISTER message.

If enabled, then the Mediatrix 4102 will send SIP messages with the "transport" parameter in the *Contact* header set to the currently supported transport type.

▶ **To include the supported transport in the contact header:**

1. In the *sipMIB*, indicate whether or not the unit must include its supported transport in the *Contact* header in the *sipTransportContactEnable* variable.

   Available values are *enable* and *disable*. If you set the variable to enable, the transport parameter is either set to:

   - *transport=tcp* when TCP is enabled and UDP is disabled
   - *transport=udp* when UDP is enabled and TCP disabled
   - no transport parameter when both TCP and UDP are enabled

## UDP Source Port Behaviour

You can configure if the Mediatrix 4102 always uses the same local port (the port on which it is listening for incoming packets) when sending SIP traffic over UDP. This is called symmetric UDP source port. Symmetric UDP ports are sometimes needed to traverse NAT/Firewall devices.

When changing this setting, all destinations are automatically sent out of the penalty box, when applicable.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To set the UDP source port behaviour:**

1. In the *sipInteropMIB*, set the *sipInteropSymmetricUdpSourcePortEnable* variable to **enable**.

   The SIP signalling sent over UDP originates from the same port as the port on which the user agent is listening (see "User Agents" on page 287 for details). ICMP messages are not processed, which means that unreachable targets will take longer to detect.

   If you set the variable to **disable**, the SIP signalling over UDP uses a randomly-generated originating port. ICMP errors are processed correctly.

2. Restart the Mediatrix 4102 so that the changes may take effect.

# SIP Penalty Box

The penalty box feature is used to “quarantine” a given host which address times out. During that time, the address is considered as “non-responding” for all requests.

This feature is most useful when using multiple servers and some of them are down. It ensures that users wait a minimal period of time before trying a secondary host.

You can also set these parameters via the web interface, as described in “SIP Penalty Box” on page 81.

## Penalty Box vs Transport Types

Media5 recommends to use this feature with care when supporting multiple transports (see “SIP Transport Type” on page 293 for more details) or you may experience unwanted behaviours.

When the Mediatrix 4102 must send a packet, it retrieves the destination from the packet. If the destination address does not specify a transport to use and does not have a DNS SRV entry that configures which transport to use, then the Mediatrix 4102 tries all transports it supports, starting with UDP. If this fails, it tries with TCP. The unit begins with UDP because all SIP implementations must support this transport, while the mandatory support of TCP was only introduced in RFC 3261.

**Note:** It is not the destination itself that is placed in the penalty box, but the combination of address, port and transport. When a host is in the penalty box, it is never used to try to connect to a remote host unless it is the last choice for the Mediatrix 4102 and there are no more options to try after this host.

Let’s say for instance that the Mediatrix 4102 supports both the UDP and TCP transports. It tries to reach endpoint “B” for which the destination address does not specify a transport and there is no DNS SRV entry to specify which transports to use in which order. It turns out that this endpoint “B” is also down. In this case, the Mediatrix 4102 first tries to contact endpoint “B” via UDP. After a timeout period, UDP is placed in the penalty box and the unit then tries to contact endpoint “B” via TCP. This fails as well and TCP is also placed in the penalty box.

Now, let’s assume endpoint “B” comes back to life and the Mediatrix 4102 tries again to contact it before UDP and TCP are released from the penalty box. First, the unit tries UDP, but it is currently in the penalty box and there is another transport left to try. The Mediatrix 4102 skips over UDP and tries the next target, which is TCP. Again, TCP is still in the penalty box, but this time, it is the last target the Mediatrix 4102 can try, so penalty box or not, TCP is used all the same to try to contact endpoint “B”.

There is a problem if endpoint “B” only supports UDP (RFC 2543-based implementation). Endpoint “B” is up, but the Mediatrix 4102 still cannot contact it: with UDP and TCP in the penalty box, the unit only tries to contact endpoint “B” via its last choice, which is TCP.

The same scenario would not have any problem if the penalty box feature was disabled. Another option is to disable TCP in the Mediatrix 4102, which makes UDP the only possible choice for the unit and forces to use UDP even if it is in the penalty box.

You must fully understand the above problem before configuring this feature. Mixing endpoints that do not support the same set of transports with this feature enabled can lead to the above problems, so it is suggested to either properly configure SRV records for the hosts that can be reached or be sure that all hosts on the network support the same transport set before enabling this feature.

## Penalty Box Configuration

The following steps describe how to configure the penalty box feature.

▶ **To set the penalty box feature:**

1. In the *sipMIB*, locate the *sipPenaltyBox* group.
2. Set the amount of time, in seconds, that a host spends in the penalty box in the *sipPenaltyBoxTime* variable.

   Changing the value does not affect IP addresses that are already in the penalty box. The *sipPenaltyBoxTime* only affects new entries in the penalty box.

3. Enable the SIP penalty box feature by setting the *sipPenaltyBoxEnable* variable to **enable**.

   The penalty box is always “active”. This means that even if the feature is disabled, IP addresses are marked as invalid, but they are still tried. This has the advantage that when the feature is enabled, IP addresses that were already marked as invalid are instantly put into the penalty box.

# Registration Parameters

The following describes registration parameters and behaviours you can configure.

## Refreshing Registration

You can refresh the registration, i.e., commit the changes you have done to the registration. When refreshing the registration, all enabled endpoints unregister themselves from the previous registrar and send a new registration to the current registrar with the current parameters.

Variables whose modification require a registration refresh are:

- sipRegistrarStaticHost
- sipRegistrarStaticPort
- sipUAMainUsername
- sipUADisplayName
- sipServerSelectConfigSource
- sipTransportRegistrationEnable
- sipTransportEnable (if sipTransportRegistrationEnable is enabled)
- sipTransportQValue (if sipTransportRegistrationEnable is enabled)

**To refresh the registrations:**

1. In the *sipMIB*, set the *sipRegistrationCmdRefresh* variable with the proper behaviour.

   The following values are available:

   - noOp: No operation.
   - refresh: Refresh registrations.

2. Define the time, in seconds, at which a registered unit begins updating its registration before the registration expiration in the *sipReRegistrationTime* variable.

   For instance, if the variable is set to 43 and the registration lasts one hour, the unit will send new REGISTER requests 59 minutes and 17 seconds after receiving the registration acknowledgement (43 seconds before the unit becomes unregistered).

**Note:** Normally, the Mediatrix 4102 cannot make or receive calls until the REGISTER has completed successfully. Because the timeout for a SIP transaction in UDP is 32 seconds, it is possible to have an ongoing re-REGISTER transaction at the same moment that the registration itself expires. This could happen if *sipReRegistrationTime* is set to a value lower than 32.

In that case, the user agent becomes unregistered, and will become registered again only when the re-REGISTER request is answered with a positive response from the server. See “Unregistered Line Behaviour” on page 256 for a workaround if the unit cannot make calls during that period.

This value MUST be lower than the value of the Expires header of the contact in the 200 OK response to the REGISTER, otherwise the unit rapidly sends REGISTER requests continuously.

## Registration Expiration

The SIP protocol allows an entity to specify the “expires” parameter of a contact in a REGISTER request. The server can return this “expires” parameter in the 200 OK response or select another “expires”. In the REGISTER request, the “expires” is a suggestion the entity makes.

The “expires” parameter indicates how long, in seconds, the user agent would like the binding to be valid.

You can configure the “expires” parameter the Mediatrix 4102 sends.

▶ **To configure the registration expiration:**

1. In the *sipMIB*, set the *sipRegistrationProposedExpirationValue* variable with the suggested expiration delay, in seconds, of a contact in the REGISTER request.

   Available values are from 1 s to 86,400 s (one day).

   This value does not modify the time before a re-REGISTER.

   - The time is the “expires” of the contact in the 200 OK response to the REGISTER request minus the value set in the *sipReRegistrationTime* variable.
   - If the “expires” of the contact in the 200 OK response to the REGISTER is not present or not properly formatted, then the time is the default registration expires minus the value set in the *sipReRegistrationTime* variable.

   See “Refreshing Registration” on page 296 for more details.

   Setting the variable to **0** disables the expiration suggestion.

### Default Registration Expiration

| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol, section 20.41 (User-Agent) |
|---|---|

RFC 3261 specifies that, when the “expires” parameter or the “Expires” header is missing or not properly formatted for a contact of the 200 OK in response to a REGISTER request, the unit must use a default registration expiration value of 3600 s.

You can configure the value of the default registration expiration.

▶ **To configure the default registration expiration:**

1. In the *sipInteropMIB*, set the *sipInteropDefaultRegistrationExpiration* variable with the default registration expiration.

   The delay before a re-REGISTER is the value set in the *sipInteropDefaultRegistrationExpiration* variable minus the value set in the *sipReRegistrationTime* variable. See “Refreshing Registration” on page 296 for more details.

   The recommended value in RFC 3261 (section 10.2) is 3600 seconds.

2. Restart the Mediatrix 4102 so that the change may take effect.

# Publication Parameters

| **Standards Supported** | • RFC 3863 – Presence Information Data Format (PIDF)<br>• RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication |
|---|---|

The following describes publication parameters and behaviours you can configure.

### Refreshing Publications

You can refresh the publications, i.e., commit the changes you have done to the publications. When refreshing the publications, all enabled endpoints unpublish themselves from the previous Presence Compositor and send a new publication to the current Presence Compositor with the current parameters.

Variables whose modification require a registration refresh are:

- sipPresenceCompositorStaticHost
- sipPresenceCompositorStaticPort

▶ **To refresh the publications:**

1. In the *sipMIB*, set the *sipPublicationCmdRefresh* variable with the proper behaviour.

   The following values are available:

   - noOp: No operation.
   - refresh: Refresh publications.

2. Define the time, in seconds, at which the User Agent begins refreshing its publication before the publication expiration in the *sipPublicationRefreshTime* variable.

   A publication is valid for a period of time specified by the Presence Compositor. The User Agent is then responsible for refreshing its previously established publications before their expiration interval has elapsed.

   For instance, if the publication lasts 60 minutes and this variable is set to 43, the unit sends new PUBLISH requests 59 minutes and 17 seconds after the reception of the publication acknowledgement (43 seconds before the end of the publication period).

   Setting this variable to **0** means that the User Agent falls into the 'unpublished' state BEFORE sending the refreshing PUBLISH request.

## Publications Expiration

The SIP protocol allows an entity to specify the “expires” parameter of a contact in a PUBLISH request. The server can return this “expires” parameter in the 200 OK response or select another “expires”. In the PUBLISH request, the “expires” is a suggestion the entity makes.

The “expires” parameter indicates how long, in seconds, the user agent would like the binding to be valid.

You can configure the “expires” parameter the Mediatrix 4102 sends.

▶ **To configure the publications expiration:**

1. In the *sipMIB*, set the *sipPublicationProposedExpirationValue* variable with the suggested expiration delay, in seconds, of a publication in the PUBLISH request.

   Keep in mind that this is only a suggestion and that servers will decide the publication time following local policy.

   Available values are from 1 s to 86,400 s (one day).

   This value does not modify the delay before a re-PUBLISH.

   - The delay is the value of the Expires header in the 200 OK response to the PUBLISH minus the value set in the *sipPublicationRefreshTime* variable.
   - If the Expires header in the 200 OK response to the PUBLISH is not present or badly formatted, then the delay is the value of the variable *sipInteropDefaultPublicationExpiration* minus the value set in the *sipPublicationRefreshTime* variable.

   See “Refreshing Publications” on page 297 for more details.

   Setting the variable to **0** disables the expiration suggestion.

## Default Publication Expiration

You can configure the value of the default registration expiration.

▶ **To configure the default publication expiration:**

1. In the *sipInteropMIB*, set the *sipInteropDefaultPublicationExpiration* variable with the default publication expiration.

   The delay before a re-PUBLISH is the value set in the *sipInteropDefaultPublicationExpiration* variable minus the value set in the *sipPublicationRefreshTime* variable. See “Refreshing Publications” on page 297 for more details.

2. Restart the Mediatrix 4102 so that the change may take effect.

# Interop Parameters

The interop parameters allow the Mediatrix 4102 to properly work, communicate, or connect with specific IP devices.

## Call Transfer Capacity

The following parameters allow you to define how the Mediatrix 4102 handles call transfers.

### Call Transfer Version

| | |
|---|---|
| **Standards Supported** | • draft-ietf-sip-cc-transfer-05.txt<br>• draft-ietf-sip-cc-transfer-02.txt (expired)<br>• draft-ietf-sip-refer-02.txt<br>• draft-ietf-sipping-cc-transfer-01.txt<br>• RFC 3515 - The Session Initiation Protocol (SIP) (Refer Method) |

You can select the version of the transfer draft that the Mediatrix 4102 uses. The provisioned version is used for initiating transfers and receiving them. Transfer versions other than those provisioned do not work.

**Table 185:** Call Transfer Versions Supported

| Version | Description |
|---|---|
| transfer02 | The Mediatrix 4102 executes transfers by using the methods described in the now expired *draft-ietf-sip-cc-transfer-02.txt*. Its use is deprecated and you should use this setting for backward compatibility issues only. |
| transfer05UsingRefer02 | The Mediatrix 4102 executes transfers by using the methods described in the more recent *draft-ietf-sip-cc-transfer-05.txt*. This draft version contains several enhancements over the previous ones. Among others, it is possible to use the *Replaces* header to provide a more seamless attended transfer to the user. This method also uses *draft-ietf-sip-refer-02.txt*. Use this setting if you do not need to interop with transfer02-enabled parties. See "Replaces Configuration Setting" on page 300 for more details. |
| sippingTransfer01UsingReferRfc3515 | The Mediatrix 4102 executes transfers by using the methods described in *draft-ietf-sipping-cc-transfer-01.txt*. This draft version is more recent than Transfer02 and Transfer05UsingRefer02. This method also uses the *RFC 3515 - The Session Initiation Protocol (SIP)* Refer Method. |

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To set the version of transfer supported:**

1. In the *sipInteropMIB*, set the *sipInteropTransferVersion* variable with the proper version.
   - transfer02
   - transfer05UsingRefer02
   - sippingTransfer01UsingReferRfc3515

## Replaces Configuration Setting

You can configure how to use the *Replaces* header mechanism used in a transfer. When supported by the target of the transfer, the *Replaces* header mechanism ensures a more seamless transfer by permitting the initiating party to effectively replace a current call by another instead of disconnecting the call to be replaced and creating a second call. This allows you to control how the Mediatrix 4102 interoperates with other vendor's products and older Mediatrix units.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

**To set Replaces configuration:**

1. In the *sipInteropMIB*, set the Replaces configuration in the *sipInteropReplacesConfig* variable.
   You have the following choices:

**Table 186:** Replaces Configuration

| Configuration | Description |
|---|---|
| doNotUseReplaces | The *Replaces* header is not used. |
| useReplacesWithRequire | The *Replaces* header is used. It can be seen in the *Refer-To* header of the REFER request sent by the transferor. It can also be seen in the INVITE sent by the transferee. The target that supports *Replaces* uses its information to merge the new INVITE with an existing call specified in the *Replaces* header.<br>The transferee requires to use the replaces extension for proper completion of the transfer. If the target of the transfer does not support the replaces extension, the Mediatrix 4102 retries the transfer using replaces by reversing the roles of the target and the transferee (by resending the REFER to the initial target instead of the initial transferee). As a last resort (if none of the participants supports replaces), the transfer is carried out without using the replaces extension. |
| useReplacesNoRequire | The *Replaces* header is used. It can be seen in the *Refer-To* header of the REFER request sent by the transferor. It can also be seen in the INVITE sent by the transferee. The target that supports Replaces uses its information to merge the new INVITE with an existing call specified in the *Replaces* header.<br>This disables the transfer fallback. The replaces information is still present, but no check is made that it is effectively used to complete the transfer. |

## Replaces Version

| | |
|---|---|
| **Standards Supported** | • sip-replaces-01 draft<br>• sip-replaces-03 draft |

You can select the version of the *ietf-sip-replaces* draft to which the Mediatrix 4102 must conform. The provisioned version affects the way blind transfers are executed.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

**Table 187:** Replaces Versions Supported

| Version | Description |
|---|---|
| replaces01 | The Transferor can use a REFER with Replaces when proceeding to initiate a blind transfer. This results in the Transferee including a Replaces header in its INVITE to the Transfer Target. |
| replaces03 | When initiating a blind transfer, the Transferor first CANCELs its call with the Target and then issues a REFER without Replaces to the Transferee.<br>**Note**: A side effect is that the phone will stop ringing and start again. |

▶ **To set the version of Replaces supported:**

1. In the *sipInteropMIB*, set the *sipInteropReplacesVersion* variable with the proper version.
   - replaces01
   - replaces03

## Transmission Timeout

| | |
|---|---|
| **Standards Supported** | • RFC 2543bis – SIP: Session Initiation Protocol<br>• RFC 3261 – SIP: Session Initiation Protocol |

If a DNS SRV answer contains more than one entry, the Mediatrix 4102 will try these entries if the entry initially selected does not work. You can configure the maximum time, in seconds, to spend waiting for answers to messages, from a single source. Retransmissions still follow the algorithm proposed in *RFC 2543bis*, but the total wait time can be overridden by using this feature.

For example, if you are using DNS SRV and more than one entry is present, this timeout is the time it takes before trying the second entry.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To set the transmission timeout:**

1. In the *sipInteropMIB*, locate the *sipInteropTransmissionTimeout* variable.
2. Set the timeout value.

   Available values are from 1 to 32 seconds.

## Max-Forwards Header

| | |
|---|---|
| **Standards Supported** | RFC 3261 – SIP: Session Initiation Protocol |

You can configure whether the Mediatrix 4102 inserts the *Max-Forwards* header into sent requests, as per RFC 3261. Max-Forwards serves to limit the number of hops a request can make on the way to its destination. It consists of an integer that is decremented by one at each hop. If the Max-Forwards value reaches 0 before the request reaches its destination, it will be rejected with a "483 (Too Many Hops)" error response.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

▶ **To insert the Max-Forwards header into SIP requests:**

1. In the *sipInteropMIB*, set the *sipInteropMaxForwardsValue* variable to the value you want.

   Any positive value means that the *Max-Forwards* header is inserted into sent requests. The default value is **70**.
2. Restart the Mediatrix 4102 so that the change may take effect.

▶ **To disable inclusion of this header in SIP requests:**

1. In the *sipInteropMIB*, set the *sipInteropMaxForwardsValue* variable to **-1**.
2. Restart the Mediatrix 4102 so that the change may take effect.

## Referred-By Field

The SIP REFER method allows the referrer to provide information about the reference to the refer target by using the referree as an intermediary. The mechanism for carrying the referrer's identity, expressed as a SIP URI, is the *Referred-By* header.

You can configure the *Referred-By* field used in a SIP REFER request to decide whether it contains the permanent URL provided by the SIP stack or the address of record used when the unit registered.

▶ **To configure the *Referred-By* field:**

1. In the *sipInteropMIB*, set the *sipInteropReferredByConfig* variable to the value you want.

**Table 188:** Referred-By Field Parameters

| Parameter | Description |
|---|---|
| useSipStackDefault | The SIP stack populates the *Referred-By* header field. |
| useLocalUrl | Uses the local URL to populate the *Referred-By* header field. |

## Direction Attributes in a Media Stream

The Mediatrix 4102 allows you to define various direction attributes pertaining to the media stream.

### Direction Attribute

| | |
|---|---|
| **Standards Supported** | • RFC 2543 – SIP: Session Initiation Protocol<br>• RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP) |

You can define:

- if the SDP direction attribute is present in the initial INVITE sent by the Mediatrix 4102
- whether or not the direction attribute present in the SDP received from the peer is ignored

▶ **To define if the direction attribute is present:**

1. In the *sipInteropMIB*, set the *sipInteropSdpDirectionAttributeEnable* variable to the proper value.

**Table 189:** SDP Direction Attribute

| Parameter | Description |
|---|---|
| disable | No direction attribute is present in the SDP sent by the Mediatrix 4102.<br>The Mediatrix 4102 ignores any direction attribute found in the SDP received from the peer.<br>To put an endpoint on hold, a SDP containing a connection address of “0.0.0.0” is sent.<br>The method to put a session on hold is in conformance with RFC 2543. |

**Table 189:** SDP Direction Attribute (Continued)

| Parameter | Description |
|---|---|
| enable | The Mediatrix 4102 always sends the direction attribute in the SDP of the initial INVITE.<br>The initial handshake determines whether or not the peer supports the direction attribute.<br>• If the direction attribute is present in the SDP received from the peer, the Mediatrix 4102 sends the direction attribute in the SDP for the remainder of the session.<br>• If the direction attribute is not present in the SDP received from the peer, the Mediatrix 4102 does not send the direction attribute in the SDP for the remainder of the session.<br>If present in the SDP, the direction attribute is preferred over the connection address to transmit session modification information.<br>This method is in conformance with RFC 3264. |

## When Putting a Call on Hold

| | |
|---|---|
| **Standards Supported** | RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP) |

The Mediatrix 4102 can provide the direction attribute and the meaning of the connection address “0.0.0.0” sent in the SDP when putting an endpoint on hold.

This configuration has no effect if the *sipInteropSdpDirectionAttributeEnable* variable is set to **Disable** (see “Direction Attribute” on page 302 for more details).

See “Call Hold” on page 334 for more details on holding calls.

▶ **To define the direction attribute when putting a call on hold:**

1. In the *sipInteropMIB*, set the *sipInteropOnHoldSdpStreamDirection* variable to the proper value.

**Table 190:** Direction Attributes

| Parameter | Description |
|---|---|
| inactive | The stream is put on hold by marking it as *inactive*. This is the default value. This setting should be used for backward compatibility issues. |
| sendonly | The stream is put on hold by marking it as *sendonly*.<br>This method allows the Mediatrix 4102 to be in conformance with RFC 3264. |

## Answering a Hold Offer with the Direction Attribute “sendonly”

| | |
|---|---|
| **Standards Supported** | RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP) |

You can define how to set the direction attribute and the connection address in the SDP when answering a hold offer with the direction attribute “sendonly”.

▶ **To define the behaviour with the "sendonly" direction attribute:**

1. In the *sipInteropMIB*, set the *sipInteropOnHoldAnswerSdpStreamDirection* variable to the proper value.

**Table 191:** "sendonly" Direction Attribute

| Parameter | Description |
|---|---|
| inactive | The stream is marked as inactive and the connection address is set to '0.0.0.0'. |
| revconly | If the stream is currently active or receive only, it is marked as recvonly and the connection address is set to the IP address of the unit.<br>If the stream is currently send only or inactive, it is marked as inactive and the connection address is set to '0.0.0.0'.<br>This method is in conformance with RFC 3264. |

In both cases, no direction attribute is present in the SDP if the *sipInteropSdpDirectionAttributeEnable* variable is set to **Disable** (see "Direction Attribute" on page 302 for more details).

## Allowing Multiple Active Media in Answer

You can define the behaviour of the Mediatrix 4102 when answering a request offering more than one active media.

▶ **To allow multiple active media in answer:**

1. In the *sipInteropMIB*, set the *sipInteropAllowMultipleActiveMediaInAnswer* variable to the proper value.

**Figure 100:** Allow Multiple Active Media in Answer

| Parameter | Description |
|---|---|
| disable | The answer contains only one active media. The media specified as active in the answer is the top-most matching one in the offer. Other media are set to inactive. |
| enable | Each matching active media in the offer is specified as active in the answer. Other media are set to inactive |

## Local Ring Behaviour on Provisional Response

You can set the Mediatrix 4102 so that it starts or not the local ring upon receiving a "18x Provisional" response without SDP.

This setting does not affect the behaviour when the "18x Provisional" response contains SDP, which allows to establish an early media session before the call is answered.

▶ **To define the local ring behaviour on provisional response:**

1. In the *sipInteropMIB*, set the *sipInteropLocalRingOnProvisionalResponse* variable to the proper value.

**Figure 101:** Local Ring Behaviour

| Parameter | Description |
|---|---|
| disable | The local ring is not started on a “18x Provisional” response without SDP, except for a “180 Ringing” message. This is the default value.<br>The 180 Ringing message is a provisional or informational response used to indicate that the INVITE message has been received by the user agent and that alerting is taking place.<br>**Note**: Using this default value means you are implementing a behaviour that is different from previous versions of the Mediatrix 4102 application. |
| enable | The local ring is started on any “18x Provisional” response without SDP. |

## SIP Credential

You can configure how the Mediatrix 4102 reuses the credential in different transactions of the same call or registration. For instance, it may be required that a new SIP request does not reuse the credential negotiated in the previous transaction of the same call or registration. For example, a re-INVITE will not reuse the credential of the INVITE but will be challenged.

▶ **To set the Mediatrix 4102 not to reuse the SIP credential:**

1. In the *sipInteropMIB*, set the *sipInteropReuseCredentialEnable* variable to **disable**.

   If you set this variable to **enable** (which is the default value), the Mediatrix 4102 reuses the credential negotiated in previous transactions.

## Branch Parameter Settings

The following are settings related to the Branch parameter.

### Branch Matching Method

| | |
|---|---|
| **Standards Supported** | • RFC 2543 – SIP: Session Initiation Protocol, section 10.1.2<br>• RFC 3261 – SIP: Session Initiation Protocol, section 8.1.1.7 |

You can configure the method used to match incoming SIP packets with a branch. A branch could be described as a link that allows to match a response to a request.

▶ **To configure the branch matching method:**

1. In the *sipInteropMIB*, set the *sipInteropBranchMatchingMethod* variable with the proper method to use.

**Table 192:** Branch Matching Method

| Method | Description |
|---|---|
| rfc2543 | Follows the method described in RFC 2543 (section 10.1.2). Responses are mapped to requests by the matching *To*, *From*, *Call-ID*, and *CSeq* headers and the branch parameter of the first *Via* header. |

**Table 192:** Branch Matching Method (Continued)

| Method | Description |
|---|---|
| rfc3261 | Follows the method described in RFC 3261 (section 8.1.1.7). A *Via* is inserted into the request and the *Via* header field value must contain a branch parameter. This parameter is used to identify the transaction created by that request. It is used by both the client and the server.<br>The branch ID is used to facilitate its use as a transaction ID. It must always begin with the characters “z9hG4bK”. If this is not the case, the Mediatrix 4102 uses the branching method as described in RFC 3261, section 17.2.3. |

### Transaction Matching Procedure

You can configure the use of the *Via* branch behaviour for incoming CANCEL requests. You can specify whether the SIP stack’s transaction matching procedure ignores the branch parameter of the *Via* header field in CANCEL requests with no *To* tag.

▶ **To configure the use of the *Via* branch behaviour for CANCEL requests:**

1. In the *sipInteropMIB*, set the *sipInteropIgnoreViaBranchIdInCancelEnable* variable with the proper behaviour.

**Table 193:** Via Branch Behaviour

| Method | Description |
|---|---|
| disable | The transaction matching procedure behaves according to section 17.2.3 of RFC 3261. This is the default value. |
| enable | The branch parameter is not used as a transaction matching criterion for CANCEL requests with no *To* tag. |

## Ringing Response Code

You can configure the response code sent back when the line starts ringing.

▶ **To configure the response code sent back:**

1. In the *sipInteropMIB*, set the *sipInteropRingingResponseCode* variable with the proper code to send back.

**Table 194:** Ringing Response Code

| Method | Description |
|---|---|
| send180Ringing | The Mediatrix 4102 sends out a “180 Ringing” response without a body. In this case, the ringback the caller hears is generated by his own unit upon receiving the message. This is the default value. |
| send183WithSdp | The Mediatrix 4102 returns a “183 Session Progress” packet with SDP (needed if the endpoint is required to generate ringback on connection). In this case, the RTP channel is opened earlier to allow the callee’s unit to generate the ringback and send it to the caller. |

## URI-Parameters

You can specify whether or not the Mediatrix 4102 copies the *uri-parameters* from the Request-URI header to the level of proxy authentication. Not copying the *uri-parameters* allows to reduce the SIP packet size but it does not follow the recommendations of RFC 3261.

▶ **To copy the uri-parameters to the level of proxy authentication:**

1. In the *sipInteropMIB*, set the *sipInteropProxyAuthenticationUriParametersEnable* variable to **enable**.

## Unsupported INFO Request

You can define the Mediatrix 4102's behaviour upon reception of an unknown type of INFO request.

▶ **To define the unsupported INFO request behaviour:**

1. In the *sipInteropMIB*, set the *sipInteropAckUnsupportedInfoRequests* variable with the proper behaviour.

**Table 195:** Unsupported INFO Request Behaviour

| Parameter | Description |
|---|---|
| disable | Unknown INFO requests trigger a 415 Unsupported Media Type response. |
| enable | Reception of unknown INFO requests is acknowledged with a 200 OK response. |

## Outbound Proxy Usage

You can define whether or not SIP requests sent to the proxy contain a *Route* header displaying the proxy's address. It is only effective when an outbound proxy host is configured (see "Outbound Proxy Server" on page 180 for more details). It is useful when interoperating with SIP servers that are not in conformance with RFC 3261's recommended practice concerning outbound proxy usage.

▶ **To set the outbound proxy usage:**

1. In the *sipInteropMIB*, set the *sipInteropRemoveOutboundProxyRouteHeader* variable with the proper behaviour.

**Table 196:** Outbound Proxy Usage

| Parameter | Description |
|---|---|
| disable | SIP requests sent to the outbound proxy contain a *Route* header, as per RFC 3261's recommendation. This is the default behaviour. |
| enable | SIP requests are routed to the outbound proxy without inserting a *Route* header in the SIP packet. |

## International Code Mapping

Some international calling parties have their caller ID prepended with the "+" character. You can instruct the Mediatrix 4102 to substitute this "+" character with a user-defined value.

▶ **To define an international code mapping:**

1. In the *sipInteropMIB*, set the *sipInteropInternationalCodeMappingString* variable with the character string that is substituted to the "+" character that prepends some international caller IDs.
2. Set the *sipInteropInternationalCodeMappingEnable* to **enable**.

   The default value is *disable*, which means that no substitution is performed and the '+' character is simply removed.

## T.38 Negotiation Syntax

| Standards Supported | • ITU-T Recommendation T.38, section D.2.3 |
|---|---|

You can define the format used, in the SDP portion of SIP packets, to advertise the unit's T.38 capabilities.

▶ **To set the T.38 negotiation syntax to use:**

1. In the *sipInteropMIB*, set the *sipInteropUseItuT38Format* variable with the proper behaviour.

**Table 197:** T.38 Negotiation Syntax Usage

| Parameter | Description |
|---|---|
| disable | Support for the boolean T.38 parameters T38FaxFillBitRemoval, T38FaxTranscodingMMR, and T38FaxTranscodingJBIG is advertised by associating a value of 0 (unsupported) or 1 (supported) with the parameter in the following manner:<br>`a=T38FaxFillBitRemoval:0`<br>`a=T38FaxTranscodingMMR:0`<br>`a=T38FaxTranscodingJBIG:0`<br>This is the default value. |
| enable | Support for the above T.38 parameters is advertised in conformance with ITU-T Recommendation T.38, section D.2.3. The presence of the parameter in the SDP indicates support for it (without the need for an associated value), while its absence means that it is not supported. |

## Addressing Failed Registration Attempts

You can control whether or not failed registration attempts are retried periodically.

▶ **To address failed registration attempts:**

1. In the *sipInteropMIB*, set the *sipInteropRetryFailedRegistration* variable with the proper behaviour.

**Table 198:** Failed Registration Attempts Behaviour

| Parameter | Description |
|---|---|
| disable | No retries are performed following a failed registration attempt. Manual intervention is required for the port to re-attempt registration. |
| enable | A failure to register a port to the SIP Registrar triggers an automatic retry every 2 minutes.<br>This is the default value. |

## SIP Domain in Request URI

You can control whether or not the request URI is built using the SIP Domain.

▶ **To use the SIP domain in the Request URI:**

1. In the *sipInteropMIB*, set the *sipInteropUseSipDomainInRequestURI* variable with the proper behaviour.

**Table 199:** SIP Domain in Request URI Behaviour

| Parameter | Description |
|---|---|
| disable | The request URI is built using the home domain proxy host as defined in "Proxy Server" on page 179. |

**Table 199:** SIP Domain in Request URI Behaviour (Continued)

| Parameter | Description |
|---|---|
| enable | The request URI is built using the SIP domain, if one is specified, for all SIP methods with the exception of REGISTER and PUBLISH. The parameter maddr is also added to the Request URI with the value of the home domain proxy host. See "User Agents" on page 287 for information on how to set the SIP domain. |

## SIP From: URI Content

You can specify the composition of the domain used in the from : URI of SIP requests sent by the unit.

▶ **To specify the composition of the domain used in the from : URI:**

1. In the *sipInteropMIB*, set the *sipInteropFromUriDomainSelection* variable with the proper behaviour.

**Table 200:** SIP From: URI Content

| Parameter | Description |
|---|---|
| sipDomain | The domain of from:URI is built from the *sipDomain* variable ("User Agents" on page 287). If *sipDomain* is empty, the *sipHomeDomainProxyHost* value is used instead ("Proxy Server" on page 179). This is the default value. |
| localHostWanAddress | The domain of from:URI is built from the unit address taken from the *localHostWanAddress* variable ("WAN Address Configuration Source" on page 159). |
| localHostFqdn | The domain of from:URI is built from the unit's FQDN taken from the *localHostFqdn* variable. If *localHostFqdn* is empty, the value of the *localHostWanAddress* variable is used instead. |

## Network Asserted Caller ID

The Mediatrix 4102 can extract the caller ID information from the *P-Asserted-Identity* header (a SIP extension described in RFC 3325) instead of the *From* header of the incoming SIP request

▶ **To use the P-Asserted-Identity header:**

1. In the *sipInteropMIB*, set the *sipInteropUsePAssertedHeader* variable with the proper behaviour.

**Table 201:** Network Asserted Caller ID Behaviour

| Parameter | Description |
|---|---|
| disable | Caller ID data is based on the value assigned to the *From* header of the incoming SIP request. This is the default value. |
| enable | The unit first attempts to get the caller ID information from the *P-Asserted-Identity* header, if present. In case of failure, it falls back to the content of the *From* header. |

## Payload Type Settings

The following are settings related to the DTMF payload type.

### Using the Payload Type Found in the Answer

The default behaviour when sending an initial offer that contains an RFC 2833 payload type is to keep using that payload type even if the response comes back with a different one. You can set the Mediatrix 4102 to rather use the payload type found in the answer.

▶ **To use the payload type found in the answer:**

1. In the *sipInteropMIB*, set the *sipInteropUseDtmfPayloadTypeFoundInAnswer* variable with the proper behaviour.

**Table 202:** Payload Type in Answer

| Parameter | Description |
|---|---|
| disable | Keep using the initial payload type. This is the default value. |
| enable | Use the RFC 2833 payload type found in the received answer |

This variable only has an effect when the *voiceIfDtmfTransport* variable is set to **outOfBandUsingRtp** (see "DTMF Transport Type" on page 268 for more details). The payload type is used symmetrically meaning that it is used to send and receive the DTMF. Use the variable *sipInteropAllowAsymmetricDtmfPayloadType* to allow asymmetric payload type (see "Asymmetric DTMF Payload Type" on page 310 for more details).

### Asymmetric DTMF Payload Type

The default behaviour when receiving an answer to an offer that contained an RFC 2833 payload type is to use that payload type symmetrically (to send and receive DTMF). You can set the Mediatrix 4102 to rather use the payload type that was placed in the initial offer to receive DTMF but still use the one in the response to send them.

▶ **To use the asymmetric DTMF payload type:**

1. In the *sipInteropMIB*, set the *sipInteropAllowAsymmetricDtmfPayloadType* variable with the proper behaviour.

**Table 203:** Asymmetric DTMF Payload Type

| Parameter | Description |
|---|---|
| disable | Use the RFC 2833 payload type found in the received answer to receive DTMF. This is the default value. |
| enable | Keep using the initial payload type to receive DTMF. |

The variable does not affect the behaviour when receiving an offer. It only has an effect when the variable *sipInteropUseDtmfPayloadTypeFoundInAnswer* is set to **enable** (see "Using the Payload Type Found in the Answer" on page 309 for more details).

## Controlling the Call Waiting Tone via SIP INFO

The Mediatrix 4102 supports receiving some Call Waiting control commands via the SIP INFO method. Currently, the only supported content-type is "application/broadsoft".

The controlled call waiting tone is played through the telephony interface.

▶ **To control the call waiting tone via SIP INFO:**

1. In the *sipInteropMIB*, set the *sipInteropCallWaitingToneControlViaSipInfo* variable with the proper behaviour.

**Table 204:** Call Waiting Tone Control

| Parameter | Description |
|---|---|
| disable | The application rejects the SIP INFO with the content type 'application/broadsoft'. This is the default value. |
| enable | SIP INFO with the content type 'application/broadsoft' is accepted and the call waiting tone is started or stopped according to the body. The INFO method is also included in the 'Allow' SIP header. |

## Ignore Username Parameter

You can control whether or not the username parameter is ignored when routing an incoming SIP call to a line.

The default behaviour in the interpretation of the username parameter of an incoming request for the routing purpose is to compare the entire username. You can configure the Mediatrix 4102 so that it ignores the username parameter in the comparison.

The username parameters are all characters following a ';' in the username part of the URI.

▶ **To ignore the username parameter:**

1. In the *sipInteropMIB*, set the *sipInteropIgnoreUsernameParam* variable with the proper behaviour.

**Table 205:** Ignore Username Parameter

| Parameter | Description |
|---|---|
| disable | The username parameter is not ignored when routing an incoming SIP call to a line. The parameter is considered as part of the username. This is the default value. |
| enable | The username parameter is ignored when routing an incoming SIP call to a line. |

## Escaping the Pound (#) Character in SIP URI Username

You can define whether or not the pound character (#) must be escaped in the username part of a SIP URI

▶ **To define whether or not the pound character must be escaped:**

1. In the *sipInteropMIB*, set the *sipInteropEscapePoundInSipUriUsername* variable with the proper behaviour.

**Table 206:** Escaping Pound Character Parameter

| Parameter | Description |
|---|---|
| Enable | The Pound character (#) is escaped in the username part of a SIP URI. |
| Disable | The Pound character (#) is not escaped in the username part of a SIP URI.<br>Note that RFC 3261 specifies that the pound character (#) needs to be escaped in the username part of a SIP URI. |

## SIP OPTIONS Method Support

You can define the behaviour of the Mediatrix 4102 when answering a SIP OPTIONS request.

▶ **To define the SIP OPTIONS method support:**

1. In the *sipInteropMIB*, set the *sipInteropSipOptionsMethodSupport* variable with the proper behaviour.

**Table 207:** SIP OPTIONS Method Support Parameter

| Parameter | Description |
|---|---|
| none | The Mediatrix 4102 responds with an error 405 Method not allowed. |
| alwaysOk | The Mediatrix 4102 responds with a 200 OK regardless of the content of the OPTIONS request. |

# Offer/Answer Model

| | |
|---|---|
| **Standards Supported** | • RFC 3264 – An Offer/Answer Model with the Session Description Protocol (SDP) |

You can define whether or not the Mediatrix 4102 requires strict adherence to RFC 3264 from the peer when negotiating capabilities for the establishment of a media session.

▶ **To define how to process the Offer/Answer model:**

1. In the *sipExperimentalMIB*, set the *sipEnforceOfferAnswerModel* variable to the proper value.

**Table 208:** Offer/Answer Model Parameters

| Parameter | Description |
|---|---|
| disable | The peer can freely:<br>• Send back a brand new list of codecs or add new ones to the offered list.<br>• Add media lines AFTER the ones found in the offer.<br>As long as at least one codec sent back is supported by the Mediatrix 4102, the call is allowed to go on. Any media line added by the peer is simply ignored. |
| enable | The following guidelines from the Offer-Answer Model must be strictly followed. An answer must:<br>• Include at least one codec from the list that the Mediatrix 4102 sent in the offer.<br>• Avoid adding extra codecs that were not present in the offer.<br>• Contain the same number of media lines that the unit put in its offer.<br>Otherwise, the answer is rejected and the unit ends the call. This is the default value. |

The *sipAllowMediaReactivationInAnswerEnable* (see "Allow Media Reactivation in Answer" on page 312) and *sipAllowAudioAndImageNegotiationEnable* (see "Allow Audio and Image Negotiation" on page 313) variables allow to enable or disable other deviations from the Offer/Answer model.

## Allow Media Reactivation in Answer

You can define the behaviour of the Mediatrix 4102 when receiving a SDP answer activating a media that had been previously deactivated in the offer.

You can also set this parameter via the web interface, as described in "Interop Parameters" on page 83.

▶ **To define the behaviour when receiving a SDP answer:**

1. In the *sipExperimentalMIB*, set the *sipAllowMediaReactivationInAnswerEnable* variable with the proper behaviour.

**Table 209:** Media Reactivation Parameters

| Parameter | Description |
|---|---|
| Enable | A media reactivated in an incoming answer is ignored. This behaviour goes against the SDP Offer/Answer model described by IETF RFC 3264. |
| Disable | A media reactivated in an incoming answer ends the current media negotiation and the call. This behaviour follows the SDP Offer/Answer model described by IETF RFC 3264. |

The *sipEnforceOfferAnswerModel* (see "Offer/Answer Model" on page 312) and *sipAllowAudioAndImageNegotiationEnable* (see "Allow Audio and Image Negotiation" on page 313) variables allow to enable or disable other deviations from the Offer/Answer model.

## Allow Audio and Image Negotiation

You can define the behaviour of the Mediatrix 4102 when offering media or answering to a media offer with audio and image negotiation.

You can also set this parameter via the web interface, as described in "Interop Parameters" on page 83.

▶ **To allow audio and image negotiation:**

1. In the *sipExperimentalMIB*, set the *sipAllowAudioAndImageNegotiationEnable* variable with the proper behaviour.

**Table 210:** Audio and Image Negotiation Parameters

| Parameter | Description |
|---|---|
| Enable | The unit offers audio and image media simultaneously in outgoing SDP offers and transits to T.38 mode upon reception of a T.38 packet. Also, when the unit answers positively to a SDP offer with audio and image, it transits to T.38 mode upon reception of a T.38 packet. |
| Disable | Outgoing offers never include image and audio simultaneously. Incoming offers with audio and image media with a non-zero port are considered as offering only audio. |

The *sipEnforceOfferAnswerModel* (see "Offer/Answer Model" on page 312) and *sipAllowMediaReactivationInAnswerEnable* (see "Allow Media Reactivation in Answer" on page 312) variables allow to enable or disable other deviations from the Offer/Answer model.

## Codec Order in Answer

You can define the behaviour of the Mediatrix 4102 when answering to a media offer.

▶ **To define the codec order in answer:**

1. In the *sipExperimentalMIB*, set the *sipCodecOrderInAnswer* variable with the proper behaviour.

**Table 211:** Codec Order in Answer Parameters

| Parameter | Description |
|---|---|
| localOrder | The codecs contained in the answer are prioritized according to the configured preferred codec. This means that the codecs in the answer may have a different order than in the received offer. |
| offerOrder | The codecs contained in the answer have the same order as the received offer. This means that the preferred codec configured has no effect on the codec order of the answer. |

CHAPTER 20

# STUN Configuration

This chapter describes how to configure the STUN client of the Mediatrix 4102.

## What is STUN?

| | |
|---|---|
| **Standards Supported** | RFC 3489 – STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs) |

STUN (Simple Traversal of UDP through NATs) is a simple client / server protocol that uses UDP packets to discover the configuration information of NATs and firewalls between a device and the public Internet:

- NAT type
- NAT binding public address
- NAT binding time to live

NAT (Network Address Translator) is a device that translates the IP address used within a “private” network to a different IP address known in another “public” network. See “NAT Traversal” on page 292 for more details.

STUN supports a variety of existing NAT devices and does not require any additional hardware or software upgrades on the NAT device.

The Mediatrix 4102 uses the STUN protocol to discover its NAT binding for the following three IP addresses/ports (sockets):

- Signalling protocol (SIP) IP address/port
- RTP IP address/port
- T.38 IP address/port

### SIP Outbound Proxy

For a unit to work properly behind a firewall, it must keep a pinhole opened by sending keepalive packets through the firewall.

The Mediatrix 4102 only sends keepalive packets to the last destination for a specific socket. When a unit is not configured with an outbound proxy, it can send, through its SIP socket, messages to various destinations, such as a SIP redirect server, another SIP unit, or a MWI server. If, for instance, the last SIP message was sent to the MWI server, the Mediatrix 4102 will keep the pinhole opened for the MWI server only (sending keepalive message to the MWI server) and won't be reacheable by other units outside the firewall.

To avoid those issues, all SIP message should come and go from the same source/destination on the public side of the firewall, i.e., a SIP outbound proxy. Media5 thus recommends that you use a SIP outbound proxy. See “Outbound Proxy Server” on page 180 for more details.

### Restrictions on the Media5 STUN Implementation

- The Mediatrix 4102 does not currently support NAT type discovery.
- The Mediatrix 4102 does not currently support STUN NAT binding time to live discovery.
- The Mediatrix 4102 does not currently support the TLS security mechanism.
- Due to a limitation of most routers, an RTP portal might be required in order for two units behind the same NAT/firewall to be able to communicate with each other.

# STUN Client Configuration

The following describes how to configure the Mediatrix 4102 STUN client via SNMP. You can also use the web interface to configure the STUN parameters. See "STUN Page" on page 38 for more details.

**To configure the STUN client:**

1. In the *ipAddressConfig* folder, locate the *ipAddressConfigStunStatic* group.

   No DHCP value is available, you can only define STUN server information with static values.

2. Set the static STUN server IP address or Fully Qualified Domain Name (FQDN) in the *stunStaticHost* variable.

   The default value is **192.168.0.10**.

3. Set the static STUN server IP port number in the *stunStaticPort* variable.

   The default value is **3478**.

4. In the *stunMIB*, set the amount of time, in seconds, the Mediatrix 4102 should keep a STUN query result in its internal cache in the *stunQueryCacheDuration* variable.

   Keeping a query in a cache helps to reduce the amount of unnecessary STUN queries when an RTP or T.38 socket is re-used within a short period of time. Available values range from 0 s to 3600 s.

   When set to **0**, the cache is disabled and the unit performs a STUN query each time a socket needs to be used.

5. Set the maximum amount of time, in milliseconds, the Mediatrix 4102 should wait for an answer to a STUN query sent to a STUN server in the *stunQueryTimeout* variable.

   Available values range from 500 ms to 10000 ms. The default value is 1000 ms.

   Caution is advised in setting long timeouts. In the advent of an unresponsive STUN server, the unit may end up waiting a long time before it determines that a call cannot be made due to the STUN server failure.

6. Define the interval, in seconds, at which the Mediatrix 4102 sends blank keepalive messages to keep a firewall hole opened in the *stunKeepAliveInterval* variable.

   Keepalive messages are used by both the signalling protocol socket and the RTP socket to keep those connections opened through a firewall. Available values range from 0 s to 120 s. The default value is 30 s.

   When set to **0**, no keepalive packet is sent.

**Note:** Keepalive messages are not supported on the T.38 socket.

7. Enable the STUN client by setting the *stunEnable* variable to **enable**.

   This enables the STUN client for all sockets (VoIP signalling, RTP and T.38) altogether.

   The following behaviour also applies:

   - If a unit is unable to re-register and there are no ongoing calls, it tries to rediscover its NAT binding for the signalling protocol socket.
   - If a STUN server is unresponsive, it is put in a "penalty box" for 60 seconds. See "SIP Penalty Box" on page 295 for more details.

8. Restart the Mediatrix 4102 so that the changes may take effect.

CHAPTER 21

# SNTP Settings

The Simple Network Time Protocol (SNTP) enables the notion of time (date, month, time) into the Mediatrix 4102. It updates the internal clock of the unit, which is the client of a SNTP server. It is required when dealing with features such as the caller ID.

SNTP is used to synchronize a SNTP client with a SNTP or NTP server by using UDP as transport.

You can also set these parameters via the web interface, as described in "SNTP Settings" on page 51.

## Enabling the SNTP Client

| Standards Supported | RFC 1769 – Simple Network Time Protocol (SNTP) |
|---|---|

You must enable the SNTP client of the Mediatrix 4102 to properly connect to a a SNTP or NTP server.

▶ **To enable the SNTP feature:**

1. In the *sntpMIB*, set the *sntpEnable* variable to **enable**.
2. Set the following synchronization information:

**Table 212:** SNTP Synchronization Information

| Variable | Description |
|---|---|
| sntpSynchronizationPeriod | Time interval (in minutes) between requests made to the SNTP server. The result is used to synchronize the unit with the time server. The maximum value is set to 1440 minutes (24 hours).<br>**Default Value**: 1440 |
| sntpSynchronizationPeriodOnError | Time interval (in minutes) between retries after an unsuccessful attempt to reach the SNTP server. The maximum value is set to 1440 (24 hours).<br>**Default Value**: 60 |

# Configuration Source

The Mediatrix 4102 must know the IP address and port number of the SNTP server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *SNTP*.

## DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

▶ **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfig Source* variable (under the *ipAddressConfigSntp* group).

   This variable defines whether the Mediatrix 4102 must ask for its SNTP server settings through a DHCP server or not.

2. Set the *sntpSelectConfigSource* variable to **dhcp**.

   You can query the SNTP server's IP address and port number assigned by the DHCP server in the *sntpHost* and *sntpPort* read-only variables (under the *ipAddressStatusSntp* group).

3. Set the DHCP Vendor Specific code of the SNTP feature in your DHCP server.

   See "SNTP" on page 165 for more details.

## Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfig Source* variable (under the *ipAddressConfigSntp* group).

   This variable defines whether the Mediatrix 4102 must ask for its SNTP server settings through a DHCP server or not.

2. Set the *sntpSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 213:** SNTP Static Address

| Variable | Description |
|---|---|
| sntpStaticHost | Static SNTP server IP address or domain name.<br>**Default Value**: 192.168.0.10 |
| sntpStaticPort | Static SNTP server IP port number.<br>**Default Value**: 123 |

# Defining a Custom Time Zone

| **Standards Supported** | bootp-dhcp-option-88.txt Internet draft |
|---|---|

When starting, the Mediatrix 4102 queries a NTP or SNTP server to receive time information. It receives the information in Greenwich Mean Time (GMT) format (also known as Universal Time Coordinated - UTC), so it needs to convert this GMT time into the proper time zone. To do this, the Mediatrix 4102 offers time zone configuration with daylight saving settings.

**To define a custom time zone:**

1. In the *sntpMIB*, enter a valid POSIX (Portable Operating System Interface) string in the *sntpTimeZoneString* variable as defined in the <bootp-dhcp-option-88.txt> Internet draft.

   The format of the string is validated upon entry. Invalid entries are refused. The default value is:

   ```
   EST5DST4,M4.1.0/02:00:00,M10.5.0/02:00:00
   ```

   A POSIX string is a set of standard operating system interfaces based on the UNIX operating system. The format of the IEEE 1003.1 POSIX string is defined in the <bootp-dhcp-option-88.txt> Internet draft as:

   ```
   STDOFFSET[DST[OFFSET],[START[/TIME],END[/TIME]]]
   ```

   Refer to the following sub-sections for explanations on each part of the string.

## STD / DST

Three or more characters for the standard (STD) or alternative daylight saving time (DST) time zone. Only STD is mandatory. If DST is not supplied, the daylight saving time does not apply. Lower and upper case letters are allowed. All characters are allowed except digits, leading colon (:), comma (,), minus (-), plus (+), and ASCII NUL.

## OFFSET

Difference between the GMT time and the local time. The offset has the format *h[h][:m[m][:s[s]]]*. If no offset is supplied for DST, the alternative time is assumed to be one hour ahead of standard time. One or more digits can be used; the value is always interpreted as a decimal number.

The hour value must be between 0 and 24. The minutes and seconds values, if present, must be between 0 and 59. If preceded by a minus sign (-), the time zone is east of the prime meridian, otherwise it is west, which can be indicated by the preceding plus sign (+). For example, New York time is GMT 5.

## START / END

Indicates when to change to and return from the daylight saving time. The *START* argument is the date when the change from the standard to the daylight save time occurs; *END* is the date for changing back. If *START* and *END* are not specified, the default is the US Daylight saving time start and end dates. The format for start and end must be **one** of the following:

- **n** where *n* is the number of days since the start of the year from 0 to 365. It must contain the leap year day if the current year is a leap year. With this format, you are responsible to determine all the leap year details.
- **Jn** where *n* is the Julian day number of the year from 1 to 365. Leap days are not counted. That is, in all years – including leap years – February 28 is day 59 and March 1 is day 60. It is impossible to refer to the occasional February 29 explicitly. The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is *02:00:00*.
- **Mx[x].y.z** where *x* is the month, *y* is a week count (in which the z day exists) and *z* is the day of the week starting at 0 (Sunday). As an example:

  ```
  M10.4.0
  ```

  is the fourth Sunday of October. It does not matter if the Sunday is in the 4th or 5th week.

  ```
  M10.5.0
  ```

is the last Sunday of October (5 indicates the last z day). It does not matter if the Sunday is in the 4th or 5th week.

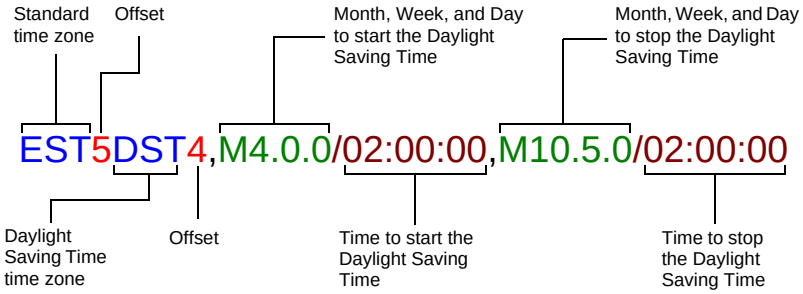
`M10.1.6`

is the first week with a Saturday (thus the first Saturday). It does not matter if the Saturday is in the first or second week.

The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If TIME is not specified, the default is *02:00:00*.

## Example

The following is an example of a proper POSIX string:




The following are some valid POSIX strings:

**Table 214:** Valid POSIX Strings

| Time Zone | POSIX String |
|---|---|
| Pacific Time (Canada & US) | PST8PDT7,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Mountain Time (Canada & US) | MST7MDT6,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Central Time (Canada & US) | CST6CDT5,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Eastern Time Canada & US) | EST5EDT4,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| Atlantic Time (Canada) | AST4ADT3,M3.2.0/02:00:00,M11.1.0/02:00:00 |
| GMT Standard Time | GMT0DMT-1,M3.5.0/01:00:00,M10.5.0/02:00:00 |
| W. Europe Standard Time | WEST-1DWEST-2,M3.5.0/02:00:00,M10.5.0/03:00:00 |
| China Standard Time | CST-8 |
| Tokyo Standard Time | TST-9 |
| Central Australia Standard Time | CAUST-9:30DCAUST-10:30,M10.5.0/02:00:00,M3.5.0/02:00:00 |
| Australia Eastern Standard Time | AUSEST-10AUSDST-11,M10.5.0/02:00:00,M3.5.0/02:00:00 |
| UTC (Coordinated Universal Time) | UTC0 |

CHAPTER 22

# Digit Maps

This chapter describes how to use a digit map.

| | |
|---|---|
| **Standards Supported** | RFC 2705 – Media Gateway Control Protocol (MGCP) Version 1.0, section 3.4 (Formal syntax description of the protocol). |

In the *Unit Manager Network Administration Manual*, refer to chapter *Dial Map Parameters*.

You can also set these parameters via the web interface, as described in "Digit Maps" on page 87.

## What is a Digit Map?

A digit map allows you to compare the number users just dialed to a string of arguments. If they match, users can make the call. If not, users cannot make the call and get an error signal. It is thus essential to define very precisely a digit map before actually implementing it, or your users may encounter calling problems.

Because the Mediatrix 4102 cannot predict how many digits it needs to accumulate before transmission, you could use the digit map, for instance, to determine exactly when there are enough digits entered from the user to place a call.

## Syntax

The permitted digit map syntax is taken from the core MGCP specification, RFC 2705, section 3.4:

```
DigitMap = DigitString  / '(' DigitStringList ')'
DigitStringList = DigitString 0*( '|' DigitString )
DigitString = 1*(DigitStringElement)
DigitStringElement = DigitPosition ['.']
DigitPosition = DigitMapLetter / DigitMapRange
DigitMapLetter = DIGIT / '#' / '*' / 'A' / 'B' / 'C' / 'D' / 'T'
DigitMapRange =  'x' / '[' 1*DigitLetter ']'
DigitLetter ::= *((DIGIT '-' DIGIT ) / DigitMapLetter)
```

Where "x" means "any digit" and "." means "any number of".

For instance, using the telephone on your desk, you can dial the following numbers:

**Table 215:** Number Examples

| Number | Description |
|---|---|
| 0 | Local operator |
| 00 | Long distance operator |
| xxxx | Local extension number |
| 8xxxxxxx | Local number |
| #xxxxxxx | Shortcut to local number at other corporate sites |
| 91xxxxxxxxxx | Long distance numbers |
| 9011 + up to 15 digits | International number |

The solution to this problem is to load the Mediatrix 4102 with a digit map that corresponds to the dial plan.

A Mediatrix 4102 that detects digits or timers applies the current dial string to the digit map, attempting a match to each regular expression in the digit map in lexical order.

- If the result is under-qualified (partially matches at least one entry in the digit map), waits for more digits.
- If the result matches, dials the number.
- If the result is over-qualified (i.e., no further digits could possibly produce a match), sends a fast busy signal.

# Special Characters

Digit maps use specific characters and digits in a particular syntax. Those characters are:

**Table 216:** Digit Map Characters

| Character | Use |
|---|---|
| Digits (0, 1, 2... 9) | Indicates specific digits in a telephone number expression. |
| T | The Timer indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the SIP Server can make the call. |
| x | Matches any digit, excluding “#” and “*”. |
| \| | Indicates a choice of matching expressions (OR). |
| . | Matches an arbitrary number of occurrences of the preceding digit, including 0. |
| [ | Indicates the start of a range of characters. |
| ] | Indicates the end of a range of characters. |

# How to Use a Digit Map

Let’s say you are in an office and you want to call a co-worker’s 3-digits extension. You could build a digit map that says “after the user has entered 3 digits, make the call”. The digit map could look as follows:

```
xxx
```

You could refine this digit map by including a range of digits. For instance, you know that all extensions in your company either begin with 2, 3, or 4. The corresponding digit map could look as follows:

```
[2-4]xx
```

If the number you dial begins with anything other than 2, 3, or 4, the call is not placed and you get a busy signal.

## Combining Several Expressions

You can combine two or more expressions in the same digit map by using the “|” operator, which is equal to OR.

Let’s say you want to specify a choice: the digit map is to check if the number is internal (extension), or external (a local call). Assuming that you must first dial “9” to make an external call, you could define a digit map as follows:

```
([2-4]xx|9[2-9]xxxxxx)
```

The digit map checks if:

- the number begins with 2, 3, or 4 **and**
- the number has 3 digits

If not, it checks if:

- the number begins with 9 **and**
- the second digit is any digit between 2 and 9 **and**
- the number has 7 digits

**Note:** Enclose the digit map in parenthesis when using the "|" option.

## Using the # and * Characters

It may sometimes be required that users dial the "#" or "*" to make calls. This can be easily incorporated in a digit map:

```
xxxxxxx#
xxxxxxx*
```

The "#" or "*" character could indicate users must dial the "#" or "*" character at the end of their number to indicate it is complete. You can specify to remove the "#" or "*" found at the end of a dialed number. See "Setting up Digit Maps" on page 324.

## Using the Timer

You can configure the Timer. See "Digit Maps Timeouts" on page 326 for more details. It indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the Mediatrix 4102 can make the call. A digit map for this could be:

```
[2-9]xxxxxxT
```

**Note:** When making the actual call and dialing the number, the Mediatrix 4102 automatically removes the "T" found at the end of a dialed number, if there is one (after a match). This character is for indication purposes only.

## Calls Outside the Country

If your users are making calls outside their country, it may sometimes be hard to determine exactly the number of digits they must enter. You could devise a digit map that takes this problem into account:

```
001x.T
```

In this example, the digit map looks for a number that begins with 001, and then any number of digits after that (x.).

## Example

Table 215 on page 321 outlined various call types one could make. All these possibilities could be covered in one digit map:

```
(0T|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|91xxxxxxxxxx|9011x.T)
```

## Validating a Digit Map

The Mediatrix 4102 validates the digit map as you are entering it and it forbids any invalid value.

# Processing Digits When Pressed

You can define whether or not a call is made once all DTMFs have been verified against a DTMF map.

**To define how to process digit maps:**

1. In the *digitMapMIB*, define when a digit is processed through the digit maps in the *digitMapProcessDigitsWhenPressed* variable.

**Table 217:** Processing Digits Parameters

| Parameter | Description |
|---|---|
| disable | Digits are processed only when released. Disabling this feature increases the delay needed to match a dialed string to a digit map.<br>There is also an impact on the *digitMapTimeoutFirstDigit* and *digitMapTimeoutCompletion* parameters ("Digit Maps Timeouts" on page 326) since the timers are stopped at the end of a DTMF instead of the beginning.<br>It can also lead to small DTMF leaks when using subscriber services if the variable *subscriberServicesProcessingTrigger* is set to **flashHookAndDigits** ("Service Activation Processing" on page 333). |
| enable | Digits are processed as soon as they are pressed. This can lead to a DTMF leak in the RTP at the beginning of a call if the voice stream is established before the last DTMF is released. This is the default value. |

# Setting up Digit Maps

The variables related to the digit maps are located in tables. You can create/edit ten digit maps for each Mediatrix 4102. Before changing a parameter value, build its corresponding table with your MIB browser's table functionality. Depending on the MIB browser you are using, the tables may not appear the same way.

Digit map rules are checked sequentially. If a telephone number potentially matches two of the rules, the first rule encountered is applied.

Each of these digit map rules has six specific variables to define for the digit map to work properly.

**To set up digit maps:**

1. In the *digitMapMIB*, define the digit map string that is considered valid when dialed in the *digitMapAllowedDigitMap* variable.

   The string must use the syntax described in "Digit Maps" on page 321. The string format is validated upon entry. Invalid entries are refused. The default value is **x.T**. A digit map string may have a maximum of 64 characters.

2. Define the amount of digits to remove from the beginning of the dialed number, after dialing, but before initiating the call, in the *digitMapPrefixedDigitRemovalCount* variable.

   For instance, when dialing "1-819-xxx-xxxx", specifying a value of "4" means that the call is started by using the number "xxx-xxxx". The default value is **0**.

   This rule is applied BEFORE applying both *digitMapSuffixStringToRemove* (Step 3) and *digitMapPrependedString* (Step 4).

3. Define the string to look for and remove, from the end of the dialed number, in the *digitMapSuffixStringToRemove* variable.

   This is helpful if one of the digit maps contains a terminating character that must not be dialed.

   For instance, let's take a digit map such as "25#", in which the "#" signals that the user has finished entering digits. If you want to remove the "#", specify "#" in this variable and the resulting number is "25".

This rule is applied AFTER applying *digitMapPrefixedDigitRemovalCount* (Step 2) but BEFORE applying *digitMapPrependedString* (Step 4).

4. Define the string to insert at the beginning of the dialed number before initiating the call in the *digitMapPrependedString* variable.

   For instance, let's say that you need to dial a special digit, "9", for all local calls. Dialing "xxx-xxxx" with a value of "9" would yield "9-xxx-xxxx" as the number with which to initiate the call.

   This rule is applied AFTER applying both *digitMapPrefixedDigitRemovalCount* (Step 2) and *digitMapSuffixStringToRemove* (Step 3).

5. Specify the line(s) on which to apply the digit map in the *digitMapAllowedLineToApply* variable.

   The string has the following syntax:

   - ***all***: Applies to all lines.
   - ***,***: Separator between non-consecutive lists of lines or single line.
   - ***n***: A single line, where n is the line number.
   - ***m-n***: List of lines where m is the start line number and n is the end line number.

**Note:** Line duplication is not allowed. Lines must be specified in low to high order.

Example:

```
'1,3-6': Applies to lines 1, 3, 4, 5, and 6.
```

The default value is **all**.

6. Enable the digit map by setting the *digitMapAllowedEnable* variable to **enable**.

   When enabled, this digit map is recognised and accepted only if it is also valid.

## Refused Digit Maps

A refused digit map forbids your users to call specific numbers; for instance, you want to accept all 1-8xx numbers except 1-801. You can create/edit ten refused digit maps for each Mediatrix 4102.

▶ **To set up refused digit maps:**

1. In the *digitMapMIB*, define the digit map string that is considered invalid when dialed in the *digitMapRefusedDigitMap* variable.

   The string must use the syntax described in "Digit Maps" on page 321. The string format is validated upon entry. Invalid entries are refused. A digit map string may have a maximum of 64 characters.

2. Specify the line(s) on which to apply the digit map in the *digitMapRefusedLineToApply* variable.

   The string has the following syntax:

   - ***all***: Applies to all lines.
   - ***,***: Separator between non-consecutive lists of lines or single line.
   - ***n***: A single line, where n is the line number.
   - ***m-n***: List of lines where m is the start line number and n is the end line number.

**Note:** Line duplication is not allowed. Lines must be specified in low to high order.

Example:

```
'1,3-6': Applies to lines 1, 3, 4, 5, and 6.
```

The default value is **all**.

3. Enable the refused digit map by setting the *digitMapRefusedEnable* variable to **enable**.

   When enabled, this digit map is recognised and refused only if it is also valid.

### Digit Maps Timeouts

You can define timeouts that apply to the whole unit when dialing a digit map.

**To configure digit map timeouts:**

1. In the *digitMapMIB* (*digitMapTimeouts* group), define the total time the user has to dial the DTMF sequence in the *digitMapTimeoutCompletion* variable.

   The timer starts when the dial tone is played. When the timer expires, the receiver off-hook tone is played.

   This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms. The default value is **60000** ms.

2. Define the time between the start of the dial tone and the receiver off-hook tone, if no DTMF is detected, in the *digitMapTimeoutFirstDigit* variable.

   This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms. The default value is **20000** ms.

3. Define the value of the “T” digit in the *digitMapTimeoutInterDigit* variable.

   The “T” digit is used to express a time lapse between the detection of two DTMFs.

   This value is expressed in milliseconds (ms). Values range from 500 ms to 10000 ms. The default value is **4000** ms.

## Digit Map Examples

### Digit Map Example 1 – Standard Calls

Let’s say you are located in Seattle, Washington and you want to define digit map rules for your users. You must consider at least four possibilities:

- You are making a long distance call outside the country.
- You are making a long distance call outside your area code.
- You are making a local call outside your area code (in the 425 area code).
- You are making a local call in the same area code.

#### Digit Map Rule #1

This digit map rule checks for calls outside the country.

**Table 218:** Digit Map Rules #1 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | (011x.#\|001x.T) |
| digitMapPrefixedDigitRemovalCount | 3<br>A valid telephone number must contain a country code, an area code, and a number – the “011” part is not required. |

## Digit Map Rule #2

This digit map rule checks for long distance calls outside your area code.

**Table 219:** Digit Map Rules #2 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | 1xxxxxxxxxx |
| digitMapPrefixedDigitRemovalCount | 1<br>The first digit “1” in the digit map indicates a user wants to call outside his or her own area code. It must be removed because it does not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number). |
| digitMapPrependedString | 1 (country code)<br>A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added.<br>Note that in this scenario, the country code is the same as the code used when the user wants to indicate a communication outside of his or her own area code. It is still good practice to have this number removed and to add the country code, even if these two numbers are the same. |

## Digit Map Rule #3

This digit map rule checks for local calls outside your area code (in the 425 Area Code).

**Table 220:** Digit Map Rules #3 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | 425xxxxxxx |
| digitMapPrependedString | 1 (country code)<br>A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added. |

## Digit Map Rule #4

This digit map rule checks for local calls in the same area code.

**Table 221:** Digit Map Rules #4 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | ([235-9]xxxxxx\|45[1-9]xxxx\|4[0-469]xxxxx) |
| digitMapPrependedString | 1206 (country code and area code)<br>A valid telephone number must contain a country code, an area code, and a number. The country code and area code are missing in this number and must be added. |

## Digit Map Example 2 – PBX Emulation

Let's say you are located in the 819 area code. You are in an office where you dial:

- 3 numbers to call one of your co-workers.
- "9" to get an external line.

The following four possibilities are considered:

- You are making an internal call to one of your co-workers.
- You are making a long distance call outside the country.
- You are making a long distance call outside your area code.
- You are making a local call in the same area code.

### Digit Map Rule #1

This digit map rule checks for internal calls.

**Table 222:** Digit Map Rules #1 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | [1-8]xx |

### Digit Map Rule #2

This digit map rule checks for calls outside the country.

**Table 223:** Digit Map Rules #2 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | (9011x.#\|9011x.T) |
| digitMapPrefixedDigitRemovalCount | 4<br>A valid telephone number must contain a country code, an area code, and a number – the "9011" part is not required. |

### Digit Map Rule #3

This digit map rule checks for long distance calls outside your area code.

**Table 224:** Digit Map Rules #3 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | 91xxxxxxxxx |
| digitMapPrefixedDigitRemovalCount | 2<br>The first digit "9" in the digit map indicates a user wants to make an external call, while the second digit "1" indicates a user wants to call outside his or her own area code (in North America). The two digits must be removed because they do not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number). |

**Table 224:** Digit Map Rules #3 Settings (Continued)

| Variable | Setting |
|---|---|
| digitMapPrependedString | 1 (country code)<br>A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added.<br>Note that in this scenario, the country code is the same as the code used when the user wants to indicate a communication outside of his or her own area code. It is still good practice to have this number removed and to add the country code, even if these two numbers are the same. |

## Digit Map Rule #4

This digit map rule checks for local calls in the same area code.

**Table 225:** Digit Map Rules #4 Settings

| Variable | Setting |
|---|---|
| digitMapAllowedEnable | Enable |
| digitMapAllowedDigitMap | 9[2-8]xxxxxx |
| digitMapPrefixedDigit RemovalCount | 1<br>The first digit “9” in the digit map indicates a user wants to make an external call. It has to be removed because it does not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number). |
| digitMapPrependedString | 1819 (country code and area code)<br>A valid telephone number must contain a country code, an area code, and a number. The country code and area code are missing in this number and must be added. |

CHAPTER 23

# Telephony Features

This chapter explains how to set the telephony variables of the Mediatrix 4102 to define the way the unit handles calls.

## Making Calls

Users with telephones or faxes connected to a Mediatrix 4102 dial as if they were on a standard telephony system.

### Complete Dialing Sequence

There are three ways to indicate the dialed number sequence is complete and the Mediatrix 4102 can dial the number:

- The administrator has set up the dialing process so that you must end the telephone number with a particular character to indicate it is complete, e.g., a “#”.
- The administrator has set up the dialing process with a timer. This timer checks the dialing process and, when no further digits have been dialed for the time set by the administrator, it assumes the number is complete and dials it.
- The administrator has set up the Mediatrix 4102 so it knows exactly how many digits it must collect before it places the call. It finds the number of digits to collect by looking at the first few numbers dialed. For example: a telephone number beginning by 1 should be followed by 10 more digits in North America.

### Dialing a Telephone Number or Numerical Alias

This section assumes that the Mediatrix 4102 is configured to do SCN emulation. The Mediatrix 4102 could be configured to do any other kind of emulation, thus its users would simply have to dial as if they were using their old system.

**To dial a Standard Call:**

1. Dial the telephone number as if you were using a standard telephone, with country code and area code when required.

   **Examples**:

   ```
   223
   8298749
   15145701234
   ```

   A Standard Call uses the server to contact the remote dialed user. The server takes the decision about redirecting the call on the SCN or keeping it on the network. Keeping the call on the network takes precedence over redirecting it on the SCN. If the call needs to go on the SCN, the server redirects it to a proper analog gateway (such as the Mediatrix 1204) that will place the call to the SCN network.

**Note:** You can dial one star numbers *xx (such as *69). These numbers are automatically inserted in the Request-URL of the SIP INVITE request.

▶ **To dial a Forced SCN call:**

1. Dial “**”.
2. Dial the telephone number as if you were using a standard telephone, with country code and area code when required.

   **Examples**:

   ```
   **8298749
   **15145701234
   ```

   A Forced SCN Call allows you to specify that the user you want to reach is located on the SCN network. This leaves no decision to the server; it must find a proper gateway and place the call on the SCN. This option can be useful only when a SCN number is shadowed by a network number.

**Note:** A forced SCN call is only be possible if an analog gateway such as the Mediatrix 1204 is available on the IP network.

# Emergency Call

The Emergency Call service (also called urgent gateway) allows a “911”-style service. It allows a user to dial a special digit map resulting in a message being sent to a specified urgent gateway, bypassing any other intermediaries.

If enabled, whenever the user dials the specified digit map, a message is sent to the target address.

You can also set these parameters via the web interface, as described in “Emergency Call Configuration” on page 132.

▶ **To enable the emergency call service:**

1. In the *emergencyCallMIB*, locate the *emergencyCallUrgentGatewayEnable* variable (under the *emergencyCallUrgentGatewayCustomization* group).

   This variable sets the usage state of the urgent gateway. Urgent messages bypass the outbound proxy and go directly to the urgent gateway.
2. Define the digits that users must dial to start the urgent gateway call feature in the *emergencyCallUrgentGatewayDigitMap* variable.

   For instance, you could decide to put “*60” as the sequence a user must dial to start the urgent gateway service. This sequence must follow the syntax for digit maps (see “Chapter 22 - Digit Maps” on page 321). Dialing this digit map does not have any effect unless the service's status is “enabled”.

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have different sequences for each line.
3. Set the number to reach for an urgent call in the *emergencyCallUrgentGatewayTargetAddress* variable.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   Note that this string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

CHAPTER 24

# Subscriber Services

The Mediatrix 4102 offers subscriber services users can directly access on their telephone. However, you must set these services before they can be used.

Most of the variables related to the subscriber services are located in tables. These tables display the information for all lines. Before changing a parameter value, build its corresponding table with your MIB browser's table functionality.

In the *Unit Manager Network Administration Manual*, refer to chapter *Subscriber Services Parameters*, section *SIP Configuration Window*.

You can also set these parameters via the web interface, as described in "Call Forward" on page 104 and "Services" on page 109.

## Service Activation Processing

The user can activate a service in two ways:

- By performing a standard flash hook.
- By performing a flash hook and entering a digit to activate a specific service. The digit dialed has a different behaviour depending on the current call context.

You can define which of these two methods is available to your users.

**To define the service activation processing:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesProcessingTrigger* variable.
2. Select which action the user must perform to trigger a service change by setting the *subscriberServicesProcessingTrigger* variable:

**Table 226:** Service Activation Parameters

| Parameter | Description |
|---|---|
| flashHook | The user must perform a flash hook to activate a service. |

**Table 226:** Service Activation Parameters (Continued)

| Parameter | Description |
|---|---|
| flashHookAndDigits | The user must perform a flash hook **and** enter a specific digit to activate a service. The digit dialed has a different behaviour depending on the current call context:<br>• One call active and one waiting call:<br>Flash hook then dial the digit 2: Answer the waiting call.<br>• One call active and one call on hold:<br>Flash hook then dial the digit 1: Terminate the active call and recover the call on hold.<br>Flash hook then dial the digit 2: Hold the active call and recover the call on hold.<br>Flash hook then dial the digit 3: Enter the conference mode.<br>Flash hook then dial the digit 4: Transfer the call on hold to the active call.<br>When hanging up in this context, the telephone rings to notify the user there is still a call on hold.<br>• In conference mode:<br>Flash hook then dial the digit 2: Return to one active call and one call on hold.<br>Flash hook then dial the digit 4: Transfer the second active call to the first active call.<br>When hanging up in this context, all calls are finished. |

As example, the following are the steps to perform a conference call:

1. Call the first attendee.
2. Flash hook to put the first attendee on hold.
3. Call the second attendee.
   The context is now one call active and one call on hold.
4. Flash hook then dial the digit 3 to start the conference call.

# Call Hold

The Call Hold service allows the user to temporarily put an existing call on hold, usually by using the “flash” button of the telephone. The user can resume the call in the same way.

You must enable this service for the following services to work properly:

- Call Waiting
- Second Call
- Blind Transfer
- Attended Transfer
- Conference

## Enabling Call Hold

You must enable this service before your users can use it.

▶ **To enable the call hold service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesHoldEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   You can find the current status of the service in the *subscriberServicesHoldStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

## Using Call Hold

The following is the procedure to use this service on the user's telephone.

▶ **To put the current call on hold:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.

   This puts the call on hold. You can resume the call in the same way.

# Second Call

The Second Call service allows a user with an active call to put the call on hold, and then initiate a new call on a second line. This service is most useful with the transfer and conference services.

The call hold service must be enabled for this service to work. See "Call Hold" on page 334.

## Enabling Second Call

You must enable this service before your users can use it.

▶ **To enable the second call service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesSecondCallEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   You can find the current status of the service in the *subscriberServicesSecondCallStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

## Using Second Call

The following is the procedure to use this service on the user's telephone.

▶ **To use the second call service:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.

   This puts the call on hold and the second line is automatically connected to your line.
2. Initiate the second call.

# Call Forward

The Call Forward service offers various ways to forward calls:

- Unconditional
- On Busy
- On No Answer

## Unconditional

The Call Forward Unconditional feature allows users to forward all of their calls to another extension or line.

### Setting up Call Forward Unconditional

You must condigure and enable this service before your users can use it.

**To set the Call Forward Unconditional feature:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardUnconditionalActivation* variable to **inactive** or **active**.

   This variable starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardUnconditionalActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardUnconditionalEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*70" as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardUnconditionalDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*71" as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardUnconditionalForwardingAddress* variable.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

   Because this variable is located in a table, you can have a different string for each line.

6. Enable the Call Forward Unconditional by setting the *subscriberServicesCallForwardUnconditionalEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).

   If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

## Using Call Forward Unconditional

When forwarding calls outside the system, a brief ring is heard on the telephone to remind the user that the call forward service is active. The user can still make calls from the telephone.

**To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward unconditional service.

   This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three “beeps” followed by a silent pause.

   The call forward is established.
7. Hang up your telephone.

   The calls are checked against the digit maps set up by the system administrator.

**To check if the call forward has been properly established:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial your extension or telephone number.

   The call is forwarded to the desired telephone number.
4. Hang up your telephone.

**To cancel the call forward:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward – unconditional service.

   This sequence could be something like *71.
4. Wait for three “beeps” followed by a silent pause.

   The call forward is cancelled.
5. Hang up your telephone.

## On Busy

You can automatically forward the incoming calls of your users to a pre-determined target if they are already on the line. The user does not have any feedback that a call was forwarded.

### Setting up Call Forward On Busy

You must configure and enable this service before your users can use it.

▶ **To set the Call Forward On Busy feature:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardOnBusyActivation* variable to **inactive** or **active**.

   This variable starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardOnBusyActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardOnBusyEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*72" as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardOnBusyDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*73" as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardOnBusyForwardingAddress* variable.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

   Because this variable is located in a table, you can have a different string for each line.
6. Enable the Call Forward On Busy by setting the *subscriberServicesCallForwardOnBusyEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).

   If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

## Using Call Forward on Busy

The following is the procedure to use this service on the user's telephone.

**To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward on busy service.
   This sequence could be something like *72.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.
   The call forward is established.
7. Hang up your telephone.
   The calls are checked against the digit maps set up by the system administrator.

**To cancel the call forward:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward on busy service.
   This sequence could be something like *73.
4. Wait for three "beeps" followed by a silent pause.
   The call forward is cancelled.
5. Hang up your telephone.

## On No Answer

You can forward the incoming calls of your users to a pre-determined target if they do not answer their telephone before a specific amount of time. The user does not have any feedback that a call was forwarded.

### Setting up Call Forward On No Answer

You must configure and enable this service before your users can use it.

▶ **To set the Call Forward On No Answer feature:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardOnNoAnswerActivation* variable to **inactive** or **active**.

   This variable starts the service (active) or stops the service (inactive).

   If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardOnNoAnswerActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardOnNoAnswerEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*74" as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardOnNoAnswerDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).

   Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.

   For instance, you could decide to put "*75" as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have different sequences for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardOnNoAnswerForwardingAddress* variable.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

   Because this variable is located in a table, you can have a different string for each line.
6. Define the time, in milliseconds, the telephone keeps ringing before the call forwarding activates in the *subscriberServicesCallForwardOnNoAnswerTimeout* variable.

   The default value is 5000.
7. Enable the Call Forward On No Answer by setting the *subscriberServicesCallForwardOnNoAnswerEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).

If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

Because this variable is located in a table, you can enable/disable the service on a per-line basis.

## Using Call Forward on No Answer

The following is the procedure to use this service on the user's telephone.

▶ **To forward calls:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward on no answer service.

   This sequence could be something like *74.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.

   The call forward is established.
7. Hang up your telephone.

   The calls are checked against the digit maps set up by the system administrator.

▶ **To cancel the call forward:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward on no answer service.

   This sequence could be something like *75.
4. Wait for three "beeps" followed by a silent pause.

   The call forward is cancelled.
5. Hang up your telephone.

# Call Waiting

The call waiting tone indicates to an already active call that a new call is waiting on the second line.

Your users can activate/deactivate the call waiting tone for their current call. This is especially useful when transmitting faxes. The user that is about to send a fax can thus deactivate the call waiting tone to ensure that the fax transmission will not be disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

The user can also permanently activate/deactivate the call waiting service.

Furthermore, the Mediatrix 4102 supports receiving some Call Waiting control commands via the SIP INFO method. See "Controlling the Call Waiting Tone via SIP INFO" on page 310 for more details.

## Setting up Call Waiting

You must configure and enable this service before your users can use it.

▶ **To set the Call Waiting service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Enable the Call Waiting feature by setting the *subscriberServicesCallWaitingEnable* variable to **enable**.

   This permanently activates the call waiting tone. When receiving new calls during an already active call, a special tone is heard to indicate that a call is waiting on the second line. The user can then answer that call by using the "flash" button. The user can switch between the two active calls by using the "flash" button.

   The call hold service must be enabled for this service to work. See "Call Hold" on page 334.

   If the user is exclusively using faxes, put the variable to **disable** to permanently disable the call waiting tone.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   The user may cancel this service on a per-call basis when dialing a DTMF sequence matching the digit map stored in *subscriberServicesCallWaitingCancelDigitMap* (see Step 3). The user may also disable or enable this service permanently with the *subscriberServicesCallWaitingPermanentDigitMapEnable* and *subscriberServicesCallWaitingPermanentDigitMapDisable* digit maps (See Step 4).

   You can find the current status of the service in the *subscriberServicesCallWaitingStatus* read-only variable (under the *subscriberServicesIfStatusTable*).
3. Define the digits that users must dial to disable the Call Waiting tone in the *subscriberServicesCallWaitingCancelDigitMap* variable.

   This allows a user who has call waiting enabled to disable that service on the next call only. If, for any reason, the user wishes to undo the cancel, simply unhook and re-hook the telephone to reset the service.

   For instance, you could decide to put "*76" as the sequence to disable the call waiting tone. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321). Dialing this digit map does not have any effect unless the service's status is "enabled".

   The deactivating sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.
4. Define the digits that users must dial to enable the call waiting service permanently in the *subscriberServicesCallWaitingPermanentDigitMapEnable* variable.

   This activation is permanent until the user deactivates the service as in Step 5.

   For instance, you could decide to put "*84" as the sequence to enable the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321).

   The sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

When dialing this digit map, this sets the *subscriberServicesCallWaitingEnable* variable for the line the user is currently using to **enable**.

5. Define the digits that users must dial to disable the call waiting service permanently in the *subscriberServicesCallWaitingPermanentDigitMapDisable* variable.

   This deactivation is permanent until the user enables the service as in Step 4.

   For instance, you could decide to put "*85" as the sequence to disable the service. This sequence must be unique and follow the syntax for digit maps (see "Chapter 22 - Digit Maps" on page 321).

   The sequence is set for all the lines of the Mediatrix 4102. You cannot have a different sequence for each line.

   When dialing this digit map, this sets the *subscriberServicesCallWaitingEnable* variable for the line the user is currently using to **disable**.

## Using Call Waiting

The call waiting feature alerts the user if he or she is already on the telephone and a second call happens. A "beep" (the call waiting tone) is heard and repeated every ten seconds to indicate there is a second incoming call.

▶ **To put the current call on hold:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.

   This puts the call on hold and the second line is automatically connected to your line.

2. Answer the call on the second line.

▶ **To switch from one line to the other:**

1. Perform a Flash-Hook each time you want to switch between lines.

▶ **To terminate the first call before answering the second call:**

1. Hang up the telephone.
2. Wait for the telephone to ring.
3. Answer the telephone.

   The second call is on the line.

▶ **To terminate the active call and recover the call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
2. Dial the digit 1.

▶ **To hold the active call and recover the call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
2. Dial the digit 2.

### Removing the Call Waiting Tone

You can temporarily activate/deactivate the call waiting tone indicating a call is waiting. This is especially useful when transmitting faxes. If you are about to send a fax, you can thus deactivate the call waiting tone to ensure that the fax transmission is not disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

▶ **To deactivate the call waiting tone:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone.

This sequence could be something like *70.

4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
   The call waiting tone is disabled.

**To re-enable the call waiting tone:**

1. Take the receiver off-hook.
2. Replace the receiver on-hook.
   The call waiting tone is re-enabled.

## Permanently Removing the Call Waiting Tone

You can permanently activate/deactivate the call waiting service.

**To activate the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call waiting tone service.
   This sequence could be something like *84.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Hang up your telephone.
   The call waiting tone is enabled.

**To cancel the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone service.
   This sequence could be something like *85.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
   The call waiting is cancelled.
5. Hang up your telephone.

# Call Transfer

The Call Transfer service offers various ways to transfer calls:

- Blind Transfer
- Attended Transfer

The SIP protocol also offers to set transfer-related parameters. See "Call Transfer Capacity" on page 299 and "Referred-By Field" on page 302 for more details.

## Blind Transfer

The blind call transfer service is sometimes called Transfer without Consultation or Unattended Transfer. It allows a user to transfer a call on hold to a still ringing (unanswered) call. The individual at the other extension or telephone number does not need to answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See "Call Hold" on page 334 and "Second Call" on page 335.

### Enabling Blind Call Transfer

You must enable this service before your users can use it.

**To enable the blind transfer service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesBlindTransferEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   You can find the current status of the service in the *subscriberServicesBlindTransferStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

### Using Blind Call Transfer

The following is the procedure to use this service on the user's telephone.

**To transfer a current call blind:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.

   This puts the call on hold.
2. Wait for the transfer tone (three "beeps").
3. Dial the number to which you want to transfer the call.
4. Wait for the ringback tone, then hang up your telephone.

   The call is transferred. You can also wait for the third party to answer if you want. In this case, the call transfer becomes attended.

   If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.

   You are back with the first call and the third party is released.

**To transfer a call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
2. Dial the digit 4.

## Attended Transfer

The attended call transfer service is sometimes called Transfer with Consultation. It allows a user to transfer a call on hold to an active call. The individual at the other extension or telephone number must answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See "Call Hold" on page 334 and "Second Call" on page 335.

### Enabling Attended Call Transfer

You must enable this service before your users can use it.

▶ **To enable the attended transfer service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesAttendedTransferEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   You can find the current status of the service in the *subscriberServicesAttendedTransferStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

### Using Attended Call Transfer

The following is the procedure to use this service on the user's telephone.

▶ **To transfer a current call attended:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.

   This puts the call on hold.
2. Wait for the transfer tone (three "beeps").
3. Dial the number to which you want to transfer the call.

   The third party answers.
4. Hang up your telephone.

   The call is transferred.
5. If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.

   You are back with the first call and the third party is released.

**Note:** If the number to which you want to transfer the call is busy or does not answer, quickly perform a Flash-Hook. The busy tone or ring tone is cancelled and you are back with the first call.

▶ **To transfer a call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
2. Dial the digit 4.

# Conference Call

| **Standards Supported** | RFC 4579 – Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents |
|---|---|

The Conference Call service allows a user to link two or more calls together to form a single conversation, called a conference.

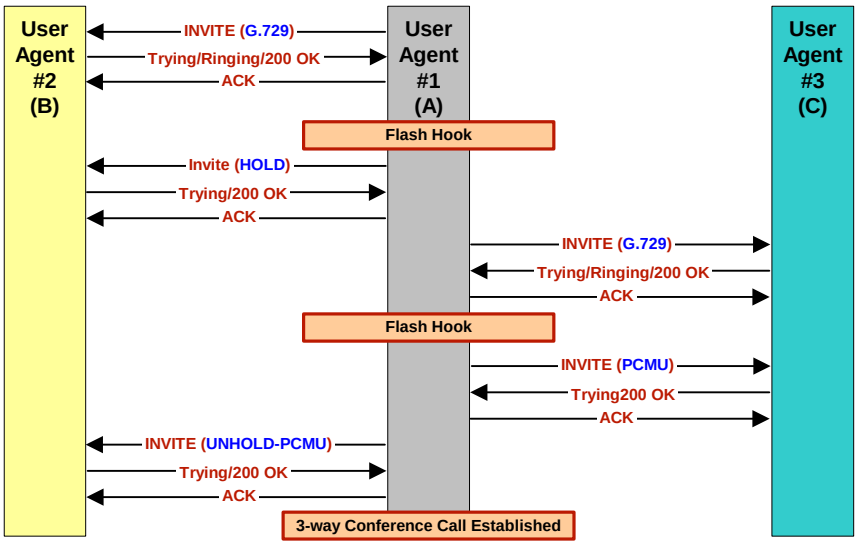
A participant of the conference can put the conference on hold and attempt other calls. This participant may then rejoin the conference at a later time by unholding it. The participant who initiated the conference cannot put it on hold.

You must enable the call hold and second call services for this service to work. See "Call Hold" on page 334 and "Second Call" on page 335.

Furthermore, you must also enable the Attended Transfer service for the two other participants to stay connected once the participant who initiated the conference has hung up. See "Attended Transfer" on page 346.

The following is a conference call flow example:

**Figure 102:** Conference Call Flow




## Requirements

For the conference call to occur successfully, all parties must meet the following requirements:

- Support at least one of the PCM codecs (G.711 µ-law and G.711 A-law) enabled on the line that is having the conference. See "Enabling Individual Codecs" on page 265 for more details.
- Ability to dynamically change codec during a call.
- The packetization period (ptime) should be the same for all the participants of the conference. If this is not the case, then part of the conversation may be lost, resulting in a choppy voice. For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See "Packetization Time" on page 266 for more information on how to set the packetization period of the Mediatrix 4102.

## Enabling the Conference Call Feature

You must enable this service before your users can use it.

**To enable the conference call service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesConferenceEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

   You can find the current status of the service in the *subscriberServicesConferenceStatus* read-only variable (under the *subscriberServicesIfStatusTable*).
3. In the *sipMIB*, locate the *sipConferenceConfigTable* group.
4. Specify how to manage a SIP conference in the *sipConferenceType* variable.

   This configuration only applies to a conference initiated by one of the unit's endpoint.

   **Table 227:** Conference Type Parameters

| Parameter | Description |
|---|---|
| local | The conference is managed locally by the unit. The maximum number of participants is 3. This is the default value. |
| conferenceServer | The conference is managed by a remote SIP conference server. When a Flash-Hook occurs, the server mechanism is determined by the variable *sipInteropConferenceServerMecanism* (see Step 6).<br>When using this conference type, both the initiator and a participant of the conference can add new participants to the conference. |

   This variable only has an effect when *subscriberServicesConferenceEnable* is set to enable.

   Because this variable is located in a table, you can set it on a per-line basis.
5. If you have selected **conferenceServer** in the previous step, set the *sipConferenceServerURI* variable with the address of a conference server.

   The format must be a SIP URI such as ”scheme:user@host”. For instance, “sip:user@foo.com”.

   Because this variable is located in a table, you can have a different string for each line.
6. If you have selected **conferenceServer** in Step 4, set the *sipInteropConferenceServerMechanism* variable (in the *sipInteropMIB*) with the mechanism used to establish a conference and how the participants are connected to the conference server.

   **Table 228:** Conference Server Mechanism

| Parameter | Description |
|---|---|
| rfc4579WithoutErrorRecovery | The connection with the conference server is made as defined in RFC 4579. The participants are connected to the conference server by sending the REFERs simultaneously. The connection with the participant is terminated if the participant fails to complete the REFER. |
| rfc4579WithErrorRecovery | The connection with the conference server is made as defined in RFC 4579. The participants are connected to the conference server by sending the REFERs sequentially. The REFER is not sent to the second participant and the call stays in the same state if the first participant fails to complete the REFER. This is the default value. |

## Managing a Conference Call

If you are on the telephone with one person and want to conference with a third one, you can do so. In the following examples, let's assume that:

- "A" is the conference initiator.
- "B" is the person called on the first line.
- "C" is the person called on the second line.
- "D" is a fourth person that "A" wants to add to the conference in **conferenceServer** conference type.

**To initiate a conference ("A" and "B" already connected):**

1. "A" performs a Flash-Hook.
   This puts "B" on hold and the second line is automatically connected. "A" hears a dial tone.
2. "A" dials "C's" number.
   "A" and "C" are now connected.
3. "A" performs another Flash-Hook.
   The call on hold ("B") is reactivated. "A" is now conferencing with "B" and "C".

**To initiate a conference with the flash hook and digit method ("A" and "B" already connected):**

1. "A" performs a flash hook.
   This puts "B" on hold and the second line is automatically connected. "A" hears a dial tone.
2. "A" dials "C's" number.
   The context is now one call active and one call on hold.
3. "A" performs a flash hook, and then dials the digit **3** to start the conference call.
   The call on hold ("B") is reactivated. "A" is now conferencing with "B" and "C".

**Note:** Performing a flash hook and dialing the digit **2** will stop conference but keep one active call and one call on hold.

**"A" wants to transfer "B" to "C" during the conference:**

This is available only in the **local** conference type.

1. "A" hangs up.
   The conference is terminated. "B" and "C" are now connected.

**"A" wants to terminate the call with "C" and get back to the call with "B" during the conference:**

This is available only in the **local** conference type.

1. "A" performs a Flash-Hook.
   The conference is terminated and the call with "C" is disconnected. "A" and "B" are still connected and can go on with their conversation.

**"B" (or "C") hangs up during the conference:**

This is available only in the **local** conference type.

1. "B" (or "C") hangs up during the conference.
   The conference is terminated, but the call between "A" and "C" (or "B") is not affected and they are still connected.

**"A" wants to add a fourth member to the conference:**

This is available only in the **converenceServer** conference type.

1. "A" performs a Flash-Hook.
   This puts "B" and "C" on hold and the second line is automatically connected. "A" hears a dial tone.

2. “A” dials “D’s” number.
   “A” and “D” are now connected.
3. “A” performs another Flash-Hook.
   The call on hold (“B” and “C”) is reactivated. “A” is now conferencing with “B”, “C”, and “D”.

CHAPTER 25

# Telephony Attributes

The telephony attributes are used to configure the characteristics of the telephony system being implemented.

## Automatic Call

The automatic call feature allows you to define a telephone number that is automatically dialed when taking the handset off hook.

In the *Unit Manager Network Administration Manual*, refer to chapter *Telephony Attributes Parameters*, section *Telephony Attributes Configuration Window*.

When this service is enabled, the second line service is disabled but the call waiting feature is still functional. The user can still accept incoming calls.

You can also set these parameters via the web interface, as described in "Automatic Call" on page 116.

▶ **To set the automatic call feature:**

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIfFeaturesTable* group.

   This group contains all of the variables required to set the automatic call feature.

2. Define the number to dial when the handset is taken off hook in the *telephonyAttributesAutomaticCallTargetAddress* variable.

   Accepted formats are:

   - telephone numbers (5551111)
   - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

   This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

   Because this variable is located in a table, you can define a different number for each line of the Mediatrix 4102.

3. Enable the automatic call feature by setting the *telephonyAttributesAutomaticCallEnable* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the feature on a per-line basis.

# Call Direction Restriction

You can define in which direction calls are allowed.

In the *Unit Manager Network Administration Manual*, refer to chapter *Telephony Attributes Parameters*, section *Telephony Attributes Configuration Window*.

**To set call direction restriction:**

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIfFeaturesTable* group.
2. Define the restriction on the direction of traffic in the *telephonyAttributesCallDirectionRestriction* variable.

**Table 229:** Call Direction Restrictions

| Restriction | Description |
|---|---|
| noRestriction | Allows incoming and outgoing calls. |
| scnToIpOnly | The Mediatrix 4102 allows to make calls but cannot receive calls. |
| ipToScnOnly | The Mediatrix 4102 allows to receive calls but does not allow to make calls. |

Because this variable is located in a table, you can define a different call direction for each line of the Mediatrix 4102.

# Hook Flash Processing

| **Standards Supported** | • RFC 2976: The SIP INFO Method |
|---|---|

Hook flash processing allows hook flash signals to be transported over the IP network allowing to use advanced telephony services. You can define how to process hook-flash detection. Users normally press the “flash” button of the telephone during a call in progress to put this call on hold, transfer it, or even initiate a conference call.

**Note:** The hook flash processing attribute is not negotiated in SDP.

**To define how to process hook flash:**

1. In the *telephonyAttributesMIB*, set the *telephonyAttributesHookFlashProcessing* variable.

   This allows the enabled subscriber services to be handled by the unit or to be delegated to a remote party.

**Table 230:** Hook Flash Settings

| Setting | Definition |
|---|---|
| processLocally | The hook-flash is processed locally. The actual behaviour of the “flash” button depends on which subscriber services are enabled for this line. See “Chapter 24 - Subscriber Services” on page 333 for more details. |

**Table 230:** Hook Flash Settings (Continued)

| Setting | Definition |
|---|---|
| transmitUsingSignalingProtocol | The hook-flash is processed by a remote party. The hook-flash event is carried by a signaling protocol message. The actual behaviour of the “flash” button depends on the remote party. The hook-flash event is relayed as a SIP INFO message as described in RFC 2976.<br>**Note**: This feature and the DTMF relay feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See “DTMF Transport Type” on page 268 for more details.<br>**Note**: This setting disables all subscriber services that use the “flash” button, such as the Call Hold service. |
| outOfBandUsingRtp | The hook-flash is processed by a remote party. The hook-flash event is relayed as telephone-event 16 via an RFC 2833 RTP packet. The actual behavior of the 'flash' button depends on the remote party. |

# IP Address Call Service

The IP address call service allows a user to dial an IP address without the help of a SIP server. Using this method bypasses any server configuration of your unit.

The user can dial an IP address and enter an optional telephone number. Note that the optional telephone number is matched by using the same digit maps as a normal call.

## Enabling IP Address Calls

▶ **To enable the IP address call service:**

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIpAddressCallCustomization* group.
2. Enable the IP address call service by setting the *telephonyAttributesIpAddressCallEnable* variable to **enable**.

## Dialing an IP Address

▶ **To make an IP address call:**

1. Dial “**” (IP address prefix).
2. Dial the numerical digits of the IP address and use the “*” for the “.” of the IP address.
3. Dial the telephone number of the specific line you want to reach.

   For example, let’s say you want to reach the telephone connected to Line 2 of the Mediatrix 4102 with the IP address 192.168.0.23. The phone number assigned to Line 2 of this Mediatrix 4102 is 1234. You must then dial the following digits:

   ```
   **192*168*0*23#1234
   ```

   In this case, the Mediatrix 4102 sends an INVITE *1234@192.168.0.23*.

# PIN Dialing

| Standards Supported | draft-choudhuri-sip-info-digit-00.txt |
|---|---|

The PIN Dialing feature allows you to configure a PIN (Personal Identification Number) that would be dialed “n” milliseconds after an outgoing call was established.

This feature could be used in the case where a user makes an automatic call to an IVR system, and after a pre-defined delay, the Mediatrix 4102 sends the DTMF tones (PIN) to indicate where the call is coming from.

The PIN is transmitted by using the DTMF out-of-band by signalling protocol transport type. Both parties involved must thus support the *draft-choudhuri-sip-info-digit-00.txt* draft. The PIN must be negotiated in the call. See “DTMF Transport Type” on page 268 for more details on the DTMF out-of-band by signalling protocol transport type.

**To configure the PIN dialing feature:**

1. In the *pinDialingMIB*, define the PIN to dial in the *pinDialingPin* variable.

   The PIN contains the DTMFs to be dialed. The supported digits are “0123456789*#abcdABCD”. Pause characters “,”, “;”, and “p” are also supported and represent 1 second.

**Note:** The *draft-choudhuri-sip-info-digit-00.txt* draft does not support the pause characters “,”, “;”, and “p”. This is a proprietary support.

2. Set the delay prior to sending the PIN in the *pinDialingDelay* variable.

   This value is expressed in milliseconds (ms). The default value is 1000 ms.

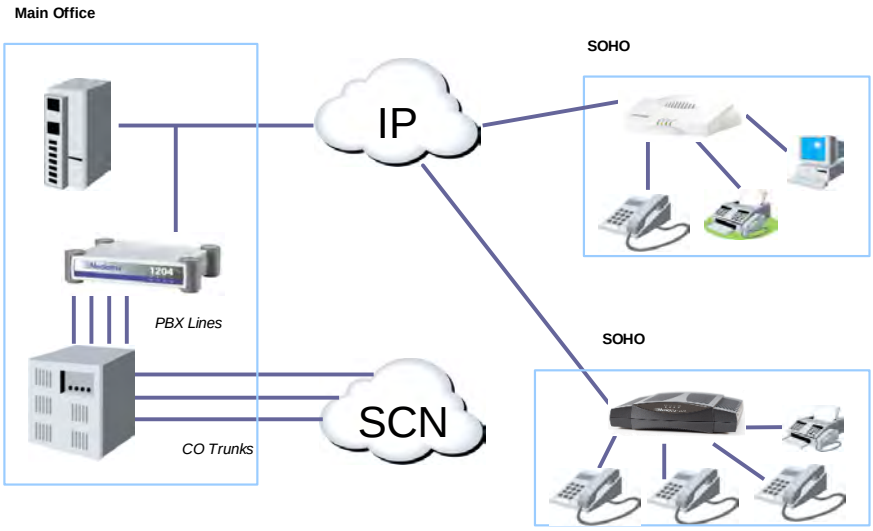
3. Enable the PIN dialing feature by setting the *pinDialingEnable* variable to **enable**.

# Remote Line Extension

The Remote Line Extension feature makes it possible to connect remote small offices with similar capabilities as if they were located in the main or head office. Essential features such as integrated voicemail, unified messaging, line extension dialing plans and many others can be offered to remote sites.

Mediatrix FXS access devices and Mediatrix 1204 FXO gateways extend PBX extensions to remote workers located at SOHOs (Small Offices/Home Offices), using VoIP. PBX extensions are connected via Mediatrix 1204 FXO ports to the IP network, instead of being connected to individual stations. At the SOHO locations, Mediatrix FXS units connect analog phones to the same IP network.

**Figure 103:** Remote Line Extensions




▶ **To configure Remote Line Extensions:**

1. Set the Mediatrix 1204 lines to perform automatic calls to a pre-defined number as defined in "Automatic Call" on page 351.

   This allows you to define a telephone number that is automatically dialed when seizing an FXO line. You can thus redirect SCN calls to a specific IP number such as a telephone connected to a FXS port of a Mediatrix FXS device.

2. If applicable, you can instruct the Mediatrix 1204 to wait until the called party answers the phone before it picks up the SCN line. You can do so in the *fxoMIB* by setting the *fxoWaitForCalleeToAnswerEnable* variable to **enable**.

3. If applicable, define how the Mediatrix 1204 behaves when it receives an IP call in the *analogScnGwDialEnable* variable.

**Table 231:** IP Call Reception Behaviour

| Parameter | Description |
|---|---|
| disable | The Mediatrix 4102 picks up the SCN line and opens the audio channel with the peer. This way, the user of a dedicated FXS/FXO combinaison will feel "closer" to the SCN: as soon as the user takes the receiver off-hook, he/she interacts with the SCN because the actions are not relayed via IP to the FXO unit. |
| enable | The Mediatrix 4102 picks up the SCN line and dials the telephone number associated with the destination of the call before opening the audio channel with the peer. This is the default value. |

4. Define the Hook Flash Processing as per "Hook Flash Processing" on page 352.

   Setting the *telephonyAttributesHookFlashProcessing* variable to **outOfBandUsingRtp** means the following:

**Table 232:** Hook Flash Processing

| Mediatrix 1204 | Mediatrix FXS Device |
|---|---|
| The hook-flash event received via an RFC 2833 RTP packet is executed. | The hook-flash event is relayed as telephone-event 16 via an RFC 2833 RTP packet. |

5. Set the behaviour for the support of RFC 2833 in the *voiceIfDtmfEnforceDefaultEvents* variable for both units.

**Table 233:** DTMF Enforce Default Events

| Parameter | Description |
|---|---|
| enable | Conformance is enforced and support for RFC 2833 implies the support of basic telephony-events. When setting the variable *voiceIfDtmfTransport* to **outOfBandUsingRtp** ("DTMF Transport Type" on page 268), or the variable *telephonyAttributesHookFlashProcessing* to **outOfBandUsingRtp** ("Hook Flash Processing" on page 352), the unit will advertise the support for events 0-15; it will assume support for events 0-15 when support for RFC 2833 is received in an announcement. |
| disable | This creates a deliberate deviance to RFC 2833 as support of basic events is not automatic. The variables *voiceIfDtmfTransport* and *telephonyAttributesHookFlashProcessing* then act independently to specify which events will be relayed via RFC 2833. If Hook Flash relay is enabled by itself, support of event 16 alone will be advertised; if both Hook Flash and DTMF relay are activated, events 0-16 are supported. |

6. If applicable, configure port mapping as per "Source Line Selection" on page 257.

   You could map FXO ports to IP Phones or analog phones connected to FXS ports. This creates transparent and user-friendly call scenarios, as IP endpoints can act as remote PBX extensions. Thus the reach of a PBX can be extended beyond the physical location of the PBX. This is especially an attractive option for SOHO users.

# Delayed Hot Line

The delayed hot line feature is used to make an automatic call to a specified address on the two following conditions:

- When the user picks up the phone but does not dial any digit.
- When the user starts dialing but does not complete a valid number before the timeout set in the *digitMapTimeoutCompletion* variable expires. This is the delayed hotline extension feature.

This feature thus places an automatic call whenever the *digitMapTimeoutFirstDigit* timout expires. It could be used as an alternative to the emergency number (for instance, the 911 number in North America).

**To configure the basic delayed hot line feature:**

1. Enable the delayed hot line feature by setting the *telephonyAttributesDelayedHotLineEnable* variable to **enable**.

   When the feature is disabled, a user picking up the phone but not pressing any telephone keys hears the Receiver Off-Hook tone after the amount of time specified in the *digitMapTimeoutFirstDigit* variable.

▶ **To configure the delayed hot line extension feature:**

1. In the *telephonyAttributesMIB*, set the destination (address or telephone number) that is automatically called in the *telephonyAttributesDelayedHotLineTargetAddress* variable.
2. Enable or disable the Delayed Hot Line extension feature in the *telephonyAttributesDelayedHotLineExtension* variable.

   With this extension, the automatic call mentioned in the basic feature is placed upon expiration of the *digitMapTimeoutCompletion* timer.

**Table 234:** Delayed Hot Line Extension Parameters

| Parameter | Description |
|---|---|
| enable | The destination specified in Step 2 is called upon expiration of the timeout. |
| disable | A user beginning to dial a telephone number but failing to complete the operation before expiration of the *digitMapTimeoutCompletion* timeout hears the Receiver Off-Hook tone. |

# Call Rejection

An incoming call can be rejected if it rings without being answered for a certain amount of time. The default value, 0, indicates that an incoming ringing call is never rejected by the unit.

▶ **To configure the call rejection feature:**

1. In the *telephonyAttributesMIB*, set the number of seconds a port is allowed to ring before automatically rejecting the call in the *telephonyAttributesAutomaticRejection* variable.

   A value of 0 means that there is no limit so the port rings until the caller hangs up.
2. In the *sipInteropMIB*, define the SIP code to use when rejecting a call once the timer specified in *telephonyAttributesAutomaticRejection* elapses in the *sipInteropAutomaticRejectionCode* variable.

   Examples of possible SIP code:

   400: Bad Request
   401: Unauthorized
   402: Payment required
   403: Forbidden
   404: Not found
   405: Method not allowed
   406: Not acceptable
   407: Proxy authentication required
   408: Request timeout
   410: Gone
   413: Request Entity too long
   414: Request-URI too long
   415: Unsupported media type
   416: Unsupported URI Scheme
   420: Bad extension
   421: Extension Required
   423: Interval Too Brief
   480: Temporarily unavailable
   481: Call/Transaction Does not Exist
   482: Loop Detected
   483: Too many hops
   484: Address incomplete
   485: Ambiguous
   486: Busy here
   500: Server internal error
   501: Not implemented
   502: Bad gateway
   503: Service unavailable
   504: Server time-out
   504: Version Not Supported
   513: Message Too Large
   600: Busy everywhere
   603: Decline
   604: Does not exist anywhere

CHAPTER 26

# Message Waiting Indicator

This chapter explains how to set the Mediatrix 4102 to use the Message Waiting Indicator service.

You can also set these parameters via the web interface, as described in "Message Waiting Indicator" on page 125.

## What is Message Waiting Indicator (MWI)?

The Message Waiting Indicator (MWI) service alerts the user when new messages have been recorded on a voice mailbox.

When the user receives a call and does not answer, the notification mechanism detects this situation and starts the auto attendant. The caller can then leave a message.

After the message is recorded, the server sends a message to the Mediatrix 4102 listing how many new and old messages are available. The Mediatrix 4102 alerts the user of the new message in two different ways:

- The telephone's LED blinks (if present).
- A message waiting stutter dial tone replaces the normal dial tone when the user picks up the first line.

**Note:** The message waiting state does not affect the Second Line feature. When in an active call, performing a flash-hook to get access to the second line plays the usual dial tone.

## Standard MWI Methods

The Mediatrix 4102 supports two MWI methods.

### MWI Method #1

| | |
|---|---|
| **Standards Supported** | • draft-ietf-sipping-mwi-01.txt (MWI draft)<br>• "Telecordia GR-1401-CORE (Issue 1, June 2000)" specification (visual message indication (LED blinking)<br>• "GR-506-CORE (Issue 1, with Revision 1, November 1996)" specification (message waiting indicator tone) |

The Mediatrix 4102 sends SUBSCRIBE requests to the server for each line, unless there is no subscription address defined. The Mediatrix 4102 then waits for NOTIFY requests containing the relevant message waiting information.

**To configure the MWI:**

1. In the *mwiMIB*, set the notification mechanism server address to which the Mediatrix 4102 subscribes in the *mwiConfigUserSubscriptionAddress* variable.

   This mechanism notifies the Mediatrix 4102 when new messages are available. The address is a SIP URL such as "scheme:user@host". For instance, "sip:user@foo.com".

   Because this variable is located in a table, you can define a different address for each line of the Mediatrix 4102.

2. Define the digits that users must dial to retrieve messages in the *mwiFetchDigitMap* variable.

   Dialing these digits initiates a call to the voice messaging system. For instance, you could decide to put “*50” as the sequence a user must dial to retrieve voice messages. This sequence must be unique and follow the syntax for digit maps (see “Chapter 22 - Digit Maps” on page 321). Dialing this digit map does not have any effect unless the service's status is “enabled”.

   The activating sequence is set for all the lines of the Mediatrix 4102. You cannot have different sequences for each line.

3. Set the destination to call to retrieve messages in the *mwiConfigFetchAddress* variable.

   The user typically initiates a call to the voice messaging system, and then uses an auto-attendant to get the messages. Available formats are:

   - telephone numbers (5551111)
   - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.

   This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

   Because this variable is located in a table, you can define a different destination for each line of the Mediatrix 4102.

4. Define the duration, in seconds, of dynamic subscription to a messaging service in the *mwiExpirationTime* variable.

5. In the *sipInteropMIB*, define how Message Waiting Indicator notifications must be validated in the *sipInteropMwiMessageSummaryValidation* variable.

   **Table 235:** Message Waiting Indicator Notifications

| Parameter | Description |
|---|---|
| enable | In order to have the Message Waiting Indicator activated, the SIP notification must meet all of the following requirements:<br>*a.* 'Messages-Waiting' must be set to yes.<br>*b.* The message waiting media type must contain 'voice-message'.<br>*c.* The number of new messages must be greater than or equal to 1. |
| disable | In order to have the Message Waiting Indicator activated, the SIP notification must meet he following requirement: 'Messages-Waiting' must be set to yes. |

   Note that only Message Waiting notifications for an established subscription are affected. Message Waiting notifications without subscription always behave as described in *disable*.

6. Enable the MWI by setting the *mwiConfigActivation* variable to **enable**.

   Because this variable is located in a table, you can enable/disable the service on a per-line basis.

**Note:** The MWI subscription refresh is not supported when the caller ID is DTMF-based, so modifiying the variable *mwiConfigActivation* will have no effect.

**To refresh the MWI subscription:**

1. In the *mwiMIB*, set the *mwiSubscriptionCmdRefresh* variable.

   Available values are:

   - noOp: No operation.
   - refresh: Refresh message waiting subscriptions. All enabled endpoints unsubscribe themselves from the service and re-subscribe by using the current provisioning.

### MWI Method #2

| | |
|---|---|
| **Standards Supported** | draft-mahy-sip-message-waiting-02.txt (expired) with proprietary modifications |

This method does not require any special settings or configuration.

# MWI Notify Service

The Mediatrix 4102 offers the possibility to extend some key features to remote extensions located in Branch or Home Offices across the SCN.

This service is available only when using the IP Communication Server v3.1 product as a SIP Redirect server.

For instance, a designated analog voice mail system at a main site can provide voice mail for the home or branch office. The home office user is notified of the message waiting via a message waiting LED on the telephone or a special tone when picking up the telephone.

## How does the Service Work?

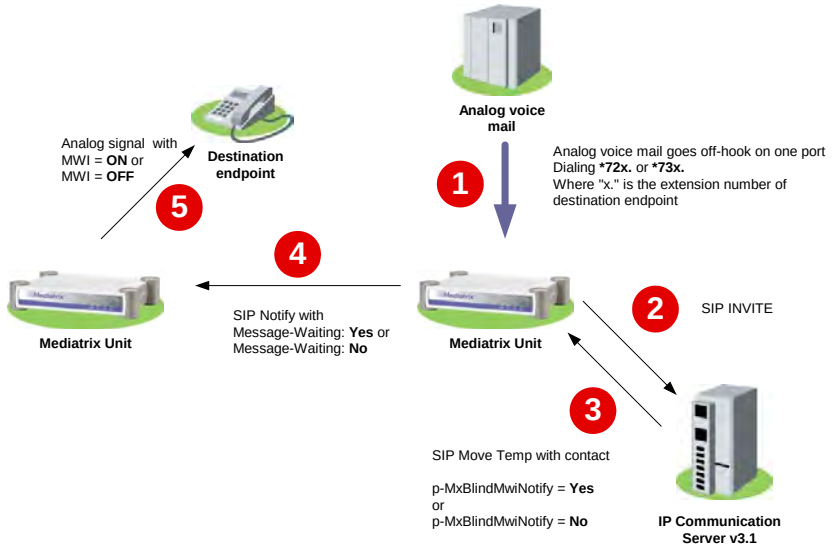
The MWI Notify service is a proprietary feature. In this solution, the analog voice mail system is configured to seize a designated outgoing line and dial a pre-defined string such as “*72xxx” to notify the server it must give a message waiting indication to extension “xxx”. Once voice messages have been retrieved, the analog voice mail system seizes the designated outgoing line and dials a pre-defined string such as “*73xxx” to notify the server to turn off the message waiting indicator for extension “xxx”.

The service uses the Route Manager currently available in the IP Communication Server v3.1 to send a special command to the Mediatrix unit.

The following is the basic sequence of operations for the MWI Notify service:

1. The analog voice mail system dials the following digits:
   `*72101`
   where `*72` is a prefix and `101` the user extension.
2. The Mediatrix unit sends a standard INVITE to the IP Communication Server v3.1 containing the complete dialed string (`*72101`).
   *a.* The IP Communication Server looks for the registered user “*72101” in the Registrar database.
   *b.* The IP Communication Server cannot find the user, so it asks the Route Manager to process the request.
   *c.* Provided that the Route Manager is properly configured, it recognizes the “*72” prefix and associates it to the proper route conditions.
3. The IP Communication Server answers the request with a “Moved Temporarily”. It contains information about the target(s) in the *Contact* header plus a proprietary *p-MxBlindMWINotify=yes/no* field.
   *a.* The Mediatrix unit retrieves the location from the IP Communication Server’s answer and the *p-MxBlindMWINotify* field.
   *b.* The Mediatrix unit parses the answer from the IP Communication Server and recognizes *p-MxBlindMWINotify* as a special command.
4. The Mediatrix unit sends a NOTIFY to the location received from the IP Communication Server by using the proper yes or no value (*72 = yes, *73 = no) specified by the route condition.
5. The unit receiving the NOTIFY enables or disables the MWI service for the specified port/user.

**Figure 104:** Example of the MWI Notify Service




## Configuring the IP Communication Server

In the Route Manager of the IP Communication Server, you must configure routes that would be triggered by a pre-defined prefix. The prefix could be any valid digits (DTMF). The example described above uses “*72” to enable the MWI and “*73” to disable the MWI.

For more information on how to configure the Route Manager, please refer to the *IP Communication Server Administration Manual* or the IP Communication Server contextual help.

## Configuring the Mediatrix 4102

There is no special unit configuration required. The Mediatrix unit behaves as if in a standard call until it receives one of the following parameters in the *Contact* field:

- p-MxBlindMwiNotify=Yes

  or
- p-MxBlindMwiNotify=No

Upon receiving one of these parameters, the unit sends a NOTIFY to the destination endpoint instead of an INVITE. The sent NOTIFY is compliant with <draft-mahy-sip-message-waiting-02.txt>.

CHAPTER 27

# Management Server Configuration

The Management Server is a generic name for a module or software that is used to remotely set up Mediatrix 4102 units. For instance, the Management Server could be the Media5's Unit Manager Network product. See "Unit Manager Network – Element Management System" on page xxiv for more details.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Unit Manager Server*.

## Using the Management Server

You have the choice of setting up Mediatrix 4102 units directly with a SNMP browser or with the Management Server. If you want to use the Management Server to setup the units, you shall tell these units how to reach the Management Server.

▶ **To use the Management Server:**

1. In the *msMIB*, locate the *msEnable* variable.

   This variable enables the Management Server to remotely manage the Mediatrix 4102.

2. Set the *msEnable* variable to **enable**.

3. Set the Trap retransmission period (*msTrapRetransmissionPeriod* variable) to the desired value.

   The available values range from 10 ms to 604 800 000 ms (1 week). The default value is 60 000 ms.

4. Set the Trap retransmission retry count (*msTrapRetransmissionRetryCount* variable) to the desired value.

   When the retry count is elapsed, the Mediatrix 4102 stops the provisioning sequence. The default value is 10. If this variable is set to -1, then the provisioning sequence never stops. The trap is sent until the Management Server replies.

### Configuration Source

The Mediatrix 4102 must know the IP address and port number of the Management Server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

#### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

▶ **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *msSelectConfig Source* variable.

   This variable defines whether the Mediatrix 4102 shall get its Management Server configuration through a DHCP server or not.

2. Set the *msSelectConfigSource* variable to **dhcp**.

   You can query the Management Server's IP address and port number assigned by the DHCP server in the *msHost* and *msTrapPort* read-only variables (in the *ipAddressStatus* folder).

3. Set how you want to define the Management Server information in the DHCP server:

**Table 236:** Management Server DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *msDhcpSiteSpecificCode* variable to **0**. Set the management server IP address in the DHCP server inside the vendor specific sub-option 200 (hexadecimal 0xC8). |
| site specific code | The *msDhcpSiteSpecificCode* variable to any value between 128 and 254. Set the management server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *msDhcpSiteSpecificCode* variable in the unit's configuration). |

See "Vendor and Site Specific DHCP Options" on page 169 for more details.

## Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *msSelectConfig Source* variable.

   This variable defines whether the Mediatrix 4102 shall get its Management Server configuration through a DHCP server or not.

2. Set the *msSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 237:** Management Server Static Address

| Variable | Description |
|---|---|
| msStaticHost | Static management server IP address or domain name.<br>**Default Value**: 192.168.0.10 |
| msStaticTrapPort | Static management server IP port number. Restart the unit to update this parameter.<br>**Default Value**: 162<br>**Note**: Change the port used in the management server. Not doing so will prevent you from viewing the received traps from the unit.<br>The management server could be a product such as the Unit Manager Network. |

CHAPTER 28

# Quality of Service (QoS)

QoS (Quality of Service) features enable network managers to decide on packet priority queuing. The Mediatrix 4102 supports the Differentiated Services (DS) field and 802.1q taggings. There are three variables – one variable for signalling (SIP) and one variable for each of voice and T.38 media.

The Mediatrix 4102 supports the Real Time Control Protocol (RTCP), which is used to send packets to convey feedback on quality of data delivery.

The Mediatrix 4102 does not support RSVP (Resource Reservation Protocol).

## Differentiated Services (DS) Field

| Standards Supported | RFC 2475 – An Architecture for Differentiated Services |
|---|---|

Differentiated Services (DiffServ, or DS) is a protocol for specifying and controlling network traffic by class so that certain types of traffic – for example, voice traffic, which requires a relatively uninterrupted flow of data, might get precedence over other kinds of traffic.

DiffServ replaces the first bits in the ToS byte with a differentiated services code point (DSCP). It uses the existing IPv4 Type of Service octet.

It is the network administrator's responsibility to provision the Mediatrix 4102 with standard and correct values.

**Note:** If you are using the Mediatrix 4102 in router mode, you may want to differentiate the packets sent by the PC from the packets sent by the Mediatrix 4102. In this case, you must use a substitution value, as described in "Configuring TAS" on page 205.

You can also set these parameters via the web interface, as described in "DiffServ Configuration" on page 131.

### What are Differentiated Services?

Differentiated Services avoids simple priority tagging and depends on more complex policy or rule statements to determine how to forward a given network packet. An analogy is made to travel services, in which a person can choose among different modes of travel – train, bus, airplane – degree of comfort, the number of stops on the route, standby status, the time of day or period of year for the trip, and so forth.

For a given set of packet travel rules, a packet is given one of 64 possible forwarding behaviors – known as per hop behaviors (PHBs). A six-bit field, known as the Differentiated Services Code Point (DSCP), in the Internet Protocol header specifies the per hop behavior for a given flow of packets. The DS field structure is presented below:

```
0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
| DSCP                  |  CU   |
+---+---+---+---+---+---+---+---+
MSB                           LSB
```

- *DSCP*: Differentiated Services CodePoint.
- *CU*: Currently Unused. The CU bits should always be set to 0.

For both signalling and media packets, the DSCP field is configurable independently. The entire DS field (TOS byte) is currently configurable.

▶ **To enable the DS field configuration:**

1. In the *qosDiffServ* group of the *qosMIB*, locate the following variables:
   - qosSignalingDiffServ
   - qosVoiceDiffServ
   - qosT38FaxDiffServ

   These variables are 1 octet scalar ranging from 0 to 255. The DSCP default value should be 101110. This results in the DS field value of 10111000 (184).

   This default value would result in a value of “101” precedence bits, low delay, high throughput, and normal reliability in the legacy IP networks (RFC 791, RFC 1812). Network managers of legacy IP networks could use the above-mentioned values to define filters on their routers to take advantage of priority queuing. The default value is based on the Expedited Forwarding PHB (RFC 2598) recommendation.

**Note:** RFC 3168 now defines the state in which to set the two least significant bits in the TOS byte. On the other hand, this RFC only applies to TCP transmissions and the bits are thus set to “0” in the Mediatrix 4102. This has the following effects:

- The TOS values for UDP packets are the same as in the MIB.
- The TOS values for TCP packets are equal to the closest multiple of 4 value that is not greater than the value in the MIB.

2. Set the value you want to use.

   You can find references on DS field under the IETF working group DiffServ. For more information, please refer to the following RFC documents:
   - Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers (RFC 2474)
   - An Architecture for Differentiated Services (RFC 2475)
   - Assured Forwarding PHB Group (RFC 2597)
   - An Expedited Forwarding PHB (RFC 2598)

# IEEE 802.1q

The 802.1q standard recommends the use of the 802.1q VLAN tags for Ethernet frames traffic prioritization. VLAN tags are 4-byte headers in which three bits are reserved for priority indication. The values of the priority bits shall be provisioned.

The 802.1q standard comprises the 802.1p standard.

It is the network administrator’s responsibility to provision the Mediatrix 4102 with standard and correct values.

You can also set these parameters via the web interface, as described in “802.1q Configuration” on page 129.

▶ **To enable the IEEE 802.1q user priority configuration:**

1. In the *qosIeee8021q* group of the *qosMIB*, locate the following variables:
   - qosSignalingIeee8021qEnable
   - qosVoiceIeee8021qEnable
   - qosT38FaxIeee8021qEnable
2. Set the value of these variables to **enable**.

   The corresponding user priority configuration is enabled.
3. In the *qosIeee8021q* group of the *qosMIB*, locate the following variables:
   - qosSignalingIeee8021qUserPriority
   - qosVoiceIeee8021qUserPriority
   - qosT38FaxIeee8021qUserPriority

These variables are 1 octet scalar ranging from 0 to 7. The 802.1q default priority value should be 6 for both signalling and media packets.

4. Set the value you want to use.

For more information, please refer to the *MIB Reference* manual.

# Voice QoS vs RTCP Packets

You can define whether or not the configuration related to the voice QoS (*qosVoiceDiffServ* and *qosVoiceIeee8021qUserPriority* variables) is also applied to the RTCP packets generated by the device.

▶ **To define the voice QoS configuration behavior :**

1. In the *qosInterop* group of the *qosMIB*, locate the *qosInteropUseVoiceQoSForRtcpEnable* variable.
2. Set the value of this variable according to your needs.

**Table 238:** Voice QoS Behavior

| Status | Description |
|---|---|
| enable | The voice QoS configuration (*qosVoiceDiffServ* and *qosVoiceIeee8021qUserPriority* variables) is also applied to the RTCP packets. |
| disable | The RTCP packets are not tagged by the *qosVoiceDiffServ* and *qosVoiceIeee8021qUserPriority* variables. |

# VLAN

You can set various VLAN parameters to control user priority.

You can also set these parameters via the web interface, as described in "802.1q Configuration" on page 129.

**To enable the VLAN configuration:**

1. In the *qosVlanIeee8021q* group of the *qosMIB*, locate the *qosVlanIeee8021qTaggingEnable* variable.
2. Set the value of this parameter to **enable**.

   The VLAN configuration is enabled.
3. Locate the following variables:
   - qosVlanIeee8021qVirtualLanID
   - qosVlanIeee8021qDefaultUserPriority

   When both VLAN tagging and VLAN Substitution are enabled and their VLAN ID is the same, VLAN Tagging has precedence over VLAN Substitution. If VLAN Substitution has the same ID as VLAN Tagging, VLAN Substitution is not enabled and the Mediatrix 4102 behaves as such. You should change the ID of one of the features to enable VLAN Substitution. See "VLAN Substitution" on page 369 for more details.
4. Set the value of these variables.
5. Restart the Mediatrix 4102 so that the changes may take effect.

For more information, please refer to the *MIB Reference* manual.

## VLANs

VLANs are created with standard Layer 2 Ethernet. A VLAN Identifier (VID) is associated with each VLAN. VLANs offer the following benefits:

- VLANs are supported over all IEEE 802 LAN MAC protocols, and over shared media LANs as well as point-to-point LANs.
- VLANs facilitate easy administration of logical groups of stations that can communicate as if they were on the same LAN. They also facilitate easier administration of moves, adds, and changes in members of these groups.
- Traffic between VLANs is restricted. Bridges forward unicast, multicast, and broadcast traffic only on LAN segments that serve the VLAN to which the traffic belongs.

The VLAN field in the Ethernet file is located after both destination and source addresses:

```
    0                   1
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7   (byte)
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Dest Addr | Src Addr  | VLAN  | Type/Length | ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The VLAN field is separated as follows:

```
   0  (bit)           1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |             0x8100            | Pri |T|          VID          |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

For both signalling and media packets, the VLAN priority section is configurable independently.

## VLAN Substitution

The Mediatrix 4102 can tag the packets relayed from the *LAN* port to the *WAN* port with a VLAN ID different from the standard value defined in "VLAN" on page 368.

**Note:** This feature only works when TAS is disabled. See "Chapter 13 - Transparent Address Sharing" on page 199 for more details.

In this case, the packets coming from the LAN (usually a router or a PC) are tagged with a substitution VLAN ID before sending the packets to the WAN. In the opposite direction, the Mediatrix 4102 removes the VLAN tags and then send the packets to the LAN. The packets generated by the Mediatrix 4102 for VoIP/Signaling/Management can also be tagged with a different VLAN ID.

This can be used to prioritize VoIP in a network.

▶ **To configure the VLAN substitution:**

1. In the *qosIeee8021qSubstitution* group of the *qosMIB*, define the substitution IEEE 802.1Q Virtual LAN ID in the *qosVlanIeee8021qSubstitutionVlanID* variable.

   The value 1 is the default Port VID (PVID) for bridge port. The 4095 VID (0xFFF) is reserved for implementation use and it must not be used in the tag header.

   You can use a VID of 0.

   As per the standard, some bridges may not support the full range of VID.

   When both VLAN tagging and VLAN Substitution are enabled and their VLAN ID is the same, VLAN Tagging has precedence over VLAN Substitution. If VLAN Substitution has the same ID as VLAN Tagging, VLAN Substitution is not enabled and the Mediatrix 4102 behaves as such. You should change the ID of one of the features to enable VLAN Substitution. See "VLAN" on page 368 for more details.

2. Set the substitution IEEE 802.1Q Virtual LAN default user priority in the *qosVlanIeee8021qSubstitutionUserPriority* variable.

   This value applies to all protocols for which no priority filtering is enabled (e.g. ARP, ICMP).

   - 7 = High priority
   - 0 = Low priority

3. Enable the VLAN substitution by setting the *qosVlanIeee8021qSubstitutionEnable* variable to **enable**.

   The QoS 802.1q fields of the packets sent from the PC to the WAN are assigned the value defined in the *qosVlanIeee8021qSubstitutionVlanID* variable.

4. Restart the Mediatrix 4102 so that the changes may take effect.

### Ethernet Frames Issue

There is currently an issue with the CPU of the Mediatrix 4102. This issue prevents Ethernet frames, which have a 802.1q tag, to be correctly forwarded through the *LAN* port of the Mediatrix 4102, if these frames have less than 68 bytes (including FCS).

For example, it is valid to have an Ethernet frame of 64 bytes, even if it includes a 802.1q tag. However, when the CPU switch forwards this packet through the *LAN* port, it removes the 802.1q tag, but does not add any padding byte. The Ethernet frame is output with only 60 bytes, which is invalid and dropped by most equipment.

A workaround would be to modify the behaviour of your router to generate 802.1q frames with at least 68 bytes.

## VLAN ID Filtering

VLAN ID filtering may be applied when VLAN substitution is enabled. This feature filters the packets and prevents the *Computer* connector from receiving untagged packets or packets with an invalid tag. It is especially useful to restrain the broadcast domain between each connector's subnet. The following table describes the VLAN ID filtering behaviour:

**Table 239:** VLAN ID Filtering

| Status | Description |
|---|---|
| Enabled | Packets sent to the *Computer* connector must have the proper VLAN ID. Packets with a different VLAN ID, or untagged packets, are dropped. |
| Disabled | Packets sent to the *Computer* connector (tagged with any tag or untagged) are forwarded from the *Network* connector to the *Computer* connector. This effectively extends the broadcast domain and allows nodes behind the *Computer* connector to share the same IP subnet as the unit itself. |

▶ **To enable VLAN ID Filtering:**

1. In the *qosIeee8021qSubstitution* group of the *qosMIB*, set the *qosVlanIeee8021qSubstitutionEnable* variable to **enable**.

   See "VLAN Substitution" on page 369 for more details.

2. Enable VLAN ID filtering by setting the *qosVlanIeee8021qSubstitutionFiltering* variable to **enable**.

3. Restart the Mediatrix 4102 so that the changes may take effect.

## LAN and WAN with VLAN substitution

LAN and WAN interfaces with VLAN substitution is used to forward network traffic between the WAN and the LAN interfaces of the Mediatrix 4102. It allows devices on the LAN side to communicate with the WAN side.

The *LAN* connector of the Mediatrix 4102 has an IP address and *WAN* connector could also have one. You can configure the IP address of the *LAN* connector statically. The PC connected to the Mediatrix 4102 could then use this address to contact the Mediatrix 4102.

For more information on how to set the *LAN* connector IP address of the Mediatrix 4102, refer to "LAN Connector Static IP Address" on page 166.

### Switch Mechanism with LAN and WAN with VLAN Substitution

The following table describes the various behaviours of the switch mechanism.

**Table 240:** Seitch Mechanism Description

| Behaviour | Description |
|---|---|
| Switch of outbound packets | The Mediatrix 4102 directly sends outbound packets either to the WAN or to the PC, depending on the destination MAC address. |
| Switch of inbound packets from the LAN | If the destination MAC address is the Mediatrix 4102's WAN MAC address, the packet is processed locally; otherwise, it is encapsulated into a VLAN 802.1q packet and sent to the WAN interface. |
| Switch of inbound packets from the WAN | If the destination MAC address is the Mediatrix 4102's WAN MAC address, the packet is processed locally; otherwise, if it is a VLAN encapsulated packet, the Mediatrix 4102 removes the encapsulation and sends it to the LAN interface. |

CHAPTER 29

# Syslog Daemon

This chapter describes how to configure and use the Syslog daemon.

## Syslog Daemon Configuration

| Standards Supported | RFC 3164 – The BSD Syslog Protocol |
|---|---|

The Syslog daemon is a general purpose utility for monitoring applications and network devices with the TCP/IP protocol. With this software, you can monitor useful messages coming from the Mediatrix 4102 unit. If no Syslog daemon address is provided by a DHCP server or specified by the administrator, no messages are sent.

For instance, if you want to download a new software into the Mediatrix 4102, you can monitor each step of the software download phase. Furthermore, if the unit encounters an abnormal behaviour, you may see accurate messages that will help you troubleshoot the problem.

In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Syslog Daemon*.

▶ **To enable the Syslog daemon:**

1. In the *syslogMIB*, locate the *syslogMsgMaxSeverity* variable.

   This variable indicates which syslog message is processed. Any syslog message with a severity value greater than the selected value is ignored by the agent.

   - disabled
   - critical
   - error
   - warning
   - informational
   - debug

   A higher level mask includes lower level masks, e.g., *Warning* includes *Error* and *Critical*. The default value is **informational**.

   The following are some of the messages the unit sends:

**Table 241:** Syslog Messages Examples

| Event | Level | Message |
|---|---|---|
| The configuration update with the specific configuration file has been successful (configuration file fetching) | Informational | `The specific configuration update succeeded.` |
| The configuration update with the specific configuration file experienced an error and has not been completed (configuration file fetching) | Error | `The specific configuration update failed.` |
| The software update has been successful | Informational | `The software update succeeded.` |
| The software update experienced an error and has not been completed | Error | `The software update failed.` |

## Configuration Source

The Mediatrix 4102 must know the IP address and port number of the Syslog server. You can assign these information to the Mediatrix 4102 through a DHCP server or manually enter them yourself with the static variables.

You can also set these parameters via the web interface, as described in "Syslog Monitoring" on page 32.

### DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See "Chapter 9 - IP Address and Network Configuration" on page 155 for more details.

▶ **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfig Source* variable.

   This variable defines whether the Mediatrix 4102 shall ask for its Syslog daemon settings through a DHCP server or not.

2. Set the *syslogSelectConfigSource* variable to **dhcp**.

   You can query the Syslog daemon's IP address and port number assigned by the DHCP server in the *syslogHost* and *syslogPort* read-only variables (under the *ipAddressStatus Syslog* group of the *ipAddressStatus* folder).

3. Set how you want to define the Syslog information in the DHCP server:

**Table 242:** Syslog DHCP Information

| To use a... | Set... |
|---|---|
| vendor specific code | The *syslogDhcpSiteSpecificCode* variable (under the *ipAddressConfigSyslogDhcp* group) to **0**. Set the Syslog server IP address in the DHCP server inside the vendor specific sub-option 110 (hexadecimal 0x6E). |
| site specific code | The *syslogDhcpSiteSpecificCode* variable (under the *ipAddressConfigSyslogDhcp* group) to any value between 128 and 254. Set the Syslog server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the *syslogDhcpSiteSpecific Code* variable in the unit's configuration). |

See "Vendor and Site Specific DHCP Options" on page 169 for more details.

### Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

▶ **To use static information:**

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfig Source* variable.

   This variable defines whether the Mediatrix 4102 shall ask for its Syslog daemon settings through a DHCP server or not.

2. Set the *syslogSelectConfigSource* variable to **static**.

3. Set the following variables:

**Table 243:** Syslog Daemon Static Address

| Variable | Description |
|---|---|
| syslogStaticHost | Syslog server static IP address or domain name.<br>**Default Value**: 192.168.0.10 |

**Table 243:** Syslog Daemon Static Address (Continued)

| Variable | Description |
|---|---|
| syslogStaticPort | Syslog server static IP port number.<br>**Default Value**: 514 |

## Customizing Syslog Messages

You can display additional information in the prefix of syslog messages the Mediatrix 4102 sends. This allows you to later filter the messages. The following is the additional information you can enable:

- MAC address
- local time
- local host

**Note:** This applies only to syslog messages sent on the network and not the local syslog messages.

**To add the MAC address of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayMacAddress* variable to **enable**.
   The MAC address of the Mediatrix 4102 is part of the prefix for all syslog messages.
   If you set the variable to **disable**, the MAC address is not displayed in the prefix of the syslog messages.

**To add the local time of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayTime* variable to **enable**.
   The current local time of the Mediatrix 4102 is part of the prefix for all syslog messages.
   If you set the variable to **disable**, the time is not displayed in the prefix of the syslog messages.

**To add the local host of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayLocalHost* variable to **enable**.
   The current local host of the Mediatrix 4102 is part of the prefix for all syslog messages.
   If you set the variable to **disable**, the local host is not displayed in the prefix of the syslog messages.

## Configuring the Syslog Daemon Application

You shall configure the Syslog daemon to capture those messages. Refer to your Syslog daemon's documentation to learn how to properly configure it to capture messages.

# Local Syslog

The local syslog is an internal syslog server to the Mediatrix 4102. It keeps the last *n* syslog messages. These syslog messages are displayed in the *System log* page of the web interface (see "Chapter 2 - Web Interface – Introduction" on page 25 for more details).

**To set the local syslog:**

1. In the *syslogMIB*, locate the *syslogMsgLocalMaxSeverity* variable.

   This variable indicates which syslog message is processed by the Mediatrix 4102. Any syslog message with a severity value greater than the selected value is ignored.

   - disabled
   - critical
   - error
   - warning
   - informational
   - debug

   A higher level mask includes lower level masks, e.g., *Warning* includes *Error* and *Critical*. The default value is **informational**.

   The following are some of the messages the unit sends:

   **Table 244:** Syslog Messages Examples

| Event | Level | Message |
|---|---|---|
| The configuration update has been successful (configuration file fetching) | Informational | `The specific configuration update succeeded.` |
| The configuration update experienced an error and has not been completed (configuration file fetching) | Error | `The specific configuration update failed.` |
| The software update has been successful | Informational | `The software update succeeded.` |
| The software update experienced an error and has not been completed | Error | `The software update failed.` |

2. Set the maximal number of syslog messages the Mediatrix 4102 handles in the *syslogMsgLocalMaxNbr* variable.

   Modifying this value resets the messages.

   If the Mediatrix 4102 sends a new syslog message and the maximum number of messages is reached, the oldest one is removed.

   You can view the syslog messages in the following locations:

   - In the *syslogLocalMsgTable* of the *syslogMIB.*
   - In the *System log* page of the web interface.

CHAPTER 30

# Statistics

The Mediatrix 4102 collects meaningful statistics that can be read via the RTP MIB.

## RTP Statistics

RTP statistics are related to the transmission of information and include, but are not limited to:

- Number of octets transmitted/received
- Number of packets transmitted/received
- Number of lost packets
- Percentage of lost packets
- Minimum, maximum and average Jitter interarrival time (time, in milliseconds, between the arrival of packets)
- Minimum, maximum and average latency time

These statistics are located under the *rtpStats* group of the *rtpMIB*. See the *MIB Reference* manual for more details.

### Statistics Buffers

Each statistics has three different buffers in which they are collected:

**Table 245:** Statistics Buffers

| Statistic | Description |
|---|---|
| Last connection | These are the statistics of the last completed connection. |
| Current | These are the statistics of the current connection. If using the Cumulated buffer, they are added to the cumulated statistics buffer and then reset. |
| Cumulated | These are the cumulated statistics of all the connections. Define a period of time and maximum number of periods you want to keep. For instance, you could define to keep the statistics for the last 24 periods of 1 hour. |

### How are Statistics Collected?

When collecting statistics, you can do so in two ways:

- Continuous collection of statistics.

  In this case, the cumulated statistics are not used (disabled) and the current statistics are constantly updated.
- Collection of statistics for a defined period of time with a user-defined accuracy.

  For instance, you could define to keep the statistics for the last 24 periods of 1 hour.

**To set statistics collection:**

1. In the *sysConfigMIB*, locate the *sysConfigStats* group.
2. Set the period length you want to keep in the *sysConfigStatsPeriodLength* variable.

   The length of a period may vary from 5 minutes to 24 hours, by 5-minutes sections. At expiration, the current statistics are added to the cumulated statistics buffer and then reset. Note that modifying the value of this variable resets statistics to 0.

3. Set the maximum number of periods to cumulate in the *sysConfigStatsNumberPeriods* variable.

   The maximum number of periods cumulated is 24. If this variable is set to 0, statistics are collected indefinitely in the current variables. Note that modifying the value of this variable resets statistics to 0.

**To reset statistics:**

1. In the *sysAdminMIB*, set the *sysAdminCommand* variable to **resetStats**.

   This resets all cumulated call statistics.

## Statistics by Syslog

You can configure the Mediatrix 4102 to send the RTP and T.38 statistics by syslog message. You will thus be able to see them by using your syslog daemon.

- The RTP statistics are sent at the end of a call.
- The T.38 statistics are sent at the end of a fax.

The syslog message level is “informational” and uses the module name “Statistics”. Table 246 lists the different statistics fields to send.

**Table 246:** Statistics by Syslog

| Short Name | Description | Corresponding MIB Variable |
|---|---|---|
| TxByte | Number of octets transmitted. | rtpStatsLastConnNumberOctetsTransmitted |
| RxByte | Number of octets received. | rtpStatsLastConnNumberOctetsReceived |
| TxPkt | Number of packets transmitted. | rtpStatsLastConnNumberPacketsTransmitted |
| RxPkt | Number of packets received. | rtpStatsLastConnNumberPacketsReceived |
| NbrPktLost | Number of packets lost. | rtpStatsLastConnNumberPacketsLost |
| PctPktLost | Percentage of packets lost. | rtpStatsLastConnPercentPacketsLost |
| JitMin | Minimum interarrival time in milliseconds. | rtpStatsLastConnInterarrivalJitterMin |
| JitMax | Maximum interarrival time in milliseconds. | rtpStatsLastConnInterarrivalJitterMax |
| JitAvg | Average interarrival time in milliseconds. | rtpStatsLastConnInterarrivalJitterAvg |
| LatMin | Minimum latency in milliseconds. | rtpStatsLastConnLatencyMin |
| LatMax | Maximum latency in milliseconds. | rtpStatsLastConnLatencyMax |
| LatAvg | Average latency in milliseconds. | rtpStatsLastConnLatencyAvg |

The syslog message sent will have the following format:

```
RTP TxByte:<TxByte>, RxByte:<RxByte>, TxPkt:<TxPkt>, RxPkt:<RxPkt>,
NbrPktLost:<NbrPktLost>, PctPktLost:<PctPktLost>, JitMin:<JitMin>,
JitMax:<JitMax>, JitAvg:<JitAvg>, LatMin:<LatMin>, LatMax:<LatMax>,
LatAvg:<LatAvg>
```

Example with the syslog message prefix:

```
Dec 31 19:15:05 10.2.130.31 Statistics [0073] RTP TxByte:32002, RxByte:24514,
TxPkt:156, RxPkt:140, NbrPktLost:0, PctPktLost:0, JitMin:0, JitMax:6, JitAvg:3,
LatMin:8, LatMax:8, LatAvg:8
```

**To enable to send statistics by syslog:**

1. In the *sysConfigMIB*, set the *sysConfigStatsBySyslogEnable* variable to **enable**.

## Example

The following is an example with *sysConfigStatsNumberPeriods* = 3 and *sysConfigStatsPeriodLength* = 1 (5 minutes).

**Table 247:** Statistics Setting Example

| Statistics | 5-minutes sections | | | | | |
|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** |
| rtpStatsCurrentTotalOctetsTransmitted | 50 | 30 | 60 | 40 | 100 | 50 |
| rtpStatsCumulatedTotalOctetsTransmitted | 0 | 50 | 80 | 140 | 130 | 200 |

1. 50 total octets transmitted in the first 5-minutes period.
2. 30 total octets transmitted in the second 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 50.
3. 60 total octets transmitted in the third 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 80.
4. 40 total octets transmitted in the fourth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 140.
5. 100 total octets transmitted in the fifth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable.

   In the above example, the *rtpStatsCumulatedxx* variables always contain the statistics for the last 15 minutes (*sysConfigStatsNumberPeriods* X *sysConfigStatsPeriodLength*) accurate to 5 minutes (*sysConfigStatsPeriodLength*). This means that the statistics for the first 5-minutes period are dropped, for a cumulated total octets transmitted of 130.
6. 50 total octets transmitted in the sixth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable.

   The statistics for the second 5-minutes period are dropped, for a cumulated total octets transmitted of 200.

CHAPTER 31

# Maximum Transmission Unit (MTU)

This chapter describes the MTU (Maximum Transmission Unit) requirements of the Mediatrix 4102.

## What is MTU?

The *Maximum Transmission Unit* (MTU) is a parameter that determines the largest packet than can be transmitted by an IP interface (without it needing to be broken down into smaller units). Each interface used by TCP/IP may have a different MTU value specified.

The MTU should be larger than or equal to the largest packet you wish to transmit unfragmented. Note that this only prevents fragmentation locally. Some other link in the path may have a smaller MTU: the packet will be fragmented at that point, although some routers may refuse packets larger than their MTU.

## Mediatrix 4102 MTU

The Mediatrix 4102 MTU is 1500 bytes, which is the Ethernet typical value.

## Possible Hardware Problem

The implementation of the IEEE Standard 802.1q in the Mediatrix 4102 may have a minor problem because of hardware limitations.

802.1q increases the Ethernet frame header by 4 bytes, adding a Virtual LAN ID and a user_priority. This is useful to limit broadcasts that cross bridges, and it may also prioritize frames in the queuing algorithm of switches. However, it also increases the maximum possible size of Ethernet frames from 1518 to 1522 bytes, and this might not be handled adequately by every hardware.

A workaround is available for PCs running Windows to avoid sending 1522 bytes packets (note that this happens only in special and rare cases). The workaround is to reduce the MTU of the interface (the one that sends packets with 802.1q framing) by 4 bytes.

1. Use the registry editor (regedt32) and go to the key:

   Windows 2000 and later:

   ```
   HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces
   \<ethernet adapter>
   ```

   Windows NT4 and 98:

   ```
   \HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<ethernet
   adapter>\Parameters\Tcpip
   ```

   where <Ethernet adapter> can be found by using the command “ipconfig /all”.

2. Add (or modify) a value named MTU of type REG_DWORD. Set it to 1496 (instead of 1500), in decimal. Restart the computer to have those changes in effect.

   In Windows 2000 and later this value is under the following key:

   - Key: *Tcpip\Parameters\Interfaces\ID* for Adapter2

- Value Type: REG_DWORD Number
- Valid Range: 68 - the MTU of the underlying network
- Default: 0xFFFFFFFF
- Description: This parameter overrides the default MTU for a network interface. The MTU is the maximum packet size in bytes that the transport will transmit over the underlying network. The size includes the transport header. Note that an IP datagram may span multiple packets. Values larger than the default for the underlying network will result in the transport using the network default MTU. Values smaller than 68 will result in the transport using an MTU of 68.

**3.** To validate that the changes are correct, try to ping the Mediatrix 4102 with large packets once restarted:

```
ping -l 2000
```

This will cause IP fragmentation, the first fragment being as large as the interface allows it. With the MTU reduced, you should now receive an answer. For more informations, see:

http://support.microsoft.com/default.aspx?scid=kb;en-us;120642.

CHAPTER 32

# Troubleshooting

You can experience some problems when connecting the Mediatrix 4102 to the network. The following section examines some of these problems and possible solutions.

A Syslog message lists the problems the Mediatrix 4102 encounters. You can see this message with the Syslog daemon.

This chapter covers the following types of issues:

- General Operation Issues
- Calling Issues
- Fax Issues
- Configuration Issues
- Software Upgrade Issues
- SNMP Management Software Issues

## General Operation Issues

The following are general operation issues you may encounter.

**DESCRIPTION**: Unit does not operate – All LEDs are OFF.

**POSSIBLE CAUSE**: Power is not fed to the unit.

**SOLUTION**: Check that:

- The power cord is connected to the electrical outlet.
- The power cord is fully inserted into the Mediatrix 4102 power socket.

**DESCRIPTION**: There is a long delay when starting the Mediatrix 4102.

**POSSIBLE CAUSE**: If any information is set to come from the DHCP server (for example, SNTP address), the restarting unit waits for a maximum period of two minutes if the DHCP server cannot be reached, even if most other settings are set to “static”.

This delay is caused by the Mediatrix 4102 that cannot function as configured if part of its configuration (the DHCP information) is unavailable.

The two minutes waiting period is an issue with switches that use the Spanning Tree Protocol. When this protocol is enabled, the restarting Mediatrix 4102 may be denied from the network for a certain time (about two minutes). The unit must not ignore transmission errors (i.e., timeouts) because these errors might be caused by the Spanning Tree Protocol.

**SOLUTION**: Media5 recommends to set up all information to use a static value, or have a DHCP server answer the requests. See “Static Configuration” on page 157 for more details.

**DESCRIPTION**: I changed the IP address of my unit, but I can't reach the DHCP server anymore.

**POSSIBLE CAUSE**: A subnet mask is used to determine to which subnet an IP address belongs. An IP address has two components, the network address and the host address. For example, let's consider the IP address 192.168.0.1. Assuming this is part of a Class B network, the first two numbers (192.168) represent the Class B network address, and the second two numbers (0.1) identify a particular host on this network.

Let's say you have the following information:

- Mediatrix 4102 IP address: 192.168.0.1
- Subnet Mask: 255.255.0.0 (Class B)
- DHCP Server IP address: 192.168.0.20

If you happen to change the Mediatrix 4102 IP address to 192.169.0.1, for instance, the subnet mask is still valid, but cannot reach your DHCP server anymore. Refer to subnet mask documentation for more details.

**DESCRIPTION**: Unable to reach the Mediatrix 4102 after changing the Ethernet speed at run-time.

**POSSIBLE CAUSE**: Some hubs cannot adapt completely their port speed at run-time.

**SOLUTION**: Always restart the Mediatrix 4102 for the new setting to take effect. See "Ethernet Connection Speed" on page 176 for more details.

**DESCRIPTION**: In the case where my NAT/Firewall device is connected to the *Computer* port of the Mediatrix 4102 and I have a PC connected to the NAT/Firewall, the PC has limited web access in time.

**POSSIBLE CAUSE**: The NAT/Firewall device does not support well the small DHCP lease time of the Mediatrix 4102 (30 seconds).

**SOLUTION**: Modify the *ipRoutingDhcpServerLeaseTime* variable with a value greater than 30 seconds, for instance 3600 seconds (1 hour). See "Enabling TAS" on page 209 for more details.

However, note that the downtime will be greater when the ISP gives another IP address. Try several values and find out what is the smallest setting for your NAT/firewall device.

**DESCRIPTION**: The PC connected to the *LAN* connector of the Mediatrix 4102 cannot register to IGMP services.

**POSSIBLE CAUSE**: The Mediatrix 4102 does not support the IGMP (Internet Group Management Protocol) protocol.

**SOLUTION**: There are no solutions.

**DESCRIPTION**: When I install a Mediatrix 4102 in an enterprise network and there is a PC connected to the port, the PC will not receive the WINS Server.

**POSSIBLE CAUSE**: The embedded DHCP server of the Mediatrix 4102 does not support the WINS Server. See "DHCP Server" on page 212 for more details.

**SOLUTION**: Disable the TAS feature as described in "Enabling TAS" on page 209 because the IP address is not an issue in private networks.

**DESCRIPTION**: Setting the MIB variable *voiceIfAdaptativeJitterBufferEnable* to **disable** has no effect.

**POSSIBLE CAUSE**: You cannot disable the adaptative jitter buffer on the Mediatrix 4102.

**SOLUTION**: If you set the *voiceIfTargetJitterBufferLength* and *voiceIfMaxJitterBufferLength* variables to the same value, you will have a non-adaptative jitter buffer. See "Adaptative Jitter Buffer" on page 274 for more details.

**DESCRIPTION**: When I set values such as the User Name and Display Name, the value is not accepted and is reset to its default value once the Mediatrix 4102 restarts.

**POSSIBLE CAUSE**: When you enter values that contain non-standard English characters in entries that accept strings of characters, this invalidates the value and resets it to its default value. However, this may be visible only once the Mediatrix 4102 restarts.

**SOLUTION**: Make sure that your string of characters contain only characters that are part of the following ASCII characters list:

```
10  LF, line feed              62  >, greater than          94  ^, caret
13  CR, carriage return        63  ?, question mark         95  _, underscore
32  space                      64  @, commercial at         96  `, back quote
33  !, exclamation mark        65  A                        97  a
34  ", double quote            66  B                        98  b
35  #, hash                    67  C                        99  c
36  $, dollar                  68  D                        100 d
37  %, percent                 69  E                        101 e
38  &, ampersand               70  F                        102 f
39  ', quote                   71  G                        103 g
40  (, open parenthesis        72  H                        104 h
41  ), close parenthesis       73  I                        105 i
42  *, asterisk                74  J                        106 j
43  +, plus                    75  K                        107 k
44  ,, comma                   76  L                        108 l
45  -, minus                   77  M                        109 m
46  ., full stop               78  N                        110 n
47  /, oblique stroke          79  O                        111 o
48  0, zero                    80  P                        112 p
49  1                          81  Q                        113 q
50  2                          82  R                        114 r
51  3                          83  S                        115 s
52  4                          84  T                        116 t
53  5                          85  U                        117 u
54  6                          86  V                        118 v
55  7                          87  W                        119 w
56  8                          88  X                        120 x
57  9                          89  Y                        121 y
58  :, colon                   90  Z                        122 z
59  ;, semicolon               91  [, open square bracket   123 {, open curly bracket
60  <, less than               92  \, backslash            124 |, vertical bar
61  =, equals                  93  ], close square bracket  125 }, close curly bracket
                                                            126 ~, tilde
```

**DESCRIPTION**: Media5 Technical Support personnel asked me to enable the PCM traces. How do I do it?

**POSSIBLE CAUSE**: PCM traces are an efficient tool to identify problems with:

- Echo in your network
- DTMF signals
- Caller ID signals

- Fax signals (or false Fax detection)
- Message Waiting Indicator signals
- Any other analog signal

**SOLUTION**: Do the following:

*a.* Enable the PCM traces by setting the *mxDebugPcmCaptureEnable* MIB variable to **enable**.

*b.* Set the destination IP address for the PCM traces in the *mxDebugPcmCaptureIpAddress* MIB variable.

- This IP address does not have to be listening on ports 5001/2 - 6001/2, as it is easy to filter out ICMP “port unreachable” messages afterwards.
- The PCM traces destination must be set so it can be recorded in a Wireshark capture on your network, normally sent to the PC doing the capture.

*c.* Set the endpoint number on which to perform the PCM capture in the *mxDebugPcmCaptureEndpointNumber* variable.

For more details on the PCM traces, refer to *Technical Bulletin 0648 - PCM Traces*.

# Calling Issues

The following are general calling issues you may encounter.

**DESCRIPTION**: Impossible to make a call.

If the following happens:

- Dial tone present.
- *Power* LED lit.
- LED lit.

**POSSIBLE CAUSE**: Network communication is not working.

**SOLUTION**: Check that:

- The LAN cable is securely connected to the Mediatrix 4102 and to the network connector.
- You did not connect a crossover network cable.

**POSSIBLE CAUSE**: Configurable parameters of the Mediatrix 4102 are not set properly.

**SOLUTION**: Refer to this manual for a complete description of the configurable Mediatrix 4102 parameters.

**DESCRIPTION**: Cannot make or receive calls.

**POSSIBLE CAUSE**: There may be calls that have not been properly terminated, which causes a “leak” in the system.

**SOLUTION**: You can enable the SIP Context Snapshot time feature. This feature is used to find if there are improperly terminated calls. This could help to debug the system.

*a.* In the *syslogMIB*, set the *syslogMsgMaxSeverity* variable to **debug**.

*b.* Configure and enable the syslog feature.

*c.* In the *sipDebugMIB*, set the time, in minutes, between snapshots in the *sipDebugContextSnapshotTime* variable.

The list of contexts currently in use are periodically output as debug-level syslog messages. Note that enabling this feature will also trigger an instant snapshot.

To disable the feature, set this variable to zero (0).

Note that this feature will generate more syslog traffic, about 20 messages at each x minutes.

**Note:** This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "MIB Structure" on page 147 for more details.

**POSSIBLE CAUSE**: It is possible that the unit is refreshing its registration and has entered a race condition between the refresh and the SIP timeouts. Normally, the Mediatrix 4102 cannot make or receive calls until the REGISTER request has completed successfully. Because the timeout for a SIP transaction in UDP is 32 seconds, it is possible to have an ongoing re-REGISTER transaction at the same moment that the registration itself expires. This could happen if the *sipReRegistrationTime* variable is set to a value lower than 32. In that case, the user agent becomes unregistered, and will become registered again only when the re-REGISTER request is answered with a positive response from the server.

**SOLUTION**: Set the *sipUnregisteredPortBehavior* variable to **enablePort**. This way, when an endpoint is not registered, it is still enabled. The user can receive and initiate calls. See "Unregistered Line Behaviour" on page 154 for more details. See also "Refreshing Registration" on page 296 for more details on the re-registration feature.

---

**DESCRIPTION**: When making a 3-way conference, part of the conversation is lost, resulting in a choppy voice.

---

**POSSIBLE CAUSE**: The packetization period (ptime) is not the same for all the participants of the conference, which causes the choppy voice issue.

**SOLUTION**: For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See "Packetization Time" on page 266 for more information on how to set the packetization period of the Mediatrix 4102.

---

**DESCRIPTION**: Unable to establish a call from the Mediatrix 4102 to a user agent such as an IP phone, a gateway or another access device.

---

**POSSIBLE CAUSE**: When the Mediatrix 4102 – with its T.38 capability enabled – tries to establish a call with a user agent that does not support T.38, this a user agent rejects the call instead of ignoring the capability it does not support, i.e., T.38.

**SOLUTION**: Disable the T.38 capability in the Mediatrix 4102. See "T.38 Fax" on page 284 for more details.

# Fax Issues

The following gives information pertaining to faxes. This includes a list of fax models tested with the Mediatrix 4102 and some specific issues the unit may encounter.

---

**DESCRIPTION**: "Poor line condition" error during a fax transmission.

---

**POSSIBLE CAUSE**: The analog transmission between the fax machine and the Mediatrix 4102 is flaky, preventing the fax transmission to terminate properly. This problem is known to occur with some fax machines and it can also occur with a few fax modems.

**SOLUTION**: Set the *Input sound level* to **-6 dB**. If this still does not solve the problem, try the **+6 dB** value. See "User Gain" on page 278 for more details.

**DESCRIPTION**: Unable to send a fax in T.38 and Clear Channel.

**POSSIBLE CAUSE**: To properly send faxes, both units must be configured with the same settings. If you are attempting to send a fax and the transmission fails, there could be many reasons for this, but most likely the fax codec settings are at fault. The following explains the logic behind fax transmissions.

When transmitting a fax, Unit A first verifies if Unit B supports the codec you have set in Unit A. If the codec is supported, the fax should be transmitted properly.

If the fax codec is not supported by Unit B, Unit A tries to find a common preferred G.711 clear channel codec between the two units. If Unit A finds one, it uses this common clear channel codec and the fax should be transmitted properly. If there are no common clear channel codecs between the units, the fax transmission fails.

**SOLUTION**: To avoid fax transmission problems, configure both units with the same T.38 and clear channel settings and the fax should be sent properly.

**DESCRIPTION**: The T.38 fax transmission fails.

**POSSIBLE CAUSE**: The Mediatrix 4102 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4102 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message.

Information from RFC 3264 (An Offer/Answer Model with Session Description Protocol (SDP)) - section 5.1: Once the offerer has sent the offer, it must be prepared to receive media for any recvonly streams described by that offer. It must be prepared to send and receive media for any sendrecv streams in the offer, and send media for any sendonly streams in the offer (of course, it cannot actually send until the peer provides an answer with the needed address and port information). In the case of RTP, even though it may receive media before the answer arrives, it cannot send RTCP receiver reports until the answer arrives.

**SOLUTION**: Be sure to reply to the “INVITE” message by a “200 OK” before sending any T.38 message to the Mediatrix 4102.

**DESCRIPTION**: Voice does not switch back to the original negotiated codec after a clear channel fax is performed.

**POSSIBLE CAUSE**: The Mediatrix 4102 suffers from a limitation of its DSP. The Mediatrix 4102 cannot detect the end of a clear channel fax, which means that the unit cannot switch back to the original negotiated codec if this codec was not a clear channel codec, e.g., a session established in G.729.

When the unit detects a fax, it automatically switches to a negotiated clear channel codec such as PCMU (if there is no T.38 or if T.38 negotiation failed). Once the fax is terminated, the Mediatrix 4102 is not notified by the DSP. The unit thus stays in the clear channel codec and does not switch back to G.729.

**SOLUTION**: There is no solution.

## Tested Fax Models

The following table lists the fax models tested with the Mediatrix 4102 for the T.38 protocol. Each of these fax models has been emulated and tested with each other by using the FaxLab® fax/telephony testing tool.

**Table 248:** Tested Fax Models

| Make | Models | |
|---|---|---|
| Brother | • 6650MC<br>• 7150C<br>• FAX-190<br>• FAX-580MC<br>• Intellifax 600<br>• Intellifax 625 | • Intellifax 950M<br>• Intellifax 2500<br>• MFC 4550<br>• MFC 4600<br>• MFC 4650 |
| Panasonic | • KXF-500<br>• KXF-580<br>• KXF-1600<br>• KXF-3000<br>• KX-FP270<br>• KX-FPC95 | • PX-5<br>• PX-150<br>• PX-350<br>• UF-880<br>• UF-V60 |
| Sharp | • FO-145<br>• FO-235<br>• FO-445<br>• FO-5400<br>• UX-104<br>• UX-108 | • UX-117<br>• UX-256<br>• UX-460<br>• UX-1400<br>• UX-3600M |
| Canon | • B70<br>• Fax 750<br>• Fax B340<br>• L777<br>• MultiPass C530<br>• MultiPass C545<br>• MultiPass C555 | • MultiPass C560<br>• MultiPass C755<br>• MultiPass C2500<br>• MultiPass C5500<br>• MultiPass L6000<br>• MultiPass TF-301 |
| Xerox | • 3004<br>• 7021<br>• 7024<br>• 7033<br>• WorkCenter 250 | • WorkCenter 470cx<br>• WorkCenter 480cx<br>• WorkCenter XE90fx<br>• WorkCenter XK50cx |
| Hewlett Packard | • Fax-200<br>• Fax 920<br>• LaserJet 3200 | • OfficeJet<br>• OfficeJet 350<br>• OfficeJet 570 |

## Issues Arising from Specific Combinations/Scenarios

The following are very specific issues the Mediatrix 4102 may experience with certain types of faxes.

**DESCRIPTION**: Canon B320 fax limitations.

**ISSUE**: The Mediatrix 4102 may not be compatible with the Canon B320 fax machine. The problem is in a T.38 transmission only and when the Canon B320 is the local fax attached to the Mediatrix 4102 and is the document receiver.

**DESCRIPTION**: When using the Mediatrix 4102 with the Cyberguard SG530 broadband router, the router blocks fax transmissions.

**POSSIBLE CAUSE**: Cyberguard Version 2.0.2 seems to be the problem.

**SOLUTION**: Upgrade the Cyberguard to version 2.1.3.

# Configuration Issues

The following are issues you may encounter when changing the Mediatrix 4102 configuration.

**DESCRIPTION**: When the Mediatrix 4102 configuration is entirely static and I change the configuration source of any server from static to DHCP, the service related to the server is not accessible.

**POSSIBLE CAUSE**: If none of the *xxxConfigSource* variables (in the *ipAddressStatus* folder) are set to **dhcp**, then the Mediatrix 4102 does not send a DHCP REQUEST message. This is the case if:

- you set all *xxxSelectConfigSource* variables to something other than **dhcp** and you restart the Mediatrix 4102, or
- you select the **setConfigSourcesStatic** option of the *sysAdminCommand* variable and you restart the Mediatrix 4102.

Whenever the *xxxSelectConfigSource* variable of a specific server, e.g., syslog server, is set to **dhcp**, then no IP address can be assigned to that server (this does not trigger a DHCP request).

The service is therefore not functional, the corresponding *xxxHost* variable is set to **0.0.0.0**, and the corresponding *xxxPort* variable is not accessible (the GET request result is ERROR), in the *ipAddressStatus* folder.

**Note:** In the case of the SIP servers, the corresponding *xxxPort* variable is accessible.

**SOLUTION**: Restart the Mediatrix 4102 or set the proper *xxxSelectConfigSource* variable to **static**.

# Software Upgrade Issues

The following are issues you may encounter when performing a software upgrade operation.

**DESCRIPTION**: An error occurs when the Mediatrix 4102 attempts to communicate with the image server.

**POSSIBLE CAUSE**: The directory specified in the upgrade command does not exist or does not contain the files required for the software download process.

**SOLUTION**:

- Check the directory name.
- Be sure that the directory contains files. If not, extract them from the zip file again. See "Download Procedure" on page 243 for more details.
- Be sure that the software server is running and properly configured.

**POSSIBLE CAUSE**: The IP address of the software server is not the correct one.

**SOLUTION**:

- Check the given IP address.
- Check the IP port.

---

**DESCRIPTION**: An error occurs when the Mediatrix 4102 attempts to transfer the software upgrade.

---

**POSSIBLE CAUSE**: The Ethernet cable has become disconnected from the Mediatrix 4102 or the PC running the file transfer.

**SOLUTION**: Reconnect the cable and start again.

**POSSIBLE CAUSE**: Power to the Mediatrix 4102 has been disrupted during the file transfer.

**SOLUTION**: Check the power connection to the Mediatrix 4102 and start again.

---

**DESCRIPTION**: When downgrading the Mediatrix 4102 to a previous version of the application software, the unit does not restart, the *ETH* LED is blinking and all other LEDs are off.

---

**POSSIBLE CAUSE**: The default router IP address is set to 0.0.0.0, which is not supported by the version to which you downgraded.

**SOLUTION**: Perform a recovery mode or a factory reset procedure after proceeding with the downgrade operation.

- If you perform a recovery mode as per "Recovery Mode" on page 21, you must manually change the default router IP address to a valid address other than 0.0.0.0, then restart the Mediatrix 4102.
- If you perform a factory reset procedure as per "Factory Reset" on page 22, everything should be working properly. However, this deletes any custom setting you may have done in other variables as it reverts the Mediatrix 4102 back to its default factory settings.

---

**DESCRIPTION**: The TFTP server does not recognize the download path and produces an error.

---

**POSSIBLE CAUSE**: You should use the "/" character when defining the path to indicate sub-directories, i.e., *c:/temp/download*. However, some TFTP servers on the Windows operating system do not recognize the "/" character and produce an error.

**SOLUTION**: Use the "\" character in the path definition.

---

**DESCRIPTION**: Performing a software download takes an unusually long time.

---

**POSSIBLE CAUSE**: If the following happens:

- Any information is set to come from the DHCP server (for example, the SNTP server address) and the DHCP server cannot be reached.
- The primary software server address is invalid (either set by DHCP or static).

The unit tries to reach the primary software server without realizing that the address is invalid. It keeps trying for a few minutes, even if the download procedure fails.

This delay is caused by the Mediatrix 4102 that cannot function as configured if part of its configuration (the DHCP information) is unavailable. Furthermore, there is an issue with switches that use the Spanning Tree Protocol. When this protocol is enabled, the Mediatrix 4102 may be denied from the network for a certain time, which causes the long delay.

**SOLUTION**: Media5 recommends to set up all information to use a valid static value, or have a DHCP server answer the requests. See "Static Configuration" on page 157 for more details.

# SNMP Management Software Issues

The following are issues you may encounter when trying to contact the Mediatrix 4102 with a SNMP management software.

**DESCRIPTION**: The SNMP network management software cannot access the Mediatrix 4102.

**POSSIBLE CAUSE**: The SNMP network management software does not have the proper Mediatrix 4102 information.

**SOLUTION**: Check that:

- The IP information for the Mediatrix 4102 is correctly configured.
- The Mediatrix 4102 was restarted after defining the IP information.
- The line through which you are trying to access the Mediatrix 4102 has been unlocked or is not the correct line. If it is locked, check the connections and network cabling for the connector.

Try to locate the Mediatrix 4102 IP address. If impossible, perform a recovery reset as indicated in section "Reset / Default Switch" on page 20.

**DESCRIPTION**: There is no response when trying to access the Mediatrix 4102.

**POSSIBLE CAUSE**: The Mediatrix 4102 speaks the three most common SNMP protocols: SNMPv1, SNMPv2c, and SNMPv3. If you try to access it by using any other protocol, it stays silent.

**DESCRIPTION**: The SNMP network manager does not receive Traps.

**POSSIBLE CAUSE**: The IP information is not correct.

**SOLUTION**: Check that the IP information (IP address + IP port) of the SNMP network manager software is correctly recorded by the Mediatrix 4102.

**DESCRIPTION**: When trying to set a variable, the Mediatrix 4102 does no respond, nor sends an error message.

**POSSIBLE CAUSE**: In secure management mode, the Mediatrix 4102 does not accept SNMPv1 and SNMPv2c SET requests. However, the MIB variables are viewable in any management mode (secure and not secure).

**DESCRIPTION**: When entering a value such as ".23" in a MIB variable (for instance, *sipTransportQValue*), the Mediatrix 4102 returns a "Wrong value" error message.

**POSSIBLE CAUSE**: The Mediatrix 4102 does not support a value such as ".23".

**SOLUTION**: Enter a value such as "0.23" instead.

**DESCRIPTION**: When I try to set a variable with a MIB configuration tool such as Media5 Unit Manager Network, nothing happens.

**POSSIBLE CAUSE**: The variable may be in a MIB that is located under the *mediatrixExperimental* branch of the MIB structure.

Media5 configuration tool – the Unit Manager Network – does not support MIBs that are located under the *mediatrixExperimental* branch of the MIB structure. The Unit Manager Network does not have specific tasks to manage variables in experimental MIBs.

The *mediatrixExperimental* branch is the area where objects and events in MIBs under development can be placed without fear of conflicting with other MIBs. When the items rooted under an experimental sub-tree are ready for release, they will be under a permanent branch.

Even though the Unit Manager Network can view experimental MIBs, SNMP operations may not work properly on them.

**DESCRIPTION**: When viewing a table, the unit does not respond.

**POSSIBLE CAUSE**: It may take time to fill completely a table: from 1 to 5 seconds. This is normal, because the unit is an embedded device with limited processing power.

**DESCRIPTION**: Is it possible for a hacker to change the content of SNMPv3 variables once the Mediatrix 4102 is in secure mode management?

**POSSIBLE CAUSE**: In secure management mode, the Mediatrix 4102 works in SNMPv1 read-only, SNMPv2c read-only, and SNMPv3 read/write. SNMP requests using the first two protocols are read-only, and tables used for setting up SNMPv3 users hide the passwords they carry. Because hackers do not know what password to use in SNMPv3 requests, they cannot access the Mediatrix 4102 with read-write permission.

# Appendices

**Page Left Intentionally Blank**

APPENDIX

A

# Standards Compliance and Safety Information

This Appendix lists the various standards compliance of the Mediatrix 4102.

## Standards Supported

The Mediatrix 4102 complies to the following standards:

**Table 249:** Standards Compliance

| Category | Specification |
|---|---|
| Agency approvals | • UL<br>• European Union, CE mark (Declaration of Conformity)<br>• Anatel<br>• FCC |
| Safety standards | • UL60950-1 1st Edition (2006)<br>• CAN/CSA-C22.2 No. 60950-1-03 1st Edition (2006)<br>• IEC 60950 (1st Edition 2001 With all national deviations)<br>• Anatel Resolution 442 |
| Emissions | • FCC Part 15:1998 Class B<br>• EN55022 (2006) Class B, with amendments A1 (2007)<br>• EN61000-3-2 (1995) Harmonic current emissions<br>• EN61000-3-3 (1995) Voltage fluctuations and flicker<br>• Anatel Resolution 512:2008 |
| Immunity | EN55024:1998 with amendments A1 (2001) and A2 (2003) including the following:<br>• EN61000-4-2 (1995), ESD<br>• EN61000-4-3 (1996), Radiated RF<br>• EN61000-4-4 (1995), Burst Transients<br>• EN61000-4-5 (1995), Surge<br>• EN61000-4-6 (1996), Conducted RF<br>• EN61000-4-11 (1995), Voltage Dips and Interruptions |

**Note:** The standards compliance of the Mediatrix 4102 are printed on a sticker located on the bottom of the unit.

# Disclaimers

The following are the disclaimers related to the Mediatrix 4102.

## Federal Communications Commission (FCC) Part 15

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help

**Note:** Any changes or modifications not expressly approved by Media5 could void the user's authority to operate the equipment.

## CE Marking

DECLARATION OF CONFORMITY

We Media5 Corporation located at 4229 Garlock st. Sherbrooke, Québec, Canada J1L 2C8 declare that for the hereinafter mentioned product the presumption of conformity with the applicable essential requirements of DIRECTIVE 1999/5/EC OF THE EUROPEAN PARLIAMENT (RTTE DIRECTIVE) is given.

Any unauthorized modification of the product voids this declaration.

For a copy of the original signed Declaration Of Conformity please contact Media5 at the above address.

## RoHS China

**这个文件涉及的是在中华人民共和国境内进口或销售的电子信息产品**

**Include this document with all Electronic Information Products imported or sold in the People's Republic of China**

| 部件名称 (Parts) | 有毒有害物质或元素 (Hazardous Substance) | | | | | |
|---|---|---|---|---|---|---|
| | 铅 (Pb) | 汞 (Hg) | 镉 (Cd) | 六价铬 ($Cr^{6+}$) | 多溴联苯 (PBB) | 多溴二苯醚 (PBDE) |
| 塑料和聚合物部件 (Plastic and Polymeric parts) | ○ | ○ | ○ | ○ | × | × |
| 集成电路 (Integrated Circuit ) | × | ○ | × | ○ | × | × |

○： 表示该有毒有害物质在该部件所有均质材料中的含量均在 SJ/T-11363 - 2006 规定的限量要求以下。
Indicates that the concentration of the hazardous substance in all homogeneous materials in the parts is below the relevant threshold of the SJ/T-11363 - 2006 standard.

×： 表示该有毒有害物质至少在该部件的某一均质材料中的含量可能超出SJ/T-11363 - 2006规定的限量要求。
Indicates that the concentration of the hazardous substance of at least one of all homogeneous materials in the parts might exceed the relevant threshold of the SJ/T-11363 - 2006 standard.

除非另外特别的标注,此标志为针对所涉及产品的环保使用期限标志. 某些可更换的零部件会有一个不同的环保使用期限(例如,电池单元模块)贴在其产品上.

此环保使用期限只适用于产品是在产品手册中所规定的条件下工作.

The Environmentally Friendly Use Period (EFUP) for all enclosed products and their parts are per the symbol shown here, unless otherwise marked. Certain field-replaceable parts have a different EFUP (for example, battery modules) and so are marked to reflect such. The Environmentally Friendly Use Period is valid only when the product is operated under the conditions defined in the product manual.

# Translated Warning Definition

The following information provides an explanation of the symbols which appear on the Mediatrix 4102 and in the documentation for the product.

**Warning:** Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, you must be aware of the hazards involved with electrical circuitry and familiar with standard practices for preventing accidents.

**Waarschuwing:** Dit waarschuwingssymbool betekent gevaar. U overtreat in een situatie die lichamelijk letsel kan veroorzaken. Voordat u aan enige apparatuur gaat werken, dient u zich bewust te zijn van de bij elektrische schakelingen betrokken risico's en dient u op de hoogte te zijn van standaard maatregelen om ongelukken te voorkomen.

**Varoitus:** Tämä varoitusmerkki merkitsee vaaraa. Olet tilanteessa, joka voi johtaa ruumiinvammaan. Ennen kuin työskentelet minkään laitteiston parissa, ota selvää sähkökytkentöihin liittyvistä vaaroista ja tavanomaisista onnettomuuksien ehkäisykeinoista.

**Attention:** Ce symbole d'avertissement indique un danger. Vous vous trouvez dans une situation pouvant causer des blessures ou des dommages corporels. Avant de travailler sur un équipement, soyez conscient des dangers posés par les circuits électriques et familiarisez-vous avec les procédures couramment utilisées pour éviter les accidents.

**Warnung:** Dieses Warnsymbol bedeutet Gefahr. Sie befinden sich in einer Situation, die zu einer Körperverletzung führen könnte. Bevor Sie mit der Arbeit an irgendeinem Gerät beginnen, seien Sie sich der mit elektrischen Stromkreisen verbundenen Gefahren und der Standardpraktiken zur Vermeidung von Unfällen bewußt.

**Avvertenza:** Questo simbolo di avvertenza indica un pericolo. La situazione potrebbe causare infortuni alle persone. Prima di lavorare su qualsiasi apparecchiatura, occorre conoscere i pericoli relativi ai circuiti elettrici ed essere al corrente delle pratiche standard per la prevenzione di incidenti.

**Advarsel:** Dette varselsymbolet betyr fare. Du befinner deg i en situasjon som kan føre til personskade. Før du utfører arbeid på utstyr, må du vare oppmerksom på de faremomentene som elektriske kretser innebærer, samt gjøre deg kjent med vanlig praksis når det gjelder å unngå ulykker.

**Aviso:** Este símbolo de aviso indica perigo. Encontra-se numa situação que lhe poderá causar danos físicos. Antes de começar a trabalhar com qualquer equipamento, familiarize-se com os perigos relacionados com circuitos eléctricos, e com quaisquer práticas comuns que possam prevenir possíveis acidentes.

**¡Advertencia!:** Este símbolo de aviso significa peligro. Existe riesgo para su integridad física. Antes de manipular cualquier equipo, considerar los riesgos que entraña la corriente eléctrica y familiarizarse con los procedimientos estándar de prevención de accidentes.

**Varning!:** Denna varningssymbol signalerar fara. Du befinner dig i en situation som kan leda till personskada. Innan du utför arbete på någon utrustning måste du vara medveten om farorna med elkretsar och känna till vanligt förfarande för att förebygga skador.

# Safety Warnings

This section lists the following safety warnings:

- Circuit Breaker (20A) Warning
- TN Power Warning
- Product Disposal Warning
- No. 26 AWG Warning
- WAN, LAN, Phone-Fax 1 and Phone-Fax 2 Connectors Warning
- LAN and FXS Ports Connectors Warning
- Socket Outlet Warning

## Circuit Breaker (20A) Warning

**Warning:** This product relies on the building's installation for short-circuit (overcurrent) protection. Ensure that a fuse or circuit breaker no larger than 120 VAC, 20A U.S. (240 VAC, 10A international) is used on the phase conductors (all current-carrying conductors).

## TN Power Warning

**Warning:** The device is designed to work with TN power systems.

## Product Disposal Warning

**Warning:** Ultimate disposal of this product should be handled according to all national laws and regulations.

## No. 26 AWG Warning

**Warning:** To reduce the risk of fire, use only No. 26 AWG or larger telecommunication line cord.

## WAN, LAN, Phone-Fax 1 and Phone-Fax 2 Connectors Warning

**Warning:** Do not connect the *WAN*, *LAN*, *Phone-Fax 1* and *Phone-Fax 2* connectors directly to the Public Switched Telephone Network (PSTN), to an off premise application, an out of plant application, any exposed plant application, or to any equipment other than the intended application, connection may result in a safety hazard, and/or defective operation and/or equipment damage.

Exposed plant means where any portion of the circuit is subject to accidental contact with electric lighting or power conductors operating at a voltage exceeding 300V between conductors or is subject to lightning strikes.

### LAN anf FXS Ports Connectors Warning

**Warning:** Do not connect the LAN and the FXS ports connectors directly to the Public Switched Telephone Network (PSTN), to an off premise application, an out of plant application, any exposed plant application, or to any equipment other than the intended application, connection may result in a safety hazard, and/or defective operation and/or equipment damage.

Exposed plant means where any portion of the circuit is subject to accidental contact with electric lighting or power conductors operating at a voltage exceeding 300V between conductors or is subject to lightning strikes.

### Socket Outlet Warning

**Warning:** The socket outlet, if used, shall be located near the equipment and shall be easily accessible by the user.

## Safety Recommendations

To insure general safety follow these guidelines:

- Do not open or disassemble this product.
- Do not get this product wet or pour liquids into it.
- Do not perform any action that creates a potential hazard to people or makes the equipment unsafe.

**Caution:** When using this equipment, basic safety precautions should always be followed to reduce the risk of fire, electric shock and injury to persons, including the following:

- Do not use this product near water, for example, near a bath tub, wash bowl, kitchen sink or laundry tub, in a wet basement or near a swimming pool.
- Avoid using a telephone (other than a cordless type) during an electrical storm. There may be a remote risk of electric shock from lightning.
- Do not use the telephone to report a gas leak in the vicinity of the leak.

APPENDIX B

# Standard Hardware Information

The specifications and information regarding this product are subject to change without notice. Every effort is made to ensure the accuracy of this document. Because of ongoing product improvements and revisions, Media5 cannot guarantee its accuracy, nor can be responsible for errors or omissions. Please contact your Media5 sales representative to obtain the latest version of the technical specifications.

## Industry Standard Protocols

The Mediatrix 4102 has been designed to support all major industry standards used today, as well as those that will eventually be implemented at a later date. Because of this specific design characteristic, the Mediatrix 4102 can be integrated with existing telephone, fax and data equipment such as PCs and routers.

**Table 250:** Industry Standard Protocols

| Parameter | Description |
|---|---|
| Vocoders | • G.711 (a-law, u-law)<br>• G.726 (40, 32, 24, 16 kbit/s)<br>• G.729a<br>• G.729ab |
| IP Telephony Protocols | • SIP - RFC 3261 |
| Real-Time Transport Protocols | • RTP/RTCP - RFC 1889, RFC 1890, RFC 2833, RFC 3389<br>• Hook Flash Relay (RFC 2833) |
| Network Management Protocols | • SNMPv3<br>• DHCP - RFC 2131, RFC 2132<br>• TFTP - RFC 1350, RFC 2347, RFC 2348, RFC 2349<br>• Syslog - RFC 3164<br>• HTTP 1.0 - RFC 1945<br>• HTTP 1.1 - RFC 2616<br>• Basic and digest HTTP authentication - RFC 2617 |
| Data Features | • PPPoE client - RFC 1332, RFC 1661, RFC 1334, RFC 1994, RFC 2516, RFC 1471, RFC 1472, RFC 1473, RFC 1877. Note: some PPPoE RFCs are implemented partially.<br>• TFTPHTTPS or HTTP auto-provisioning<br>• Transparent IP address sharing (Media5 patent pending technology allowing the same IP address to be shared between both Ethernet ports and distinguishing voice traffic from data traffic)<br>• DHCP server<br>• STUN client |

**Table 250:** Industry Standard Protocols (Continued)

| Parameter | Description |
|---|---|
| QoS | • ToS<br>• DiffServ<br>• 802.1p<br>• 802.1Q<br>• STUN - RFC 3489 |

# Hardware Features

## Display

- Power LED
- LANactivity LED
- Activity/In-Use LED indication on FXS ports
- Ready LED

## Interfaces

- 2 x RJ-11 connectors, analog phone/fax (FXS) interface.
- 2 x RJ-45 connectors, 10/100 BaseT Ethernet access (autosense: up to 100 Mbits).

## Power

- External 12 Vdc power supply (wall plug or desktop, based on country/area model).
- Seamless switch over period if the client UPS detects a power loss and activates within  ms.

## Casing / Installation

- Casing: Desktop (Plastic ABS UL94 ).
- Installation: The Mediatrix 4102 is designed for the desktop or can be wall-mountable.

# Product Architecture Details

- Supports two concurrent communications using any vocoders.
- DSP-based DTMF detection generation.
- DSP-based fax/data relay.
- Embedded operating system with 32-bit real-time multitasking Kernel.
- Embedded IPv4 TCP/IP stack with configurable QoS implemented by:
    - ToS byte at Network layer 3
    - 802.1p at Data Link layer 2
- Network parameters assigned via DHCP

# Real Time Fax Router Technical Specifications

Automatic selection between voice and fax.

**Table 251:** Fax Technical Specifications

| Parameter | Description |
|---|---|
| Ethernet | 10/100 BaseT Ethernet |
| Data Link | Ethernet II |
| Network | IP (Internet Protocol) |
| Transport | TCP / UDP |
| Protocols | Group 3 Fax<br>Clear channel (G.711), G.726, or T.38 Real Time Fax Over IP protocol Stack |
| Fax Data Compression | MH |
| Fax Transmission | Up to 14.4 kbps |

# Analog Line Interface (FXS)

- RJ-11 connectors
- Direct connection to a fax machine or telephone (Internal installation and internal cabling)
- DC feeding of the access line protected for over voltage
- Loop current detection and hook flash detection capable
- Generation of Selective Ring

**Table 252:** Analog Line Interface

| Parameter | Description |
|---|---|
| Trunk Type | Loop Start: capable of Wink and Immediate signalization |
| Ring Source | 50 VRMS max @ 20 up to 50 Hz (selectable) sine signal |
| Nominal Impedance | BellCore compliant 600/900 ohms default setting. Impedance Software Configurable. |
| Ring Drive Capacity | Up to 4 ringer equivalents (4 RENs) per port. |
| Loop Current Range | 15 to 32 mA factory set. Default 20 mA regulated. |
| Ring Trip Detection Time | 2 ring cycles max |
| On Hook Voltage | -48 VDC |
| Frequency Response | 200 Hz to 3400 Hz ±3 dB (Tx/Rx) |
| Return Loss | 500-3200 Hz: 30 dB |

# Audio Specifications

- Software input and output level adjustable within the range of -30 dB to +20 dB.
- Software-adjustable dynamic and static jitter buffer protection.
- Programmable by country: Call progress tone generation including dial tone, busy tone, ringback and error tones.
- DSP-based echo control device.
- Silence detection/suppression level software adjustable.

# DTMF Tone Detection

**Table 253:** DTMF Tone Detection

| Parameter | Description |
|---|---|
| 16-Digit DTMF Decoding | 0 to 9, *, #, A, B, C, D |
| Permitted Amplitude Tilt | High frequency can be +2 dB to -8 dB relative to low frequency |
| Dynamic Range | -35 dBm to +3 dBm per tone |
| Frequency Accept | ± 1.5% of nominal frequencies |
| Minimum Tone Duration | 40 ms, can be increased with software configuration |
| Interdigit Timing | Detects like digits with a 40 ms interdigit delay |

# DTMF Tone Generation

**Table 254:** DTMF Tone Generation

| Parameter | Description |
|---|---|
| Per Frequency Nominal | -6 dBm to -4 dBm |
| Frequency Deviation | Within 1.5% of nominal values |

# MTBF Value

The estimated Mean Time Before Failure (MTBF) value of the Mediatrix 4102 is 750,000 hours at 25 degrees Celsius ambient temperature (excluding the power adaptor). It has been defined using RelCalc v5.0, Bellcore method (LimitedStress - Method I, Case 3).

# Power Consumption

## Measurements at the DC input

**Table 255:** Power Consumption at the DC Input

| Parameter | Description |
|---|---|
| Idle Mode, 12Vdc | 240 mA |
| 2 Extensions Off-Hook: 12Vdc | 450 mA |
| 2 Extensions ringing: 12Vdc | 475 mA |

# Operating Environment

**Table 256:** Operating Environment

| Parameter | Description |
|---|---|
| Operating Temperature | -40°C to 85°C |
| Humidity | Up to 85 %, non-condensing |
| Storage | -40°C to +85°C |

# Dimensions and Weight

**Table 257:** Dimensions and Weight

| Parameter | Description |
|---|---|
| Dimensions | 3.1 cm x 12.7 cm x 9.9 cm - 1.2 in. x 5 in. x 3.9 in. (approx.) |
| Weight | 170 g (0.37 lb) |

# Warranty

All Media5 products carry Media5's standard three-year hardware and software warranty. An extended warranty is available.

APPENDIX C

# Cabling Considerations

This Appendix describes the pin-to-pin connections for cables used with the Mediatrix 4102.

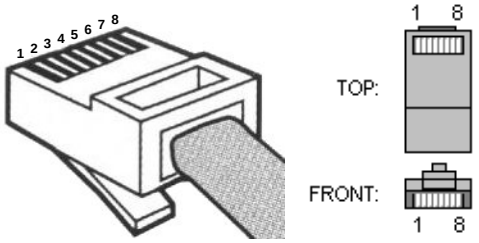
**Warning:** To reduce the risk of fire, use only No. 26 AWG or larger telecommunication line cord.

## RJ-45 Cable

The RJ-45 connector is commonly used for network cabling and for telephony applications. It is used to wire both ends identically so the signals pass straight through.

RJ-45 cabling is also known as Twisted-pair Ethernet (TPE), Unshielded twisted pair (UTP) and 10/100 Base-T.

**Figure 105:** RJ-45 Cable




### Straight Through Cable

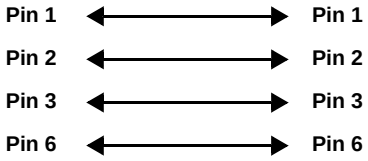
A RJ-45 straight through cable is used to connect a computer to a network device. For example straight through cables are the type of cables that connect a computer to a network hub, network switch, and network routers.

**Table 258:** RJ-45 Pinout Information

| Pin # | Function | Colour Coding | |
|---|---|---|---|
| | | EIA/TIA 568A | EIA/TIA 568B<br>AT&T 258A |
| 1 | Transmit + | White with green stripe | White with orange stripe |
| 2 | Transmit - | Green with white stripe or solid green | Orange with white stripe or solid orange |
| 3 | Receive + | White with orange stripe | White with green stripe |
| 4 | N/A | Blue with white stripe or solid blue | Blue with white stripe or solid blue |
| 5 | N/A | White with blue stripe | White with blue stripe |
| 6 | Receive - | Orange with white stripe or solid orange | Green with white stripe or solid green |
| 7 | N/A | White with brown stripe or solid brown | White with brown stripe or solid brown |
| 8 | N/A | Brown with white stripe or solid brown | Brown with white stripe or solid brown |

The RJ-45 cable uses two pairs of wires: one pair for transmission and the second pair for reception. It is wired so that pins 1 & 2 are on one twisted pair and pins 3 & 6 are on a second pair according to common wiring standards which meet the EIA/TIA T568A and T568B requirements.

**Figure 106:** Straight Through Connectivity




### Pin Name And Function

The following is the meaning of each pin in a RJ-45 cable.

**Table 259:** Pin Name and Function

| Pin # | Name | Function |
|---|---|---|
| 1 | Transmit Data Plus | The positive signal for the TD differential pair. This signal contains the serial output data stream transmitted onto the network. |
| 2 | Transmit Data Minus | The negative signal for the TD differential pair. This contains the same output as pin 1. |
| 3 | Receive Data Plus | The positive signal for the RD differential pair. This signal contains the serial input data stream received from the network. |
| 4 | not connected | |
| 5 | not connected | |
| 6 | Receive data minus | The negative signal for the RD differential pair. This signal contains the same input as pin 3. |
| 7 | not connected | |
| 8 | not connected | |

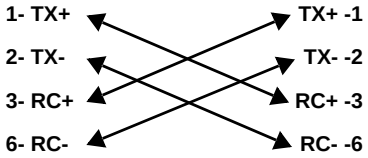
## Crossover Cable

A RJ-45 crossover cable is used when only two systems are to be connected to each other, peer to peer, at the Ethernet Cards by “crossing over” (reversing) their respective pin contacts. An example would be connecting two computers together to create a network. The crossover eliminates the need for a hub when connecting two computers. A crossover cable may also be required when connecting a hub to a hub, or a transceiver to transceiver or repeater to repeater. When connecting a hub to a transceiver, a straight through cable is always used.

**Note:** This is not an IEEE supported configuration and should be used for test purposes only.

A crossover cable is sometimes called a null modem. The coloured wires at either end are put into different pin numbers, or crossed over.

**Figure 107:** Crossover Connectivity



# RJ-11 (Telephone) Cable

The RJ-11 cable is commonly used for telephone connection.

**Caution:** Do not plug a phone jack connector into an RJ-45 port.

## Wiring Conventions

For telephone connections, a cable requires one pair of wires. Each wire is identified by different colors. For example, one wire might be red and the other, red with white stripes. Also, an RJ-11 connector must be attached to both ends of the cable.

Each wire pair must be attached to the RJ-11 connectors in a specific orientation. The following figure illustrates how the pins on the RJ-11 connector are numbered. Be sure to hold the connectors in the same orientation when attaching the wires to the pins.

**Figure 108:** RJ-11 Connector Pin Numbers

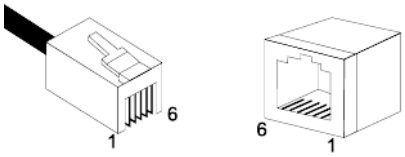



**Table 260:** RJ-11 Pinout Information

| Pin # | Function |
|---|---|
| 1 | Not used |
| 2 | Not used |
| 3 | Ring |
| 4 | Tip |
| 5 | Not used |
| 6 | Not used |

The RJ-11 pair of wires is wired so that pins 3 and 4 are connected to the Ring and Tip, which meets the following requirements:

- EIA/TIA-IS 968
- CS-03 Issue 8, Part III requirements.

**Warning:** The RJ-11 cable should comply with UL 1863 and CSA C22.2 No 233 standards.

APPENDIX D

# Country-Specific Parameters

The following parameters differ depending on the country in which you are.

## Definitions

The following are some useful definitions.

**Table 261:** Definitions

| Term | Description |
|---|---|
| Dial Tone | Indicates the line is ready to receive dialing. |
| Busy Tone | Indicates the line or equipment is in use, engaged or occupied. |
| Ringback Tone | Indicates the called line is ringing out. |
| Special Information Tone | Identifies network-provided announcements. |
| Stutter Dial Tone | Notifies the user that they have a voice mail message when the phone does not or cannot have a message-waiting light. |
| Confirmation Tone | Confirms a command performed by the user (such as activate a service). |
| Receiver Off Hook (ROH) Tone | Indicates that the telephone is not hung up correctly. |
| Message Waiting Indicator Tone | Indicates there is a message waiting somewhere for the owner of the phone. |
| Network Congestion Tone | Indicates that all switching paths are busy, all toll trunks are busy, or there are equipment blockages. |

### Conventions

The following conventions apply to this Appendix.

#### Frequencies

- Symbol “*” means modulated. For instance: 425 Hz * 25 means 425 Hz modulated at 25 Hz.
- Symbol “+” means added. For instance: 425 Hz + 330 Hz means that both 425 Hz and 330 Hz sines are played at the same time.
- When a tone is composed of more than one frequency, if not otherwise specified, the given electrical level applies to each frequency taken separately.

#### Impedance

Impedance is the apparent resistance, in an electric circuit, to the flow of an alternating current, analogous to the actual electrical resistance to a direct current, being the ratio of electromotive force to the current.

When representing an impedance, the following applies:

- Symbol “//” means parallel.
- Symbol “+” means serial.

Furthermore, there are two types of impedances:

- Input Impedance
- Terminal Balance Return Loss (TBRL) Impedance

### *Input Impedance*

Impedance of the Mediatrix 4102 at the Tip and Ring wires.

### *Terminal Balance Return Loss (TBRL) Impedance*

Balance return loss attributable to transmission loss between two points. It is used to characterize an impedance balancing property of the 2-wire analog equipment port.

Each country has its own definition of the TBRL value. For instance, in North America, TIA/EIA 464 (and TIA/EIA 912) define two TBRL values:

- 600 Ω for “on-premise” or short loop ports.
- 350 Ω + (1000 Ω || 21 nF) for “off-premise” or long loop ports.

A wire length above 2.5 km is considered long loop according to TIA/EIA 912 section 6.4 (7)(b)).

In Europe, ETSI 300 439 also mentions a TBRL value. However, most European countries have different requirements regarding the TBRL Impedance. This is also true for other countries around the world. Each one of them has different requirements.

## Line Attenuation

Values are given in dBr (deciBel relative):

- A “+” for input means that the digital side is attenuated by x decibels relative to the analog side.
- A “+” for output means that the analog side is amplified by x decibels relative to the digital side.
- A “-” for input means that the digital side is amplified by x decibels relative to the analog side.
- A “-” for output means that the analog side is attenuated by x decibels relative to the digital side.

## On-Off Sequences

Values in bold are “on” cycles, where tones are audible. Values in normal style are “off” cycles, where tones are not audible. When not otherwise specified, sequences repeat forever. A “x” symbol means that the sequences between parenthesis is repeated x times. The next cycle(s) repeat forever, unless otherwise specified. Values are in seconds.

For instance:

`3*(`**`0.1`**` – 0.1) then `**`0.6`**` – 1.0 - `**`0.2`**` – 0.2`

means that the 0.1s on and 0.1s off sequence is repeated 3 times, afterwards the 0.6s on, 1.0s off, 0.2s on and 0.2s off sequence repeats forever.

## Distinctive Ring

The distinctive ring service allows you to have three different numbers with each their own ring. The numbers ring through a single line coming into the business or residence and each number can be distinguished by the pattern of the ring. These ring patterns are made up of various combinations of ring bursts.

This feature uses the "*Alert-Info*" header from the initial INVITE of a call to know if the call requires a distinctive ringing.

The supported values of the "A*lert-Info*" are:

**Table 262:** Distinctive RIng Patterns

| Alert-Info value | Ring Name | On – Off Sequence (s) |
|---|---|---|
| <http://127.0.0.1/Bellcore-dr2> or <Bellcore-dr2> | Bellcore-dr2 | **0.8** – 0.4, **0.8** – 4.0 |
| <http://127.0.0.1/Bellcore-dr3> or <Bellcore-dr3> | Bellcore-dr3 | **0.4** – 0.2, **0.4** – 0.2, **0.8** – 4.0 |
| <http://127.0.0.1/Bellcore-dr4> or <Bellcore-dr4> | Bellcore-dr4 | **0.3** – 0.2, **1.0** – 0.2, **0.3** – 4.0 |

The Mediatrix 4102 plays the default ring of the country selected if the *Alert-Info* value is not present or the value is not supported.

**Note:** Since the first pause of the distinctive ring is lower that 1 second, a splash ring followed by an Off of 1 second precedes the distinctive ring pattern.

# Australia

The following parameters apply if you have selected Australia as location.

## Australia 1

The following parameters apply if you have selected Australia1 as location.

**Table 263:** Australia 1 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz * 25 | **CONTINUOUS** | -18 dBm |
| Busy Tone | 425 Hz | **0.375** – 0.375 | -18 dBm |
| Ringback Tone | 425 Hz * 25 | **0.4** – 0.2, **0.4** – 2.0 | -17 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -20 dBm |
| Stutter Dial Tone | 425 Hz | **CONTINUOUS** | -18 dBm |
| Confirmation Tone | 450 Hz | (**0.15** – 0.15 – **0.15**) x 2 End | -18 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2067+2467+2600 Hz | **0.1** – 0.1 | -21 dBm |
| Message Waiting Indicator Tone | 425 Hz * 25 | **0.1** – 0.04, x72 | -18 dBm |
| Network Congestion Tone | 400 Hz | **0.375** – 0.375 | -18 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -6 dBr |

## Australia 2

The following parameters apply if you have selected Australia2 as location.

**Table 264:** Australia 2 Parameters

| Parameter | Value | **On –** Off **Sequence (s)** | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz * 25 | **CONTINUOUS** | -18 dBm |
| Busy Tone | 425 Hz | **0.375** – 0.375 | -18 dBm |
| Ringback Tone | 425 Hz * 25 | **0.4** – 0.2, **0.4** – 2.0 | -17 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -20 dBm |
| Stutter Dial Tone | 425 Hz | **CONTINUOUS** | -18 dBm |
| Confirmation Tone | 450 Hz | (**0.15** – 0.15 – **0.15**) x 2 End | -18 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2067+2467+2600 Hz | **0.1** – 0.1 | -21 dBm |
| Message Waiting Indicator Tone | 425 Hz * 25 | **0.1** – 0.04, x72 | -18 dBm |
| Network Congestion Tone | 400 Hz | **0.375** – 0.375 | -18 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -6 dBr |

## Australia 3

The following parameters apply if you have selected Australia3 as location.

**Table 265:** Australia 3 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz * 25 | **CONTINUOUS** | -18 dBm |
| Busy Tone | 425 Hz | **0.375** – 0.375 | -18 dBm |
| Ringback Tone | 425 Hz * 25 | **0.4** – 0.2, **0.4** – 2.0 | -17 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -20 dBm |
| Stutter Dial Tone | 425 Hz | **CONTINUOUS** | -18 dBm |
| Confirmation Tone | 450 Hz | (**0.15** – 0.15 – **0.15**) x 2 End | -18 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2067+2467+2600 Hz | **0.1** – 0.1 | -21 dBm |
| Message Waiting Indicator Tone | 425 Hz * 25 | **0.1** – 0.04, x72 | -18 dBm |
| Network Congestion Tone | 400 Hz | **0.375** – 0.375 | -18 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Austria

The following parameters apply if you have selected Austria as location.

## Austria 1

The following parameters apply if you have selected Austria1 as location.

**Table 266:** Austria Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 450 Hz | **CONTINUOUS** | -20 dBm |
| Busy Tone | 450 Hz | **0.3** – 0.3 | -20 dBm |
| Ringback Tone | 450 Hz | **1.0** – 5.0 | -20 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -20 dBm |
| Stutter Dial Tone | 450 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -20 dBm |
| Confirmation Tone | 450 Hz | (**0.1** – 0.1) x 3 End | -20 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 450 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -20 dBm |
| Network Congestion Tone | 450 Hz | (**0.1** – 0.1) x10, **CONTINUOUS** | -20 dBm |
| Ring | AC: 45 VRMS, 50 Hz<br>DC: 15 Vdc | **1.0** – 5.0 | |
| Input Impedance | 270 Ω + 750 Ω // 150 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -10 dBr |

## Austria 2

The following parameters apply if you have selected Austria2 as location.

**Table 267:** Austria 2 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 420 Hz | **CONTINUOUS** | -20 dBm |
| Busy Tone | 420 Hz | **0.4** – 0.4 | -20 dBm |
| Ringback Tone | 420 Hz | **1.0** – 5.0 | -20 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -20 dBm |
| Stutter Dial Tone | 380 + 420 Hz | **CONTINUOUS** | -20 dBm |
| Confirmation Tone | 420 Hz | (**0.1** – 0.1) x 3 End | -20 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 420 Hz | (**0.1** – 0.1) x 10 **CONTINUOUS** | -20 dBm |
| Network Congestion Tone | 420 Hz | **0.2** – 0.2 | -20 dBm |
| Ring | AC: 45 VRMS, 50 Hz<br>DC: 15 Vdc | **1.0** – 5.0 | |
| Input Impedance | 270 Ω + 750 Ω // 150 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -10 dBr |

# Brazil

The following parameters apply if you have selected Brazil as location.

**Table 268:** Brazil Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -15 dBm |
| Busy Tone | 425 Hz | **0.25** – 0.25 | -10 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -15 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | (3 x **0.3** – 2 x 0.03) – 1.0 | -15 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -15 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3 End | -15 dBm |
| Receiver Off Hook (ROH) Tone | 425 Hz | **0.25** – 0.25 | -10 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -15 dBm |
| Reorder Tone | 425 Hz | **0.75** – 0.25, **0.25** – 0.25 | -10 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2 | -10 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 900 Ω | | |
| Default Caller ID | TELEBRAS_DTMF | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# Chile

The following parameters apply if you have selected Chile as location.

## Chile 1

The following parameters apply if you have selected Chile1 as location.

**Table 269:** Chile 1 Parameters

| Parameter | Value | **On –** Off **Sequence (s)** | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -10 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -10 dBm |
| Ringback Tone | 400 Hz | **1.0** – 3.0 | -10 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -10 dBm |
| Stutter Dial Tone | 330 + 440 Hz | **CONTINUOUS** | -10 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3 End | -10 dBm |
| Receiver Off Hook (ROH) Tone | 425 Hz | **0.25** – 0.25 | -10 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 CONTINUOUS | -10 dBm |
| Network Congestion Tone | 400 Hz | **0.2** – 0.2 | -10 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 3.0 | |
| Input Impedance | 600 Ω | | |
| Tbrl-Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

## Chile 2

The following parameters apply if you have selected Chile2 as location.

**Table 270:** Chile 2 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -10 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -10 dBm |
| Ringback Tone | 400 Hz | **1.0** – 3.0 | -10 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -10 dBm |
| Stutter Dial Tone | 330 + 440 Hz | **CONTINUOUS** | -10 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3 End | -10 dBm |
| Receiver Off Hook (ROH) Tone | 425 Hz | **0.25** – 0.25 | -10 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 CONTINUOUS | -10 dBm |
| Network Congestion Tone | 400 Hz | **0.2** – 0.2 | -10 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 3.0 | |
| Input Impedance | 900 Ω | | |
| Tbrl-Impedance | 900 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# China

The following parameters apply if you have selected China as location.

**Table 271:** China Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 450 Hz | **CONTINUOUS** | -10 dBm |
| Busy Tone | 450 Hz | **0.35** – 0.35 | -10 dBm |
| Ringback Tone | 450 Hz | **1.0** – 4.0 | -10 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -10 dBm |
| Stutter Dial Tone | 450 Hz | **0.4** – 0.04 | -10 dBm |
| Confirmation Tone | 450 Hz | (**0.1** – 0.1) x 3, End | -10 dBm |
| Receiver Off Hook (ROH) Tone | 950 Hz<br>950 Hz<br>950 Hz<br>950 Hz | **15.0** – 15.0 – 15.0<br>15.0 – **15.0** – 15.0<br>15.0 – 15.0 – **15.0**<br>15.0 – 15.0 – 15.0 – **CONTINUOUS** | -25 dBm<br>-16 dBm<br>-8 dBm<br>-6 dBm |
| Message Waiting Indicator Tone | 450 Hz | **0.4** – 0.04 | -10 dBm |
| Network Congestion Tone | 450 Hz | **0.7** – 0.7 | -10 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Czech Republic

The following parameters apply if you have selected Czech Republic as location.

**Table 272:** Czech Republic Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **0.33** – 0.33, **0.66** – 0.66 | -12 dBm |
| Busy Tone | 425 Hz | **0.33** – 0.33 | -12 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -12 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -12 dBm |
| Stutter Dial Tone | 425 Hz | (**0.165** – 0.165) x 3, **0.66** – 0.66 | -12 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -12 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **0.33** – 0.33, **0.66** – 0.66 | -12 dBm |
| Network Congestion Tone | 425 Hz | **0.165** – 0.165 | -12 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | ETSI FSK | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# Denmark

The following parameters apply if you have selected Denmark as location.

**Table 273:** Denmark Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -15 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -10 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -15 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -15 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -15 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -15 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -15 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2 | -10 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 300 Ω + 1000 Ω // 220 nF | | |
| Default Caller ID | TDK_DTMF | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -6 dBr |

# France

The following parameters apply if you have selected France as location.

**Table 274:** France Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 440 Hz | **CONTINUOUS** | -16.9 dBm |
| Busy Tone | 440 Hz | **0.5** – 0.5 | -19.9 dBm |
| Ringback Tone | 440 Hz | **1.5** – 3.5 | -19.9 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | (3 x **0.3** – 2 x 0.03) – 1.0 | -19.9 dBm |
| Stutter Dial Tone | 440 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -16.9 dBm |
| Confirmation Tone | 440 Hz | (**0.1** – 0.1) x 3, End | -16.9 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 440 Hz | (**0.1** – 0.1) x 10 | -16.9 dBm |
| Network Congestion Tone | 440 Hz | **0.25** – 0.25 | -19.9 dBm |
| Ring | AC: 45 VRMS, 50 Hz<br>DC: 15 Vdc | **1.5** – 3.5 | |
| Input Impedance | 215 Ω + 1000 Ω // 137 nF | | |
| Default Caller ID | FRANCE: BELLCORE<br>FRANCE_ETSI_FSK: ETSI_FSK<br>FRANCE_ETSI_DTMF:ETSI_DTMF | | |
| FXS Line Attenuation (Input) | | | +1.9 dBr |
| FXS Line Attenuation (Output) | | | -8.9 dBr |

# Germany

The following parameters apply if you have selected Germany as location.

## Germany 1

The following parameters apply if you have selected Germany 1 as location.

**Table 275:** Germany 1 Parameters

| Parameter | Value | **On –** Off **Sequence (s)** | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -16 dBm |
| Busy Tone | 425 Hz | **0.48** – 0.48 | -16 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -16 dBm |
| Special Information Tone | 900 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -16 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -16 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -16 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 | -16 dBm |
| Network Congestion Tone | 425 Hz | **0.24** – 0.24 | -16 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -10 dBr |

## Germany 2

The following parameters apply if you have selected Germany 2 as location.

**Table 276:** Germany 2 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -16 dBm |
| Busy Tone | 425 Hz | **0.48** – 0.48 | -16 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -16 dBm |
| Special Information Tone | 900 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -16 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -16 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -16 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 | -13 dBm |
| Network Congestion Tone | 425 Hz | **0.24** – 0.24 | -13 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

## Germany 3

The following parameters apply if you have selected Germany 3 as location.

**Table 277:** Germany 3 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **0.2** – 0.2, **0.2** – 0.2, **0.2** – 0.8 | -16 dBm |
| Busy Tone | 425 Hz | **0.48** – 0.48 | -16 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -16 dBm |
| Special Information Tone | 900 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -16 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -16 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -16 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 | -16 dBm |
| Network Congestion Tone | 425 Hz | **0.24** – 0.24 | -16 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -10 dBr |

# Hong Kong

The following parameters apply if you have selected Hong Kong as location.

**Table 278:** Hong Kong Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 350 + 440 Hz | **CONTINUOUS** | -13 dBm |
| Busy Tone | 480 + 620 Hz | **0.5** – 0.5 | -13 dBm |
| Ringback Tone | 440 + 480 Hz | **0.4** – 0.2, **0.4** –3.0 | -13 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -16 dBm |
| Stutter Dial Tone | 350 + 440 Hz | (**0.1** – 0.1) x 20, **CONTINUOUS** | -16 dBm |
| Confirmation Tone | 350 + 440 Hz | **0.1** – 0.1, **0.3** – End | -16 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 350 + 440 Hz | (**0.2** – 0.2, **0.5** – 0.2) x 4, **CONTINUOUS** | -16 dBm |
| Network Congestion Tone | 480 + 620 Hz | **0.25** – 0.25 | -13 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** –3.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -6 dBr |

# Indonesia

The following parameters apply if you have selected Indonesia as location.

**Table 279:** Indonesia Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -9 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -9 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -9 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33** – 0.03, **0.33** – 0.03, **0.33** – 1.0 | -9 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -9 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -9 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 950 Hz | **0.33** – 0.03 | -9 dBm |
| Network Congestion Tone | 425 Hz | **0.25** – 0.25 | -9 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -3 dBr |

# Israel

The following parameters apply if you have selected Israel as location.

**Table 280:** Israel Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -15 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -15 dBm |
| Ringback Tone | 400 Hz | **1.0** – 3.0 | -15 dBm |
| Special Information Tone | 1000 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -15 dBm |
| Stutter Dial Tone | 400 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -15 dBm |
| Confirmation Tone | 400 Hz | **0.17** – 0.14, **0.34** | -15 dBm |
| Receiver Off Hook (ROH) Tone | 1440+2060+2452+2600 Hz | **0.12** – 0.88 | -20 dBm |
| Message Waiting Indicator Tone | 400 Hz | (**0.16** – 0.16) x 10, **CONTINUOUS** | -15 dBm |
| Network Congestion Tone | 400 Hz | **0.25** – 0.25 | -15 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 3.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Italy

The following parameters apply if you have selected Italy as location.

**Table 281:** Italy Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **0.2** – 0.2, **0.6** – 1.0 | -13 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -13 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -13 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -20 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **0.2** – 0.2, **0.6** – 1.0 | -13 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -13 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **0.2** – 0.2, **0.6** – 1.0 | -13 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2 | -13 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 180 Ω + 630 Ω // 60 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# Japan

The following parameters apply if you have selected Japan as location.

**Table 282:** Japan Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -13 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -13 dBm |
| Ringback Tone | 400 Hz * 16 | **1.0** – 2.0 | -16 dBm |
| Special Information Tone | 400 Hz | **0.1** – 0.1 | -13 dBm |
| Stutter Dial Tone | 400 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -13 dBm |
| Confirmation Tone | 400 Hz | (**0.1** – 0.1) x 3, End | -13 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 400 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -13 dBm |
| Network Congestion Tone | 400 Hz | **0.5** – 0.5 | -13 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 2.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Malaysia

The following parameters apply if you have selected Malaysia as location.

**Table 283:** Malaysia Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -14 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -18 dBm |
| Ringback Tone | 425 Hz | **0.4** – 0.2, **0.4** – 2.0 | -16 dBm |
| Special Information Tone | 900 Hz<br>1400 Hz<br>1800 Hz | **1.0**<br>**1.0**<br>**1.0** – 1.0 | -14 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -14 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3 End | -14 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -14 dBm |
| Network Congestion Tone | 425 Hz | **0.** – 0.25 | -18 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Mexico

The following parameters apply if you have selected Mexico as location.

**Table 284:** Mexico Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -14 dBm |
| Busy Tone | 425 Hz | **0.25** – 0.25 | -18 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -16 dBm |
| Special Information Tone | 900 Hz<br>1400 Hz<br>1800 Hz | **1.0**<br>**1.0**<br>**1.0** – 1.0 | -14 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -14 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -14 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 **CONTINUOUS** | -14 dBm |
| Network Congestion Tone | 425 Hz | **0.25** – 0.25 | -18 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -3 dBr |

# Netherlands

The following parameters apply if you have selected Netherlands as location.

**Table 285:** Netherlands Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -17 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -17 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -17 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -17 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -17 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -17 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 **CONTINUOUS** | -17 dBm |
| Network Congestion Tone | 425 Hz | **0.25** – 0.25 | -17 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 270 Ω + 750 Ω // 150 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# New Zealand

The following parameters apply if you have selected New Zealand as location.

**Table 286:** New Zealand Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -17 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -17 dBm |
| Ringback Tone | 400 Hz + 450 Hz | **0.4** – 0.2, **0.4** – 2.0 | -19 dBm |
| Special Information Tone | 1400 Hz | **0.1** – 0.1 | -17 dBm |
| Stutter Dial Tone | 400 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -17 dBm |
| Confirmation Tone | 400 Hz | (**0.1** – 0.1) x 3, End | -17 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 400 Hz | (**0.1** – 0.1) x12, **CONTINUOUS** | -17 dBm |
| Network Congestion Tone | 400 Hz | **0.25** – 0.25 | -17 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 300 Ω + 1000 Ω // 220 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# North America

The following parameters apply if you have selected North America as location.

## North America 1

The following parameters apply if you have selected North America 1 as location.

**Table 287:** North America 1 Parameters

| Parameter | Value | **On –** Off **Sequence (s)** | Elect. Levels |
|---|---|---|---|
| Dial Tone | 350+440 Hz | **CONTINUOUS** | -17 dBm |
| Busy Tone | 480+620 Hz | **0.5** – 0.5 | -21 dBm |
| Ringback Tone | 440+480 Hz | **2.0** – 4.0 | -19 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -14 dBm |
| Stutter Dial Tone | 350+440 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -17 dBm |
| Confirmation Tone | 350+440 Hz | (**0.1** – 0.1) x 3, End | -17 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 350+440 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -17 dBm |
| Network Congestion Tone | 480+620 Hz | **0.25** – 0.25 | -21 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **2.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Tbrl-Impedance[a] | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -3 dBr |

a. TBRL-Impedance for “on-premise" or short loop ports.

## North America 2

The following parameters apply if you have selected North America 2 as location.

**Table 288:** North America 2 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 350+440 Hz | **CONTINUOUS** | -17 dBm |
| Busy Tone | 480+620 Hz | **0.5** – 0.5 | -21 dBm |
| Ringback Tone | 440+480 Hz | **2.0** – 4.0 | -19 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -14 dBm |
| Stutter Dial Tone | 350+440 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -17 dBm |
| Confirmation Tone | 350+440 Hz | (**0.1** – 0.1) x 3, End | -17 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 350+440 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -17 dBm |
| Network Congestion Tone | 480+620 Hz | **0.25** – 0.25 | -21 dBm |
| Ring | AC: 45 VRMS, 20 Hz<br>DC: 15 Vdc | **2.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Tbrl-Impedance[a] | 350 Ω + 1000 Ω // 210 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | 0 dBr |

a. TBRL-Impedance for “off-premise" or long loop ports (wire length longer than 2.5 km).

# Russia

The following parameters apply if you have selected Russia as location.

**Table 289:** Russia Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -10 dBm |
| Busy Tone | 425 Hz | **0.4** – 0.4 | -10 dBm |
| Ringback Tone | 425 Hz | **0.8** – 3.2 | -10 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -17 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -10 dBm |
| Confirmation Tone | 1400 Hz<br>2060 Hz<br>2450 Hz<br>2600 Hz | **0.1** – 0.1 | -19 dBm<br>-19 dBm<br>-19 dBm<br>-19 dBm |
| Receiver Off Hook (ROH) Tone | 425 Hz | 3 x (**0.1** – 0.1), **CONTINUOUS** | -10 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10 **CONTINUOUS** | -10 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2 | -10 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.8** – 3.2 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | +2 dBr |
| FXS Line Attenuation (Output) | | | -2 dBr |

# South Africa

The following parameters apply if you have selected South Africa as location.

**Table 290:** South Africa Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 Hz | **CONTINUOUS** | -17 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -17 dBm |
| Ringback Tone | 400 Hz + 450 Hz | **0.4** – 0.2, **0.4** – 2.0 | -19 dBm |
| Special Information Tone | 1400 Hz | **0.1** – 0.1 | -17 dBm |
| Stutter Dial Tone | 400 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -17 dBm |
| Confirmation Tone | 400 Hz | (**0.1** – 0.1) x 3, End | -17 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 400 Hz | (**0.1** – 0.1) x12, **CONTINUOUS** | -17 dBm |
| Network Congestion Tone | 400 Hz | **0.25** – 0.25 | -17 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 300 Ω + 1000 Ω // 220 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

# Spain

The following parameters apply if you have selected Spain as location.

**Table 291:** Spain Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -10 dBm |
| Busy Tone | 425 Hz | **0.2** – 0.2 | -13 dBm |
| Ringback Tone | 425 Hz | **1.5** – 3.0 | -13 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -20 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -10 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -10 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -10 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2, **0.2** – 0.2, **0.2** – 0.6 | -13 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.5** – 3.0 | |
| Input Impedance | 220 Ω + 820 Ω // 120 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -7 dBr |

# Sweden

The following parameters apply if you have selected Sweden as location.

**Table 292:** Sweden Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -12.5 dBm |
| Busy Tone | 425 Hz | **0.25** – 0.25 | -12.5 dBm |
| Ringback Tone | 425 Hz | **1.0** – 5.0 | -12.5 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -22 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -12.5 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -12.5 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -12.5 dBm |
| Network Congestion Tone | 425 Hz | **0.25** – 0.75 | -12.5 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 5.0 | |
| Input Impedance | 200 Ω + 1000 Ω // 100 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -5 dBr |

# Switzerland

The following parameters apply if you have selected Switzerland as location.

**Table 293:** Switzerland Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 425 Hz | **CONTINUOUS** | -8 dBm |
| Busy Tone | 425 Hz | **0.5** – 0.5 | -13 dBm |
| Ringback Tone | 425 Hz | **1.0** – 4.0 | -13 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.333**<br>**0.333**<br>**0.333** – 1.0 | -13 dBm |
| Stutter Dial Tone | 425 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -8 dBm |
| Confirmation Tone | 425 Hz | (**0.1** – 0.1) x 3, End | -8 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 425 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -8 dBm |
| Network Congestion Tone | 425 Hz | **0.2** – 0.2 | -13 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 220 Ω + 820 Ω // 115 nF | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | 0 dBr |
| FXS Line Attenuation (Output) | | | -6.5 dBr |

# Thailand

The following parameters apply if you have selected Thailand as location.

**Table 294:** Thailand Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 400 * 50 Hz | **CONTINUOUS** | -16 dBm |
| Busy Tone | 400 Hz | **0.5** – 0.5 | -10 dBm |
| Ringback Tone | 400 Hz | **1.0** – 4.0 | -10 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -15 dBm |
| Stutter Dial Tone | 400 * 50 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -16 dBm |
| Confirmation Tone | 400 * 50 Hz | (**0.1** – 0.1) x 3, End | -16 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 400 * 50 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -16 dBm |
| Network Congestion Tone | 400 Hz | **0.3** – 0.3 | -10 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **1.0** – 4.0 | |
| Input Impedance | 600 Ω | | |
| Default Caller ID | BELLCORE | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -3 dBr |

# United Arab Emirates

The following parameters apply if you have selected the United Arab Emirates 2 as location.

**Table 295:** United Arab Emirates 2 Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 350+440 Hz | **CONTINUOUS** | -13 dBm |
| Busy Tone | 400 Hz | **0.375** – 0.375 | -13 dBm |
| Ringback Tone | 425 Hz | **0.4** – 0.2, **0.4** – 2.0 | -13 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -15 dBm<br>-15 dBm<br>-15 dBm |
| Stutter Dial Tone | 350+440 Hz | (**0.4** – 0.04) x 5, **CONTINUOUS** | -13 dBm |
| Confirmation Tone | 400 Hz | (**0.1** – 0.1) x 3 End | -13 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | (**0.1** – 0.1) | -19 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 600 Ω | | |
| FXS Line Attenuation (Input) | | | 3 dBr |
| FXS Line Attenuation (Output) | | | -3 dBr |

# UK

The following parameters apply if you have selected the United Kingdom as location.

**Table 296:** UK Parameters

| Parameter | Value | On – Off Sequence (s) | Elect. Levels |
|---|---|---|---|
| Dial Tone | 350+440 Hz | **CONTINUOUS** | -22 dBm |
| Busy Tone | 400 Hz | **0.375** – 0.375 | -19 dBm |
| Ringback Tone | 400+450 Hz | **0.4** – 0.2, **0.4** – 2.0 | -22 dBm |
| Special Information Tone | 950 Hz<br>1400 Hz<br>1800 Hz | **0.33**<br>**0.33**<br>**0.33** – 1.0 | -19 dBm |
| Stutter Dial Tone | 350+440 Hz | (**0.1** – 0.1) x 3, **CONTINUOUS** | -22 dBm |
| Confirmation Tone | 350+440 Hz | (**0.1** – 0.1) x 3, End | -22 dBm |
| Receiver Off Hook (ROH) Tone | 1400+2060+2450+2600 Hz | **0.1** – 0.1 | -19 dBm |
| Message Waiting Indicator Tone | 350+440 Hz | (**0.1** – 0.1) x 10, **CONTINUOUS** | -22 dBm |
| Network Congestion Tone | 400 Hz | **0.4** – 0.35, **0.225** – 0.525 | -19 dBm |
| Ring | AC: 45 VRMS, 25 Hz<br>DC: 15 Vdc | **0.4** – 0.2, **0.4** – 2.0 | |
| Input Impedance | 300 Ω + 1000 Ω // 220 nF | | |
| Default Caller ID | UK: BRITISH_TELECOM<br>UK_BELLCORE: BELLCORE<br>UK_CCA: CCA<br>UK_ETSI_FSK: ETSI_FSK | | |
| FXS Line Attenuation (Input) | | | -3 dBr |
| FXS Line Attenuation (Output) | | | -9 dBr |

APPENDIX E

# Glossary

**10 BaseT**

An Ethernet local area network that works on twisted pair wiring.

**100 BaseT**

A newer version of Ethernet that operates at 10 times the speed of a 10 BaseT Ethernet.

**Access Device**

Device capable of sending or receiving data over a data communications channel.

**A-Law**

The ITU-T companding standard used in the conversion between analog and digital signals in PCM (Pulse Code Modulation) systems. A-law is used primarily in European telephone networks and contrasts with the North American mu (µ)-law standard. See also *mu (µ)-law*.

**Access Concentrator**

A device that merges many data transmission signals onto a single shared channel in such a way that all the data channels can be active at the same time. The access concentrator supports dial-up modem calls, ISDN connections, frame relay traffic and multiprotocol routing.

**Analog Display Services Interface (ADSI)**

Telecommunications protocol standard that enables alternate voice and data capability over the existing analog telephone network. This means that in addition to the familiar voice response audio interface (where you listen to voice recordings and make menu selections using the telephone keypad), you can now see the menu and information on the screen display and make selections using soft keys. To use ADSI, you would need an ADSI capable device (as you would if you want the caller ID service).

**Area Code**

The preliminary digits that a user must dial to be connected to a particular outgoing trunk group or line. In North America, an area code has three digits and is used with a NXX (office code) number. For instance, in the North American telephone number *561-955-1212*, the numbers are defined as follows:

**Table 297:** North American Numbering Plan

| No. | Description |
|---|---|
| 561 | Area Code, corresponding to a geographical zone in a non-LNP (Local Number Portability) network. |
| 955 | NXX (office code), which corresponds to a specific area such as a city region. |
| 1212 | Unique number to reach a specific destination. |

Outside North America, the area code may have any number of digits, depending on the national telecommunication regulation of the country. In France, for instance, the numbering terminology is *xZABPQ 12 34*, where:

**Table 298:** France Numbering Plan

| No. | Description |
|---|---|
| x | Operator forwarding the call. This prefix can be made of 4 digits. |

**Table 298:** France Numbering Plan (Continued)

| No. | Description |
|---|---|
| Z | Geographical (regional) zone of the number (in France, there are five zones). It has two digits. |
| ABPQ | First four digits corresponding to a local zone defined by central offices. |
| 12 34 | Unique number to reach a specific destination. |

In this context, the area code corresponds to the *Z* portion of the numbering plan. Because virtually every country has a different dialing plan nomenclature, it is recommended to identify the equivalent of an area code for the location of your communication unit.

**Cable Modem**

A device that connects a computer to a local cable television line and receives data at about 1.5 Mbps. This data rate far exceeds that of the prevalent 28.8 and 56 Kbps telephone modems and the up to 128 Kbps of Integrated Services Digital Network (ISDN). It is about the data rate available to subscribers of Digital Subscriber Line (DSL) telephone service. A cable modem can be added to or integrated with a set-top box that provides your TV set with channels for Internet access. In most cases, cable modems are furnished as part of the cable access service and are not purchased directly and installed by the subscriber.

**Country Code (CC)**

In international direct telephone dialing, a code that consists of 1-, 2-, or 3-digit numbers in which the first digit designates the region and succeeding digits, if any, designate the country.

**Custom Local Area Signalling Services (CLASS)**

One of an identified group of network-provided enhanced services. A CLASS group for a given network usually includes several enhanced service offerings, such as incoming-call identification, call trace, call blocking, automatic return of the most recent incoming call, call redial, and selective forwarding and programming to permit distinctive ringing for incoming calls.

**Digital Signal Processor (DSP)**

Specialized computer chip designed to perform speedy and complex operations on digitized waveforms. Useful in processing sound (like voice phone calls) and video.

**Digital Subscriber Lines (DSL)**

A technology for bringing high-bandwidth information to homes and small businesses over ordinary copper telephone lines. xDSL refers to different variations of DSL, such as ADSL, HDSL, and RADSL.

**Domain Name Server (DNS)**

Internet service that translates domain names into IP addresses. To use a domain name, a DNS service must translate the name into the corresponding IP address. For instance, the domain name *www.example.com* might translate to 198.105.232.4.

**Dual-Tone Multi-Frequency (DTMF)**

In telephone systems, multi-frequency signalling in which a standard set combinations of two specific voice band frequencies, one from a group of four low frequencies and the other from a group of four higher frequencies, are used. Although some military telephones have 16 keys, telephones using DTMF usually have 12 keys. Each key corresponds to a different pair of frequencies. Each pair of frequencies corresponds to one of the ten decimal digits, or to the symbol “#” or “*”, the “*” being reserved for special purposes.

**Dynamic Host Configuration Protocol (DHCP)**

TCP/IP protocol that enables PCs and workstations to get temporary or permanent IP addresses (out of a pool) from centrally-administered servers.

**Echo Cancellation**

Technique that allows for the isolation and filtering of unwanted signals caused by echoes from the main transmitted signal.

**Far End Disconnect**

Refers to methods for detecting that a remote party has hung up. This is also known as Hangup Supervision. There are several methods that may be used by a PBX/ACD/CO to signal that the remote party has hung up, including cleardown tone, or a wink.

**Federal Communications Commission (FCC)**

U.S. government regulatory body for radio, television, interstate telecommunications services, and international services originating in the United States.

**Foreign Exchange Service/Station (FXS)**

A network-provided service in which a telephone in a given local exchange area is connected, via a private line, to a central office in another, i.e., “foreign”, exchange, rather than the local exchange area’s central office. This is the station (telephone) end of an FX circuit. An FXS port will provide dial tone and ring voltage.

**G.711**

ITU-T recommendation for an algorithm designed to transmit and receive A-law PCM (Pulse Code Modulation) voice at digital bit rates of 48 kbps, 56 kbps, and 64 kbps. It is used for digital telephone sets on digital PBX and ISDN channels.

**G.726**

Nn implementation of ITU-T G.726 standard for conversion linear or A-law or µ-law PCM to and from a 40, 32, 24 or 16 kbit/s channel.

**G.729**

A codec that provides near toll quality at a low delay which uses compression to 8 kbps (8:1 compression rate).

**Gateway**

A device linking two different types of networks that use different protocols (for example, between the packet network and the Public Switched Telephone Network).

**Impedance**

Impedance is the apparent resistance, in an electric circuit, to the flow of an alternating current, analogous to the actual electrical resistance to a direct current, being the ratio of electromotive force to the current.

**International Telecommunication Union (ITU)**

Organization based in Geneva, Switzerland, that is the most important telecom standards-setting body in the world.

**Internet-Drafts**

Internet-Drafts are working documents of the IETF, its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

**Internet Protocol (IP)**

A standard describing software that keeps track of the Internet’s addresses for different nodes, routes outgoing messages, and recognizes incoming messages.

**Jitter**

A distortion caused by the variation of a signal from its references which can cause data transmission errors, particularly at high speeds.

**Layer 2**

Layer 2 refers to the Data Link Layer of the commonly-referenced multilayered communication model, Open Systems Interconnection (OSI). The Data Link Layer is concerned with moving data across the physical links in the network.

The Data-Link Layer contains two sublayers that are described in the IEEE-802 LAN standards:

- Media Access Control (MAC)
- Logical Link Control (LLC)

**Layer 3**

Layer 3 refers to the Network layer of the commonly-referenced multilayered communication model, Open Systems Interconnection (OSI). The Network Layer is concerned with knowing the address of the neighbouring nodes in the network, selecting routes and quality of service, and recognizing and forwarding to the transport layer incoming messages for local host domains.

**Light Emitting Diode (LED)**

A semiconductor diode that emits light when a current is passed through it.

**Local Area Network (LAN)**

Data-only communications network confined to a limited geographic area, with moderate to high data rates. See also WAN.

**Management Information Base (MIB)**

Specifications containing definitions of management information so that networked systems can be remotely monitored, configured and controlled.

**Management Server**

Includes a web-based provisioning client, provisioning server, and SNMP proxy server used to manage all agents connected to the system. The Management Server provides Gateway provisioning, Monitoring, and Numbering Plan.

**Media Access Control (MAC) Address**

A layer 2 address, 6 bytes long, associated with a particular network device; used to identify devices in a network; also called hardware or physical address.

**Mu (µ)-Law**

The PCM (Pulse Code Modulation) voice coding and companding standard used in Japan and North America. See also *A-Law*.

**Network**

A group of computers, terminals, and other devices and the hardware and software that enable them to exchange data and share resources over short or long distances. A network can consist of any combination of local area networks (LAN) or wide area networks (WAN).

**Off-hook**

A line condition caused when a telephone handset is removed from its cradle.

**On-hook**

A line condition caused when a telephone handset is resting in its cradle.

**Packet**

Includes three principal elements: control information (such as destination, origin, length of packet), data to be transmitted, and error detection. The structure of a packet depends on the protocol.

**Plain Old Telephone System (POTS)**

Standard telephone service used by most residential locations; basic service supplying standard single line telephones, telephone lines, and access to the public switched network.

**Point to Point Protocol over Ethernet (PPPoE)**

A proposal specifying how a host personal computer interacts with a broadband modem (i.e., DSL, cable, wireless, etc.) to access the growing number of Highspeed data networks. Relying on two widely accepted standards, Ethernet and the point-to-point protocol (PPP), the PPPoE implementation requires virtually no more knowledge on the part of the end user other than that required for standard Dialup Internet access. In addition, PPPoE requires no major changes in the operational model for Internet Service Providers (ISPs) and carriers. The base protocol is defined in RFC 2516.

**Port**

Network access point, the identifier used to distinguish among multiple simultaneous connections to a host.

**Portable Operating System Interface (POSIX)**

POSIX is a set of standard operating system interfaces based on the UNIX operating system. The need for standardization arose because enterprises using computers wanted to be able to develop programs that could be moved among different manufacturer's computer systems without having to be recoded.

**POST**

The POST method is used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.

**Private Branch Exchange (PBX)**

A small to medium sized telephone system and switch that provides communications between onsite telephones and exterior communications networks.

**Programmable Read-Only Memory (PROM)**

A memory chip where data is written only once as it remains there forever. Unlike RAM, PROMs retain their contents when the computer is turned off.

**Protocol**

A formal set of rules developed by international standards bodies, LAN equipment vendors, or groups governing the format, control, and timing of network communications. A set of conventions dealing with transmissions between two systems. Typically defines how to implement a group of services in one or two layers of the OSI reference model. Protocols can describe low-level details of machine-to-machine interfaces or high-level exchanges between allocation programs.

**Proxy Server**

An intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. Requests are serviced internally or by passing them on, possibly after translation, to other servers. A proxy interprets, and, if necessary, rewrites a request message before forwarding it.

**Public Switched Telephone Network (PSTN)**

The local telephone company network that carries voice data over analog telephone lines.

**Quality of Service (QoS)**

Measure of the telephone service quality provided to a subscriber. This could be, for example, the longest time someone should wait after picking up the handset before they receive dial tone (three seconds in most U.S. states).

**Real Time Control Protocol (RTCP)**

RTCP is the control protocol designed to work in conjunction with RTP. It is standardized in RFC 1889 and 1890. In an RTP session, participants periodically send RTCP packets to convey feedback on quality of data delivery and information of membership.

**Realtime Transport Protocol (RTP)**

An IETF standard for streaming realtime multimedia over IP in packets. Supports transport of real-time data like interactive voice and video over packet switched networks.

**Registrar Server**

A server that accepts REGISTER requests. A registrar is typically co-located with a proxy or redirect server and MAY offer location services.

**Request for Comment (RFC)**

A Request for Comments (RFC) is a formal document from the IIETF that is the result of committee drafting and subsequent review by interested parties. Some RFCs are informational in nature. Of those that are intended to become Internet standards, the final version of the RFC becomes the standard and no further comments or changes are permitted. Change can occur, however, through subsequent RFCs that supersede or elaborate on all or parts of previous RFCs.

**Router**

A specialized switching device which allows customers to link different geographically dispersed local area networks and computer systems. This is achieved even though it encompasses different types of traffic under different protocols, creating a single, more efficient, enterprise-wide network.

**Switched Circuit Network (SCN)**

A communication network, such as the public switched telephone network (PSTN), in which any user may be connected to any other user through the use of message, circuit, or packet switching and control devices.

**Server**

A computer or device on a network that works in conjunction with a client to perform some operation.

**Session Description Protocol (SDP)**

Describes multimedia sessions for the purpose of session announcement, session invitation and other forms of multimedia session initiation. SDP communicates the existence of a session and conveys sufficient information to enable participation in the session. SDP is described in RFC 2327.

**Session Initiation Protocol (SIP)**

A protocol for transporting call setup, routing, authentication, and other feature messages to endpoints within the IP domain, whether those messages originate from outside the IP cloud over SCN resources or within the cloud.

**Simple Network Management Protocol (SNMP)**

A standard of network management that uses a common software agent to manage local and wide area network equipment from different vendors; part of the Transmission Control Protocol / Internet Protocol (TCP/IP) suite and defined in RFC 1157.

**Simple Network Time Protocol (SNTP)**

SNTP, which is an adaptation of the Network Time Protocol (NTP), is widely used to synchronize computer clocks in the global Internet. It provides comprehensive mechanisms to access national time and frequency dissemination services, organize the time-synchronization subnet and adjust the local clock in each participating subnet peer. In most places of the Internet of today, NTP provides accuracies of 1-50 ms, depending on the characteristics of the synchronization source and network paths.

**Stack**

A set of network protocol layers that work together. The OSI Reference Model that defines seven protocol layers is often called a stack, as is the set of TCP/IP protocols that define communication over the Internet.

**Subnet**

An efficient means of splitting packets into two fields to separate packets for local destinations from packets for remote destinations in TCP/IP networks.

**T.38**

An ITU-T Recommendation for Real-time fax over IP. T.38 addresses IP fax transmissions for IP-enabled fax devices and fax gateways, defining the translation of T.30 fax signals and Internet Fax Protocols (IFP) packets.

**Telephony**

The science of translating sound into electrical signals, transmitting them, and then converting them back into sound.

**Transmission Control Protocol/Internet Protocol (TCP/IP)**

The basic communication language or protocol of the Internet. It can also be used as a communications protocol in a private network (either an intranet or an extranet).

**Trivial File Transfer Protocol (TFTP)**

A simplified version of FTP that transfers files but does not provide password protection, directory capability, or allow transmission of multiple files with one command.

**User Datagram Protocol (UDP)**

An efficient but unreliable, connectionless protocol that is layered over IP, as is TCP. Application programs are needed to supplement the protocol to provide error processing and retransmission of data. UDP is an OSI layer 4 protocol.

**Voice Over IP (VoIP)**

The technology used to transmit voice conversations over a data network using the Internet Protocol. Such data network may be the Internet or a corporate Intranet.

**Wide Area Network (WAN)**

A large (geographically dispersed) network, usually constructed with serial lines, that covers a large geographic area. A WAN connects LANs using transmission lines provided by a common carrier.

APPENDIX F

# List of Acronyms

| | |
|---|---|
| ABNF | Augmented Backus-Naur Form |
| AC | Access Concentrator |
| ADSI | Analog Display Services Interface |
| CE | Cummunauté européenne (French) |
| CHAP | Challenge Handshake Authentication Protocol |
| CNG | Comfort Noise Generator |
| CS-ACELP | Conjugate Structure-Algebraic Code Excited Linear Prediction |
| dB | Decibel |
| DHCP | Dynamic Host Configuration Protocol |
| DNS | Domain Name Server |
| DS | Differentiated Services |
| DSCP | Differentiated Services Code Point |
| DSL | Digital Subscriber Lines |
| DTMF | Dual Tone Multi-Frequency |
| FSK | Frequency Shift Keying |
| GMT | Greenwich Mean Time |
| HTTP | Hyper Text Transfer Protocol |
| Hz | Hertz |
| IANA | Internet Assigned Numbers Authority |
| IEEE | Institute of Electrical & Electronics Engineers |
| IETF | Internet Engineering Task Force |
| IGMP | Internet Group Management Protocol |
| IP | Internet Protocol |
| IPCP | IP Control Protocol |
| ISP | Internet Service Provider |
| ITSP | Internet Telephony Service Provider |
| LAN | Local Area Network |
| LCD | Liquid Crystal Display |
| LED | Light Emitting Diode |
| MAC | Media Access Control |
| Mb/s | Megabits Per Second |
| MIB | Management Information Base |
| MTU | Maximum Transmission Unit |
| MWI | Message Waiting Indicator |
| NAT | Name Address Translation |
| OSI | Open Systems Interconnection |
| PAP | Password Authentication Protocol |
| PBX | Private Branch eXchange |
| PCM | Pulse Code Modulation |
| PIN | Personal Identification Number |
| PPP | Point to Point Protocol |

| | |
|---|---|
| PPPoE | Point-to-Point Protocol Over Ethernet |
| PSTN | Public Switched Telephone Network |
| QoS | Quality of Service |
| RFC | Request For Comment |
| RTCP | Real Time Control Protocol |
| SCN | Switched Circuit Network |
| SDP | Session Description Protocol |
| SIP | Session Initiation Protocol |
| SME | Small and Medium-sized Enterprise |
| SMI | Structure of Management Information |
| SSL | Secure Sockets Layer |
| STP | Spanning Tree Protocol |
| STUN | Simple Traversal of User Datagram Protocol (UDP) through Network Address Translation (NAT) |
| TAS | Transparent Address Sharing |
| TBRL | Terminal Balance Return Loss |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TLS | Transport Layer Security |
| TPE | Twisted-Pair Ethernet |
| UDP | User Datagram Protocol |
| UL | Underwriters Laboratories Incorporated |
| UTC | Universal Time Coordinated |
| UTP | Unshielded Twisted pair |
| VAD | Voice Activity Detection |
| VLAN | Virtual Local Area Network |
| VoIP | Voice over Internet Protocol |
| WAN | Wide Area Network |
| XML | eXtensible Markup Language |

APPENDIX G

# List of MIB Parameters

## A


analogScnGwDialEnable ..... 355
analogScnGwDtmfDuration ..... 189, 271
analogScnGwInterDigitDialDelay ..... 189, 271


## C


certificateExpirationDate ..... 221, 241
certificateName ..... 220, 240
certificateSubjectCommonName ..... 221, 240
checkTcpIpStackForSuccessfulBoot ..... 23
configFileAutoUpdateOnRestartEnable ..... 228
configFileAutoUpdatePeriod ..... 229
configFileAutoUpdatePeriodicEnable ..... 230
configFileAutoUpdateTimeRange ..... 230
configFileAutoUpdateTimeUnit ..... 229
configFileFetchingConfigSource ..... 22
configFileFetchingDhcpSiteSpecificCode ..... 163, 222
configFileFetchingFileLocation ..... 223
configFileFetchingFileName ..... 223
configFileFetchingHost ..... 163, 221
configFileFetchingPort ..... 163, 221
configFileFetchingSelectConfigSource ..... 163, 221
configFileFetchingSpecificFileName ..... 223
configFileFetchingStaticHost ..... 163, 222
configFileFetchingStaticPort ..... 163, 222
configFilePrivacyEnable ..... 226
configFilePrivacyGenericSecret ..... 225
configFilePrivacySpecificSecret ..... 225
configFileTransferPassword ..... 227, 228, 229
configFileTransferProtocol ..... 226, 227, 228, 229
configFileTransferUsername ..... 227, 228, 229
countryCustomizationToneOverride ..... 193
countryCustomizationToneTone ..... 193


## D


dataIfAnalogCedDetectionBehavior ..... 280
dataIfCedFaxToneEnable ..... 280
dataIfClearChannelCodecPreferred ..... 281
dataIfCngToneDetectionEnable ..... 279
dataIfCodecT38Enable ..... 284
dataIfCodecT38ProtectionLevel ..... 285
dataIfT38BasePort ..... 210
dataIfT38FinalFramesRedundancy ..... 285
dataIfT38NoSignalEnable ..... 285
dataIfT38NoSignalTimeout ..... 285
digitMapAllowedDigitMap ..... 324
digitMapAllowedEnable ..... 325
digitMapAllowedLineToApply ..... 325
digitMapPrefixedDigitRemovalCount ..... 324
digitMapPrependedString ..... 325
digitMapProcessDigitsWhenPressed ..... 324

digitMapRefusedDigitMap ..... 325
digitMapRefusedEnable ..... 325
digitMapRefusedLineToApply ..... 325
digitMapSuffixStringToRemove ..... 324
digitMapTimeoutCompletion ..... 326
digitMapTimeoutFirstDigit ..... 326
digitMapTimeoutInterDigit ..... 326

## E

emergencyCallUrgentGatewayDigitMap ..... 332
emergencyCallUrgentGatewayEnable ..... 332
emergencyCallUrgentGatewayTargetAddress ..... 332

## F

fxoWaitForCalleeToAnswerEnable ..... 355
fxsBlankAnonymousCallerId ..... 260
fxsCalleeHangupDelay ..... 259
fxsCalleeHangupSupervision ..... 259
fxsCallingNumberCriteria ..... 261
fxsCallingNumberTransformation ..... 261
fxsFlashHookDetectionDelayMax ..... 257
fxsFlashHookDetectionDelayMin ..... 257
fxsLoopCurrent ..... 258
fxsLoopCurrentDropEnable ..... 258
fxsPolarityAndDenialBehavior ..... 260
fxsPowerDropOnDisconnectDuration ..... 260

## G

groupAdminState ..... 227, 247, 248
groupReset ..... 23
groupSetAdmin ..... 23, 226, 227, 247, 248

## H

httpServerAccess ..... 26
httpServerAdminAccess ..... 26
httpServerAdminPort ..... 25
httpServerAdminRealm ..... 26
httpServerAdminUsername ..... 44
httpServerDefaultAdminPassword ..... 44
httpServerDefaultPassword ..... 46
httpServerEnable ..... 25
httpServerPort ..... 25
httpServerResetToDefaultAdminPwd ..... 44, 45
httpServerResetToDefaultPwd ..... 46
httpServerUsername ..... 46
httpServerUserRealm ..... 26

## I

IfAdminInitialAdminState ..... 256
ifAdminSetAdmin ..... 255
imageAutoUpdateEnable ..... 250, 251
imageAutoUpdateOnRestartEnable ..... 250
imageAutoUpdatePeriod ..... 251
imageAutoUpdateTimeRange ..... 251
imageAutoUpdateTimeUnit ..... 250
imageConfigSource ..... 21
imageDhcpPrimarySiteSpecificCode ..... 161, 241

imageDhcpSecondarySiteSpecificCode ..... 162, 242
imageLocation ..... 243, 244
imageLocationProvisionSource ..... 244
imagePrimaryHost ..... 21, 161, 241
imagePrimaryPort ..... 21, 161, 241
imageSecondaryHost ..... 21, 162, 241
imageSecondaryPort ..... 21, 162, 241
imageSelectConfigSource ..... 161, 241
imageSelectionFileLocation ..... 243, 244
imageStaticPrimaryHost ..... 242
imageStaticPrimaryPort ..... 242
imageStaticSecondaryHost ..... 242
imageStaticSecondaryPort ..... 242
imageTransferPassword ..... 248, 250
imageTransferProtocol ..... 247, 248, 249, 250
imageTransferUsername ..... 248, 249, 250
ipAddressConfigLanInterface ..... 206
ipRoutingBandwidthControlEnable ..... 209, 216
ipRoutingDhcpServerLeaseTime ..... 209
ipRoutingEnable ..... 209, 216
ipRoutingMacAddress ..... 208
ipRoutingMacSpoofAddress ..... 207
ipRoutingMacSpoofEnable ..... 208
ipRoutingQosDiffServSubstitution ..... 206
ipRoutingQosDiffServSubstitutionEnable ..... 206
ipRoutingWanUpstreamBandwidth ..... 209, 216

## L

lanStaticAddress ..... 159, 166
lanStaticNetworkMask ..... 159, 166, 207
lineSelectionDigitMap ..... 257
lineSelectionEnable ..... 257
localHostAddress ..... 21, 158, 159, 204
localHostConfigSource ..... 21
localHostDefaultRouter ..... 21, 158
localHostDhcpServer ..... 158
localHostDnsOverrideEnable ..... 161
localHostNetworkMask ..... 207
localHostPrimaryDns ..... 21, 158
localHostSecondaryDns ..... 21, 158
localHostSelectConfigSource ..... 15, 158, 202
LocalHostSnmpPort ..... 160
localHostSnmpPort ..... 21, 160
localHostStaticDefaultRouter ..... 202
localHostStaticPrimaryDns ..... 15
localHostStaticSecondaryDns ..... 15
localHostStaticSnmpPort ..... 160
localHostStaticSubnetMask ..... 15
localHostStaticWanAddress ..... 159, 293
localHostSubnetMask ..... 21, 158
LocalHostWanAddress ..... 159
localHostWanAddress ..... 204
localHostWanAddressSelectConfigSource ..... 159, 204, 293

## M

msConfigSource ..... 22
msDhcpSiteSpecificCode ..... 162, 364
msEnable ..... 363
msHost ..... 21, 162, 363

msSelectConfigSource ..... 162, 363
msStaticHost ..... 162, 364
msStaticPort ..... 162
msStaticTrapPort ..... 162, 364
msTrapConfigInformation ..... 233
msTrapPort ..... 21, 162, 363
msTrapRetransmissionPeriod ..... 363
msTrapRetransmissionRetryCount ..... 363
mwiConfigActivation ..... 360
mwiConfigFetchAddress ..... 360
mwiConfigUserSubscriptionAddress ..... 359
mwiExpirationTime ..... 360
mwiFetchDigitMap ..... 360
mwiSubscriptionCmdRefresh ..... 360
mxDebugPcmCaptureEnable ..... 384
mxDebugPcmCaptureEndpointNumber ..... 384
mxDebugPcmCaptureIpAddress ..... 384
mxInteropHttpUAHeaderConfig ..... 227, 249

## P

pinDialingDelay ..... 354
pinDialingEnable ..... 354
pinDialingPin ..... 354
pppoeAcName ..... 202
pppoeEnable ..... 203
pppoeServiceName ..... 203
pppSecuritySecretsIdentity ..... 203
pppSecuritySecretsSecret ..... 203

## Q

qosInteropUseVoiceQoSForRtcpEnable ..... 367
qosSignalingDiffServ ..... 366
qosSignalingIeee8021qEnable ..... 366
qosSignalingIeee8021qUserPriority ..... 366
qosT38FaxDiffServ ..... 366
qosT38FaxIeee8021qEnable ..... 217, 366
qosT38FaxIeee8021qUserPriority ..... 216, 366
qosVlanIeee8021qDefaultUserPriority ..... 368
qosVlanIeee8021qSubstitutionEnable ..... 369, 370
qosVlanIeee8021qSubstitutionFiltering ..... 370
qosVlanIeee8021qSubstitutionUserPriority ..... 369
qosVlanIeee8021qSubstitutionVlanID ..... 369
qosVlanIeee8021qTaggingEnable ..... 368
qosVlanIeee8021qVirtualLanID ..... 368
qosVoiceDiffServ ..... 366
qosVoiceIeee8021qEnable ..... 217, 366
qosVoiceIeee8021qUserPriority ..... 216, 366

## R

rtpConfigBasePort ..... 211

## S

sipAllowAudioAndImageNegotiationEnable ..... 313
sipAllowMediaReactivationInAnswerEnable ..... 312
sipCodecOrderInAnswer ..... 313
sipConferenceServerURI ..... 348
sipConferenceType ..... 348

sipDebugContextSnapshotTime ..... 384
sipDomain ..... 287
sipEnforceOfferAnswerModel ..... 312
sipHomeDomainProxyDhcpSiteSpecificCode ..... 164, 179
sipHomeDomainProxyHost ..... 164, 179
sipHomeDomainProxyPort ..... 164, 179
sipHomeDomainProxyStaticHost ..... 180
sipHomeDomainProxyStaticPort ..... 180, 186
sipInteropAckUnsupportedInfoRequests ..... 307
sipInteropAllowAsymmetricDtmfPayloadType ..... 310
sipInteropAllowMultipleActiveMediaInAnswer ..... 304
sipInteropAuthenticationQop ..... 291
sipInteropAutomaticRejectionCode ..... 357
sipInteropBehaviorOnT38InviteRejectedWith606 ..... 286
sipInteropBranchMatchingMethod ..... 305
sipInteropCallWaitingToneControlViaSipInfo ..... 310
sipInteropConferenceServerMechanism ..... 348
sipInteropDefaultPublicationExpiration ..... 298
sipInteropDefaultRegistrationExpiration ..... 297
sipInteropDtmfTransportDuration ..... 271
sipInteropDtmfTransportMethod ..... 271
sipInteropEscapePoundInSipUriUsername ..... 311
sipInteropIgnoreUsernameParam ..... 311
sipInteropIgnoreViaBranchIdInCancelEnable ..... 306
sipInteropInternationalCodeMappingEnable ..... 307
sipInteropInternationalCodeMappingString ..... 307
sipInteropLocalRingOnProvisionalResponse ..... 305
sipInteropLockDnsSrvRecordPerCallEnable ..... 187
sipInteropMaxForwardsValue ..... 301
sipInteropMwiMessageSummaryValidation ..... 360
sipInteropOnHoldAnswerSdpStreamDirection ..... 304
sipInteropOnHoldSdpStreamDirection ..... 303
sipInteropProxyAuthenticationUriParametersEnable ..... 307
sipInteropReferredByConfig ..... 302
sipInteropRegisterHomeDomainHostOverride ..... 288
sipInteropRemoveOutboundProxyRouteHeader ..... 307
sipInteropReplacesConfig ..... 300
sipInteropReplacesVersion ..... 301
sipInteropRetryFailedRegistration ..... 308
sipInteropReuseCredentialEnable ..... 187, 305
sipInteropRingingResponseCode ..... 306
sipInteropSdpDirectionAttributeEnable ..... 302
sipInteropSendUAHeaderEnable ..... 288
sipInteropSessionTimersVersion ..... 289
sipInteropSipOptionsMethodSupport ..... 311
sipInteropSymmetricUdpSourcePortEnable ..... 294
sipInteropT38NoSignalBehavior ..... 285
sipInteropTransferVersion ..... 299
sipInteropTransmissionTimeout ..... 187, 301
sipInteropUAHeaderConfig ..... 288
sipInteropUseDtmfPayloadTypeFoundInAnswer ..... 310
sipInteropUseItuT38Format ..... 308
sipInteropUsePAssertedHeader ..... 309
sipInteropUseSipDomainInRequestURI ..... 308
sipOutboundProxyConfig ..... 182
sipOutboundProxyDhcpSiteSpecificCode ..... 164, 181
sipOutboundProxyHost ..... 164, 181
sipOutboundProxyPort ..... 164, 181
sipOutboundProxyStaticHost ..... 182
sipOutboundProxyStaticPort ..... 182, 186

sipPenaltyBoxEnable ..... 296
sipPenaltyBoxTime ..... 187, 295
sipPort ..... 287
sipPresenceCompositorDhcpSiteSpecificCode ..... 184
sipPresenceCompositorHost ..... 165, 183
sipPresenceCompositorPort ..... 165, 183
sipPresenceCompositorStaticHost ..... 184
sipPresenceCompositorStaticPort ..... 184
sipPublicationCmdRefresh ..... 298
sipPublicationProposedExpirationValue ..... 298
sipPublicationRefreshTime ..... 298
sipRegistrarDhcpSiteSpecificCode ..... 164, 178
sipRegistrarHost ..... 164, 177
sipRegistrarPort ..... 164, 177
sipRegistrarStaticHost ..... 178
sipRegistrarStaticPort ..... 178, 186
sipRegistrationCmdRefresh ..... 296
sipRegistrationProposedExpirationValue ..... 297
sipReRegistrationTime ..... 296
sipServerSelectConfigSource ..... 164, 177, 179, 181, 183
sipTransportContactEnable ..... 294
sipTransportEnable ..... 293
sipTransportQValue ..... 293
sipTransportRegistrationEnable ..... 293
sipTrustedSourcesEnable ..... 292
sipTrustedSourcesIpAddress ..... 292
sipUAAuthPassword ..... 291
sipUAAuthRealm ..... 291
sipUAAuthUsername ..... 291
sipUAAuthValidateRealm ..... 291
sipUADisplayName ..... 288
sipUAMainUsername ..... 287
sipUAMaximumSessionExpirationDelay ..... 289
sipUAMinimumSessionExpirationDelay ..... 289
sipUAOtherAcceptedUsernames ..... 288
sipUnitAuthPassword ..... 291
sipUnitAuthRealm ..... 291
sipUnitAuthUsername ..... 291
sipUnitAuthValidateRealm ..... 291
sipUnregisteredPortBehavior ..... 256, 385
snmpAgentAccess ..... 152
sntpConfigSource ..... 22
sntpEnable ..... 317
sntpHost ..... 165, 318
sntpPort ..... 165, 318
sntpSelectConfigSource ..... 165, 318
sntpStaticHost ..... 318
sntpStaticPort ..... 318
sntpSynchronizationPeriod ..... 317
sntpSynchronizationPeriodOnError ..... 317
sntpTimeZoneString ..... 230, 251, 319
stunEnable ..... 316
stunKeepAliveInterval ..... 316
stunQueryCacheDuration ..... 316
stunQueryTimeout ..... 316
stunStaticHost ..... 316
stunStaticPort ..... 316
subscriberServicesAttendedTransferEnable ..... 346
subscriberServicesAttendedTransferStatus ..... 346
subscriberServicesBlindTransferEnable ..... 345

subscriberServicesBlindTransferStatus ..... 345
subscriberServicesCallForwardOnBusyActivation ..... 338
subscriberServicesCallForwardOnBusyDisableDigitMap ..... 338
subscriberServicesCallForwardOnBusyEnable ..... 338
subscriberServicesCallForwardOnBusyEnableDigitMap ..... 338
subscriberServicesCallForwardOnBusyForwardingAddress ..... 338
subscriberServicesCallForwardOnNoAnswerActivation ..... 340
subscriberServicesCallForwardOnNoAnswerDisableDigitMap ..... 340
subscriberServicesCallForwardOnNoAnswerEnable ..... 340
subscriberServicesCallForwardOnNoAnswerEnableDigitMap ..... 340
subscriberServicesCallForwardOnNoAnswerForwardingAddress ..... 340
subscriberServicesCallForwardOnNoAnswerTimeout ..... 340
subscriberServicesCallForwardUnconditionalActivation ..... 336
subscriberServicesCallForwardUnconditionalDisableDigitMap ..... 336
subscriberServicesCallForwardUnconditionalEnable ..... 337
subscriberServicesCallForwardUnconditionalEnableDigitMap ..... 336
subscriberServicesCallForwardUnconditionalForwardingAddress ..... 336
subscriberServicesCallWaitingCancelDigitMap ..... 342
subscriberServicesCallWaitingEnable ..... 342
subscriberServicesCallWaitingPermanentDigitMapDisable ..... 343
subscriberServicesCallWaitingPermanentDigitMapEnable ..... 342
subscriberServicesCallWaitingStatus ..... 342
subscriberServicesConferenceEnable ..... 348
subscriberServicesConferenceStatus ..... 348
subscriberServicesHoldEnable ..... 335
subscriberServicesHoldStatus ..... 335
subscriberServicesProcessingTrigger ..... 333
subscriberServicesSecondCallEnable ..... 335
subscriberServicesSecondCallStatus ..... 335
sysAdminCommand ..... 157, 247, 248
sysAdminDefaultSettingsEnable ..... 23
sysAdminDownloadConfigFileStatus ..... 224, 233
sysConfigBootpFlags ..... 168
sysConfigCommand ..... 226, 227
sysConfigComputerEthernetSpeed ..... 176
sysConfigDhcpWait ..... 167
sysConfigDhcpWaitDelay ..... 167
sysConfigDownloadConfigFile ..... 228, 230, 233
sysConfigDownloadConfigMode ..... 233
sysConfigMaxDynamicPort ..... 210
sysConfigMinDynamicPort ..... 210
sysConfigNetworkEthernetSpeed ..... 176
sysConfigProductNamePadding ..... 168
sysConfigStatsBySyslogEnable ..... 376
sysConfigStatsNumberPeriods ..... 376
sysConfigStatsPeriodLength ..... 375
syslogConfigSource ..... 22
syslogDhcpSiteSpecificCode ..... 163, 372
syslogHost ..... 21, 163, 372
syslogMsgDisplayLocalHost ..... 373
syslogMsgDisplayMacAddress ..... 373
syslogMsgDisplayTime ..... 373
syslogMsgLocalMaxNbr ..... 374
syslogMsgLocalMaxSeverity ..... 374
syslogMsgMaxSeverity ..... 371, 384
syslogPort ..... 21, 163, 372
syslogSelectConfigSource ..... 163, 372
syslogStaticHost ..... 372
syslogStaticPort ..... 373
sysMacAddress ..... 9, 168

sysMibVersion ..... 152

## T

telephonyAttributesAutomaticCallEnable ..... 351
telephonyAttributesAutomaticCallTargetAddress ..... 351
telephonyAttributesAutomaticRejection ..... 357
telephonyAttributesCallDirectionRestriction ..... 352
telephonyAttributesDelayedHotLineEnable ..... 356
telephonyAttributesDelayedHotLineExtension ..... 357
telephonyAttributesDelayedHotLineTargetAddress ..... 357
telephonyAttributesHookFlashProcessing ..... 352, 356
telephonyAttributesIpAddressCallEnable ..... 353
telephonyCountrySelection ..... 191
telephonyDnsOverrideEnable ..... 161

## V

voiceIfAdaptativeJitterBufferEnable ..... 383
voiceIfCodecG72616kbpsEnable ..... 265
voiceIfCodecG72616kbpsMaxPTime ..... 267
voiceIfCodecG72616kbpsMinPTime ..... 267
voiceIfCodecG72616kbpsPayloadType ..... 265
voiceIfCodecG72624kbpsEnable ..... 265
voiceIfCodecG72624kbpsMaxPTime ..... 267
voiceIfCodecG72624kbpsMinPTime ..... 267
voiceIfCodecG72624kbpsPayloadType ..... 265
voiceIfCodecG72632kbpsEnable ..... 265
voiceIfCodecG72632kbpsMaxPTime ..... 267
voiceIfCodecG72632kbpsMinPTime ..... 267
voiceIfCodecG72632kbpsPayloadType ..... 265
voiceIfCodecG72640kbpsEnable ..... 265
voiceIfCodecG72640kbpsMaxPTime ..... 267
voiceIfCodecG72640kbpsMinPTime ..... 267
voiceIfCodecG72640kbpsPayloadType ..... 265
voiceIfCodecG729Enable ..... 265
voiceIfCodecG729MaxPTime ..... 268
voiceIfCodecG729MinPTime ..... 268
voiceIfCodecPcmaEnable ..... 265
voiceIfCodecPcmaMaxPTime ..... 266
voiceIfCodecPcmaMinPTime ..... 266
voiceIfCodecPcmuEnable ..... 265
voiceIfCodecPcmuMaxPTime ..... 266
voiceIfCodecPcmuMinPTime ..... 266
voiceIfCodecPreferred ..... 265
voiceIfDtmfDetectionRiseTimeCriteria ..... 274
voiceIfDtmfDetectionUnitBreakPowerThreshold ..... 273
voiceIfDtmfDetectionUnitMaxPowerThreshold ..... 273
voiceIfDtmfDetectionUnitMinPowerThreshold ..... 273
voiceIfDtmfDetectionUnitNegativeTwist ..... 273
voiceIfDtmfDetectionUnitPositiveTwist ..... 273
voiceIfDtmfEnforceDefaultEvents ..... 270
voiceIfDtmfPayloadType ..... 270
voiceIfDtmfTransport ..... 266, 268, 269, 270, 271
voiceIfEchoCancellationEnable ..... 276
voiceIfG711ComfortNoiseGenerationEnable ..... 277
voiceIfG711VoiceActivityDetectionEnable ..... 275
voiceIfG729VoiceActivityDetectionEnable ..... 276
voiceIfMaxJitterBufferLength ..... 274, 383
voiceIfSignalLimiterLevel ..... 277
voiceIfTargetJitterBufferLength ..... 274, 383

voiceIfUserInputGainOffset ..... 278
voiceIfUserOutputGainOffset ..... 278

# Index

## Symbols

'+' character substitution in caller ID 307

## Numerics

10 BaseT 7, 9, 11, 12, 49, 176
    defined 445
    see also *cabling*
100 BaseT 7, 9, 11, 12, 49, 176
    defined 445
    see also *cabling*
802.1q, in QoS 129, 366

## A

access concentrator, defined 445
acronyms 453
ADSI
    caller ID 190
    defined 445
A-Law 93, 263
    defined 445
allow multiple active media in answer, in SIP 304
analog modem, feature 93, 263
area code, defined 445
audience, intended xix
authentication information 85, 290
    request protection 291
auto MDI/MDIX 9
automatic
    call 116, 351
    configuration update 60, 228
    software update 72, 249

## B

bandwidth management, WAN upstream control 208
Bootp BROADCAST flag, in DHCP requests 167
branch matching method, in SIP 305
branch, behaviour of Via in SIP 306
broadcast storm, behaviour when restarting 23

## C

cabling
    RJ-11 409
    RJ-45
        crossover 408
        pin name 408
        pinout information 407
        straight through 407
    see also *10 BaseT*
    see also *100 BaseT*
call
    automatic 116, 351
    dialing sequence 331
    emergency, enabling 132, 332
    forced SCN 332
    forward
        on busy 104, 338
        on no answer 106, 340
        unconditional 107, 336
call (continued)
    hold 115, 334
        direction attributes 303
    IP address 353
    placing 4
    putting on hold 112, 116, 335, 343
    restriction on direction 352
    second 116, 335
    standard 331
    transfer
        attended 109, 346
        blind 109, 345
    waiting 110, 342
call forward
    on busy 105, 339
    on no answer 107, 341
    unconditional 108, 337
call rejection 357
call router, regular expression 261
call transfer
    attended 110, 346
    blind 110, 345
call waiting
    controlling via SIP INFO 310
    disabling 112, 343
    enabling 112, 343
    using 112, 343
callee hangup supervision 259
caller ID
    ADSI 190
    blanking anonymous 260
    country-specific, selecting 118, 191
    DTMF signalling 189
        ETSI 300 659-1 January 2001 (Annex B) 189
        STD 220-250-713 Issue 01. November 1993 189
        TDK-TS 900 301-1 January 2003 189
    FSK generation 190
        Bellcore GR-30-CORE 190
        British Telecom (BT) SIN227, SIN242 190
        ETSI 300 659-1 190
        UK CCA specification TW/P&E/312 190
    generation 189
CED tone, analog detection behaviour 280
CED tone, detection 280
clear channel fax
    enabling 101, 281
    preferred codec 103, 281
    setting 279
CNG tone, detection 279
codec
    data
        clear channel fax 279
            enabling 101, 281
            preferred 103, 281
        enabling 279
        T.38 101, 284, 285, 286
    voice
        defined 263
        DTMF detection 272
        DTMF transport type 96, 97, 100, 268, 269, 270
        DTMF transport type over the SIP protocol 271
        echo cancellation 97
        enabling 98, 265

codec (continued)
voice
packetization time 98, 266
preferred 95, 265
comfort noise 98, 277
compliance to standards. see *standards compliance*
configuration
file 152
using a GUI 149
web interface
configuration file download 54
configuration file upload 39
enabling 25
LAN 37
LAN interface 37
MAC address spoofing 37
Monitoring 32
password
modify 40, 43, 45
reset 40, 44, 46
PPP password 36
PPP user name 36
SIP custom NAT traversal 135
static information 36
STUN 38, 134
system log 42
user name, modify 40, 44, 46
WAN 35
WAN connection type 36
web interface. see *web interface*
configuration file
automatic update 60, 228
download server 54, 219
configuration source 55, 221
DHCP information, using 56, 221
HTTP server, configuring 54, 219
HTTPS server, configuring 220
IP address 56, 221
SNTP server, configuring 54, 219
static information, using 55, 222
TFTP server, configuring 54, 219
download, setting 56, 57, 222
encryption
decrypting generic 59, 225
decrypting specific 59, 225
defined 59, 225
example 236
HTTP, downloading via 60, 227
HTTPS, downloading via 227
management server, downloading from 233
syslog messages 58, 224
TFTP, downloading via 60, 226
uploading via web interface 39
user agent header, customizing, user agent header, customizing 227
web interface, configuring via 54
configuration sources, setting all to static 157
configuring the software
configuration file 152
DHCP (dynamic), using 157
MIB. see *SNMP*
static
setting configuration sources to 157
using 157
connecting the unit 9
connectors
LAN 7
Phone-Fax 1 7
Phone-Fax 2 7
Reset/Default 7
universal power supply unit 7
WAN 7
country-specific parameters 411
caller ID, selecting 118, 191
setting 117, 190
credential, in SIP 305
crossover cable. see *cabling*
custom tone configuration
SNMP, configuring via 192
web interface, configuring via 118
customer services xxiv

## D

default router, setting 36, 51, 158
Default Settings
factory reset procedure 22
disabling 23
in recovery mode 21
delayed hot line 356
DHCP information
Bootp BROADCAST flag 167
configuration file download server 56, 221
Image 67, 241
management server 363
network settings 50
options, waiting time to receive 167
SIP outbound proxy 78, 181
SIP Presence Compositor server 183
SIP proxy server 78, 179
SIP registrar server 78, 177
size of DHCP request 168
SNTP 51, 318
syslog daemon 32, 372
see also *static information*
DHCP server
configuring 168
defined 446
FQDN, entering 172
IP address of, setting 158
IP addresses, entering 171
network configuration 169
requirement 3
site specific option 170
vendor class ID 169
vendor specific option 169
Dial Map. see *digit map*
dialing
forced SCN call 332
IP address call 353
sequence 331
settings
DTMF duration value 271
inter-digit dial delay 271
standard call 331
telephone number 331
Differentiated Services (DS) Field, in QoS 365
substituting configured value 206
digit map
# and * characters 89, 323
combining two expressions 88, 322

digit map (continued)
definition 87, 321
examples
PBX emulation 328
standard calls 326
processing digits behaviour 324
refused 325
rules 90, 324
special characters 88, 322
timeouts 326
timer 89, 323
using 88, 322
validating 89, 323
web interface, configuring via 87
direction attribute "sendonly" 303
direction attributes, in SIP 302, 303
disabling lines 255
distinctive ring 413
DNS
defined 446
primary 36, 51, 158
static 160
requirement 3
secondary 36, 51, 158
static 160
DNS SRV
call flow 186
defined 185
enabling 186
record lock, in SIP 187
documentation
Media5 download portal xxiii
Mediatrix download portal xxiii
downgrading software, procedure 75, 252
downloading software
automatic update 72, 249
configuration source 66, 241
emergency download 76, 252
HTTP server, configuring 66, 239
HTTP, via 72, 248
HTTPS server, configuring 240
HTTPS, via 248
Image path 68, 243
LED states 71, 246
SNTP server, configuring 66, 239
Spanning Tree Protocol 75, 251
TFTP server, configuring 66, 239
TFTP, via 72, 247
troubleshooting 388
zip file 67, 243
DTMF
defined 446
detection 272
duration value 271
out-of-band 96, 268
RFC 2833 events 270
signalling, caller ID 189
transport type 96, 100, 268
over the SIP protocol 271
payload type 97, 270
using SIP INFO method 269
web interface, configuring via 96, 100

## E

echo cancellation 97, 276
signal limiter 277
emergency call, enabling 132, 332
emergency software download 76, 252
enabling lines 255
encryption, of configuration files
decrypt generic 59, 225
decrypt specific 59, 225
defined 59, 225
end user technical support xxiv
Ethernet connection
setting speed of 49, 176
web interface, configuring via 49

## F

factory reset
disabling 23
reverting to 22
see also *recovery mode*
far end disconnect, signalling 258
fax
analog CED detection behaviour 280
call waiting tone, disabling 112
call waiting tone, disabling permanently 112, 344
call waiting tone, disabling temporarily 343
calling tone detection, enabling 279
CED tone detection, enabling 280
clear channel 279
T.38 101, 284
INVITE rejected with 606 286
no-signal 285
user gain vs communication quality 278
firmware download. see *software*
flash hook, setting 256
Foreign Exchange Service/Station (FXS)
defined 447
see also *lines*
FQDN, entering 172
FSK generation, caller ID 190

## G

g a cable modem 202
G.711 93, 263
comfort noise 98, 277
defined 447
enabling 98
packetization time 98
voice activity detection 98, 275
G.726 94, 264
comfort noise 277
defined 447
enabling 99
packetization time 100
payload type 100, 265
voice activity detection 275
G.729 94, 264
defined 447
enabling 99
packetization time 99
voice activity detection 99, 276
GUI, using a 149

# H


hardware
- cleaning 9
- condensation 9
- front indicators 6
- proper location 8
- rear connections 7

header, SIP user agent
- customizing 227, 288
- sending 288

hold, putting a call on 112, 115, 116, 334, 335, 343
- direction attributes 303

home domain proxy override 79, 287
hook flash processing 352
HTTP
- configuration file download 60, 227
- server
  - configuring 54, 66, 219, 239
  - requirement 3
- software download via 72, 248

HTTPS
- configuration file download 227
- server, configuring 220, 240
- software download via 248

humidity level 8


# I


IEEE 802.1q, in QoS 129, 366
IGMP, in router service 201
ignore username parameter, in SIP 311
Image server
- DHCP information, using 67, 241
- static information, using 67, 242

indicators of the hardware 6
installation
- before proceeding 9
- connecting the hardware 9
- free standing unit 8
- package contents 3
- provisioning sequence in DHCP 14
- provisioning sequence in PPPoE or PPPoA 15
- requirements 3
- reserving IP address 9
- router, with a 11
- safety recommendations 3
- selecting site for 7
- setting up the unit for the first time 13
- single computer, with a 10
- verifying 24
- wall-mounting 8

intended audience xix
inter-digit dial delay 271
international code mapping 307
INVITE rejected with 606, in T.38 286
IP address
- default router 36, 51, 158
- defining
  - decimal 155
  - hexadecimal 155
  - octal 155
- DHCP server 158
- DHCP, using 157
- DNS, primary 36, 51, 158
  - static 160

IP address (continued)
- DNS, secondary 36, 51, 158
  - static 160
- download server 56, 221
- entering 171
- Image server 66, 241
- LAN connector, static 166
- locating 155
- Management Server 363
- of unit 36, 50, 158
- SIP outbound proxy 79, 181
- SIP Presence Compositor server 79
- SIP proxy server 79, 179, 183
- SIP registrar server 79, 177
- SNTP server 318
- static
  - setting configuration sources to 157
  - using 157
- subnet mask 36, 51, 158
- syslog 32
- syslog daemon 372
- vocal identification of 16
  - WAN 16
- WAN 204

IP address call 353
IP address, dialing 353


# J


jitter
- buffer protection 96, 274
- defined 447
- web interface, configuring via 96


# L


LAN
- cable 24
- configuration via web interface 37
- defined 448
- interface, configuring in transparent address sharing 206
  - via web interface 37

LEDs
- behaviour
  - in download mode 71, 246
  - in starting mode 16
- defined 448
- ETH 17
- In Use 16
- patterns
  - AdminMode 17, 19
  - Booting 17, 18
  - DefaultSettings ending 17
  - DiagFailed 18
  - ImageDownloadError 18
  - ImageDownloadInProgress 17
  - InitFailed 18
  - NormalMode 17, 19
  - RebootPending 17
  - RecoveryMode 17, 19
  - RecoveryModePending 17
  - ResetPending 17
- Power 17
- Ready 16
- states 16

line mapping 257

lines
anonymous caller ID, blanking of 260
callee hangup supervision 259
comfort noise 98, 277
data codecs 100, 279
analog CED detection behaviour 280
CED fax tone detection 280
clear channel fax 279
clear channel, enabling 101, 281
clear channel, preferred 103, 281
enabling 279
fax calling tone detection 279
T.38 101, 284, 285, 286
echo cancellation 276
signal limiter 277
far end disconnect, signalling 258
jitter buffer protection 96, 274
locking/unlocking 255
reversal 260
source selection 257
FXS to FXO line mapping 257
reserving FXS line 258
unregistered, behaviour when 78, 256
user gain 278
voice activity detection 98, 275
voice codecs
defined 263
DTMF detection 272
DTMF transport type 96, 97, 100, 268, 269, 270, 271
echo cancellation 97
enabling 98, 265
packetization time 98, 266
preferred codec 95, 265
local
host, in customized syslog messages 373
IP address, setting 36, 50, 158
ring behaviour, in SIP 304
time, in customized syslog messages 373
location
caller ID, selecting 118, 191
country, setting 117, 190
web interface, configuring via 117
locking lines 255
loop current, setting 258

## M

MAC address 9
defined 448
in customized syslog messages 373
spoofing 207
via web interface 37
vocal identification of 16
making
forced SCN call 332
IP address call 353
standard call 331
Management Server
defined 448
DHCP information, using 363
in configuration file download 233
static information, using 364
using 363
Max-Forwards header, in SIP 301
MDI/MDIX, auto 9
Media5 download portal xxiii
Mediatrix download portal xxiii
message waiting indicator
defined 125, 359
notifications 360
Notify service 127, 361
refresh subscription 360
setting up 126, 359
web interface, configuring via 125
MIB
defined 139
in SNMP protocol 139
see also *parameters*
MIB structure
changing a parameter value 149
conformance 148
description
mediatrixAdmin 147
mediatrixConfig 147
mediatrixExperimental 147
mediatrixIpTelephonySignaling 147
mediatrixMgmt 147
mediatrixModules 147
mediatrixProducts 147
events 148
introduction 139
objects 148
OID, defined 139
persistent parameters 149
SMI, defined 139
SNMP messages. see *SNMP*
tables
defined 150
generic 150
groupAdmin 151
ifAdmin 151
textual conventions 148
volatile parameters 149
mounting, on a wall 8
MTU, requirements 379
Mu (µ)-Law 93, 263
defined 448
multicast, in router service 201

## N

NAT
in configuration file download 60, 226
traversal, setting IP address of 292
see also *STUN*
network settings
DHCP information, using 50
static information, using 50
no-signal, in T.38 285

## O

Offer/Answer model 312
allow audio and image negotiation 313
allow media reactivation in answer 312
codec order in answer 313
OID, defined 139
operating temperature 8
outbound proxy usage with Route header 307
out-of-band DTMF 96, 268
overview of the product 4

## P


package contents 3
packetization time, setting for voice codecs 98, 266
parameters
  changing value of 149
  using a GUI 149
payload type
  asymmetric 310
  using the one found in answer 309
PCM traces, enabling 383
penalty box, in SIP 81, 295
persistent parameters, defined 149
PIN dialing, defined 354
placing a call 4
port number
  configuration file fetching 163
  configuration file server 56, 221
  DHCP setting 171
  image server 66, 161, 241
  management server 162, 363
  RTCP, setting range of 97, 211
  RTP, setting range of 97, 211
  SIP 78, 80, 287
  SIP outbound proxy 79, 164, 180
  SIP Presence Compositor 79, 183
  SIP proxy 79, 164, 179
  SIP registrar 79, 164, 177
  SNMP agent 160
  SNMP trap 160
  syslog 32, 163, 372
  T.38, setting range of 210
  TCP, setting range of 210
  UDP, setting range of 210
PPPoA service
  password, setting 36
  user name, setting 36
PPPoE service
  connection
    authentication phase 213
    discovery phase 213
    network-layer protocol phase 213
  defined 202
  enabling 202
  error handling 214
  identity 203
  password, setting 36, 203
  RFCs supported 202
  secret 203
  setting 202
  user name, setting 36, 203
primary DNS, setting 36, 51
product overview 4
provisioning
  configuration file 152
  MIB files 152
publications expiration, in SIP 298
publications refresh, in SIP 297
  via web interface 81


## Q


QoS
  802.1q 129, 366
  defined 449
QoS (continued)
  Differentiated Services (DS) Field 365
    substituting configured value 206
  VLAN 368
    substitution values 369
  voice QoS vs RTCP packets 367
  web interface, configuring via 129


## R


rear connections 7
recovery mode
  LED patterns 19
  resetting in 21
  see also *factory reset*
Referred-By field, in SIP 302
registration expiration default, in SIP 297, 298
registration expiration, in SIP 296
registration refresh, in SIP 296
  via web interface 80
regular expression, defined 261
related documentation xix
remote line extension 355
Replaces header, in SIP
  configuration 300
  version 300
requirements 3
restart
  behaviour in case of broadcast storm 23
  software-initiated 23
  unit 31
reversal, of a line 260
RFC
  RFC 1027 199
  RFC 1332 202
  RFC 1334 202, 401
  RFC 1350 401
  RFC 1362 401
  RFC 1471 202, 401
  RFC 1472 202, 401
  RFC 1473 202, 401
  RFC 1542 167
  RFC 1661 202, 401
  RFC 1769 51, 317
  RFC 1877 202, 401
  RFC 1889 401
  RFC 1890 95, 270, 401
  RFC 1945 401
  RFC 1994 202, 401
  RFC 2131 168, 212, 401
  RFC 2132 168, 169, 170, 212, 401
  RFC 2246 220, 240
  RFC 2347 401
  RFC 2348 401
  RFC 2349 401
  RFC 2459 220, 240
  RFC 2475 131, 365
  RFC 2516 202, 401
  RFC 2543 182, 302, 305
  RFC 2543bis 301
  RFC 2616 25, 401
  RFC 2617 40, 43, 45, 401
  RFC 2705 87, 321
  RFC 2782 185
  RFC 2818 220, 240
  RFC 2833 95, 268, 270, 401

RFC (continued)
RFC 2976 269, 352
RFC 3164 32, 371, 401
RFC 3261 83, 85, 182, 288, 290, 293, 297, 301, 305, 401
RFC 3263 185
RFC 3264 302, 303, 312
RFC 3280 220, 240
RFC 3389 277, 401
RFC 3489 38, 133, 315, 402
RFC 3863 77, 183, 297
RFC 3903 77, 183, 297
RFC 4579 347
ring, distinctive 413
ringing response code, in SIP 306
RJ-11. see *cabling*
RJ-45. see *cabling*
router, installing with a 11
router service
IGMP 201
multicast 201
RTCP 129, 365
base port range 97, 211
RTP, base port range 97, 211

## S

safety
recommendations 3, 400
warnings
Circuit Breaker (20A) 399
ETH, FXS1 and FXS2 Connectors 399
LAN Connector 400
No. 26 AWG 399
Product Disposal 399
Socket Outlet 400
TN Power 399
second call, service 116, 335
secondary DNS, setting 36, 51
session timer
enabling 289
session expiration delay
maximum 289
minimum 289
version supported, setting 289
signal limiter 277
signalling protocol, SIP. see *SIP, setting*
SIP INFO, controlling call waiting tone via 310
SIP, setting
allow audio and image negotiation, web interface, configuring via 84
allow media reactivation in answer, web interface, configuring via 84
allow multiple active media in answer 304
branch matching method 305
call waiting tone via SIP INFO 310
configuration 287
credential 305
direction attribute "sendonly" 303
direction attributes 303
direction attributes present 302
DNS SRV record lock 187
escape pound (#) character in SIP URI username 311
web interface, configuring via 83
failed registration attempts 308
from URI content 309
home domain in Request URI 308
home domain override 288
SIP, setting (continued)
home domain proxy override 79, 287
ignore username parameter 311
international code mapping 307
local ring behaviour 304
Max-Forwards header 301
message waiting indicator notifications 360
NAT traversal 292
network asserted caller ID 309
Offer/Answer model 312
allow audio and image negotiation 313
allow media reactivation in answer 312
codec order in answer 313
outbound proxy
DHCP information, using 78, 181
loose router status 182
static information, using 78, 181
usage with Route header 307
outbound proxy server, web interface, configuring via 78
payload type in answer, using 309
payload type, asymmetric 310
penalty box 295
web interface, configuring via 81
Presence Compositor server
DHCP information, using 183
static information, using 184
proxy server
DHCP information, using 78, 179
static information, using 78, 180
web interface, configuring via 78
publications expiration 298
publications refresh 297
web interface, configuring via 81
Referred-By field 302
registrar server
DHCP information, using 78, 177
static information, using 78, 178
web interface, configuring via 78
registration expiration 296
registration expiration, default 297, 298
registration refresh 296
web interface, configuring via 80
replaces
configuration 300
version 300
ringing response code 306
session timer 289
session expiration delay, maximum 289
session expiration delay, minimum 289
version supported 289
T.38 negotiation syntax 308
transmission timeout 301
transport type
TCP 83, 293
UDP 83, 293
web interface, configuring via 83
trusted sources 292
UDP source port behaviour 294
unsupported INFO request 307
uri-parameters 306
user agents
authentication information 290, 291
authentification information 85
display name 288
header, customizing 227, 288
header, enabling to send 288

SIP, setting (continued)
user agents
main user name 287
other accepted user names 288
setting information 80, 287
web interface, configuring via 79
Via branch behaviour 306
site specific information, DHCP setting 170
site, selecting for unit 7
SMI, defined 139
SNMP
access limitation 152
behaviour 141
non-secure management mode 142
secure management mode 142
configuring 160
custom tone configuration 192
messages 140
MIB 139
SNMP configuration file 143
versions 140
SNTP
defined 450
DHCP information, using 51, 318
enabling 51, 317
server, configuring 54, 66, 219, 239
static information, using 51, 318
time zone, defining custom 51, 319
web interface, configuring via 51
software
downgrading, procedure 75, 252
downloading
automatic update 72, 249
configuration source 66, 241
emergency procedure 76, 252
HTTP server, configuring 66, 239
HTTP, via 72, 248
HTTPS server, configuring 240
HTTPS, via 248
Image path 68, 243
LED states 71, 246
SNTP server, configuring 66, 239
Spanning Tree Protocol 75, 251
syslog messages 70, 245
TFTP server, configuring 66, 239
TFTP, via 72, 247
troubleshooting 388
user agent header, customizing, user agent header, customizing 249
zip file 67, 243
see also *downgrading*
source line selection 257
FXS to FXO line mapping 257
reserving FXS line 258
Spanning Tree Protocol 75, 251
DHCP options waiting time 167
special vocal features 16
IP address 16
IP address, WAN 16
MAC address 16
standards compliance
agency approvals 395
CE marking 396
emissions 395
FCC Part 15 disclaimer 396
immunity 395
standards compliance (continued)
RoHS China 397
safety standards 395
standards supported xxiii
caller ID
Bellcore GR-30-CORE 190
British Telecom (BT) SIN227, SIN242 190
ETSI 300 659-1 190
ETSI 300 659-1 January 2001 (Annex B) 189
STD 220-250-713 Issue 01. November 1993 189
TDK-TS 900 301-1 January 2003 189
UK CCA specification TW/P&E/312 190
draft-choudhuri-sip-info-digit-00.txt 95, 268, 269, 271, 354
draft-ietf-sip-cc-transfer-02.txt 299
draft-ietf-sip-cc-transfer-05.txt 299
draft-ietf-sipping-cc-transfer-01.txt 299
draft-ietf-sipping-mwi-01.txt 126, 359
draft-ietf-sipping-realtimefax-01.txt 101, 284
draft-ietf-sip-refer-02.txt 299
draft-ietf-sip-session-timer-04.txt 289
draft-ietf-sip-session-timer-08.txt 289
draft-mahy-sip-message-waiting-02.txt 127, 361
GR-506-CORE (Issue 1, with Revision 1, November 1996) 126, 359
ITU-T Q.24 95, 268
Recommendation ITU T.38 version 0 101, 284
Recommendation ITU-T T.38, section D.2.3 308
RFC 1027 199
RFC 1332 202
RFC 1334 202, 401
RFC 1350 401
RFC 1362 401
RFC 1471 202, 401
RFC 1472 202, 401
RFC 1473 202, 401
RFC 1542 167
RFC 1661 202, 401
RFC 1769 51, 317
RFC 1877 202, 401
RFC 1889 401
RFC 1890 95, 270, 401
RFC 1945 401
RFC 1994 202, 401
RFC 2131 168, 212, 401
RFC 2132 168, 169, 170, 212, 401
RFC 2246 220, 240
RFC 2347 401
RFC 2348 401
RFC 2349 401
RFC 2459 220, 240
RFC 2475 131, 365
RFC 2516 202, 401
RFC 2543 182, 302, 305
RFC 2543bis 301
RFC 2616 25, 401
RFC 2617 40, 43, 45, 401
RFC 2705 87, 321
RFC 2782 185
RFC 2818 220, 240
RFC 2833 95, 268, 270, 401
RFC 2976 269, 352
RFC 3164 32, 371, 401
RFC 3261 83, 85, 182, 288, 290, 293, 297, 301, 305, 401
RFC 3263 185
RFC 3264 302, 303, 312
RFC 3280 220, 240

standards supported xxiii (continued)
RFC 3389 277, 401
RFC 3489 38, 133, 315, 402
RFC 3515 299
RFC 3863 77, 183, 297
RFC 3903 77, 183, 297
RFC 4579 347
sip-replaces-01 draft 300
sip-replaces-03 draft 300
Telecordia GR-1401-CORE (Issue 1, June 2000) 126, 359
static
LAN connector IP address 166
setting DNS address as 160
static information
configuration file download server 55, 222
Image 67, 242
management server 364
network settings 50
setting all configuration sources to 157
SIP outbound proxy 78, 181
SIP Presence Compositor server 184
SIP proxy server 78, 180
SIP registrar server 78, 178
SNTP 51, 318
syslog daemon 32, 372
see also *DHCP information*
statistics
by syslog 376
resetting 376
RTP 375
setting how to collect 375
viewing 375
straight through cable. see *cabling*
STUN
configuration via web interface 38
configuring 316
defined 133, 315
web interface, configuring 134
subnet mask, setting 36, 51, 158
subscriber services
call forward
on busy 104, 338
on no answer 106, 340
unconditional 107, 336
web interface, configuring via 104
call transfer
attended transfer 109, 346
blind transfer 109, 345
web interface, configuring via 109
call waiting 110, 342
web interface, configuring via 110
conference call 113, 347
web interface, configuring via 113
hold 115, 334
direction attributes 303
web interface, configuring via 115
second call 116, 335
web interface, configuring via 116
service activation processing 333
support services xxiv
syslog
daemon
configuration via web interface 32
configuring the application 33, 373
customizing messages
local host 373
local time 373
MAC address 373
defined 32, 371
DHCP information, using 32, 372
enabling 371
messages examples 33, 371
requirement 3
static information, using 32, 372
statistics 376
local 374
messages examples 374
messages
configuration file 58, 224
examples 33, 371, 374
software download 70, 245

## T

T.38
base port range 210
defined 451
enabling 102, 284
negotiation syntax 308
not supported by other endpoint 385
number of redundancy packets 284
protection level 284
redundancy parameters 102
tables
defined 150
generic 150
groupAdmin 151
ifAdmin 151
TCP
port range 210
transport type 83, 293
technical support for end user xxiv
telephone number, dialing 331
telephony attributes
automatic call 116, 351
web interface, configuring via 116
call direction restriction 352
hook flash processing 352
IP address call 353
temperature, operating 8
textual conventions, in MIB structure 148
TFTP
configuration file download 60, 226
server
configuring 54, 66, 219, 239
defined 451
requirement 3
software donwload via 72, 247
time zone, defining custom 51, 319
TPE. see *cabling*
transfer, version supported, setting 299
transferring a call
attended transfer 110, 346
blind 110, 345
translated warning definition 398
transmission timeout, setting 301
transparent address sharing
defined 199
enabling 205
LAN interface 206
PPPoE service
connection 213

transparent address sharing (continued)
PPPoE service
enabling 202
error handling 214
password, setting 203
RFCs supported 202
setting 202
user name, setting 203
QoS substitution value 206
routing mechanism 215
RTCP base port range 211
RTP base port range 211
T.38 base port range 210
TCP port range 210
UDP port range 210
WAN IP address 204
WAN upstream bandwidth control 208
troubleshooting
call
3-way conference lost conversation 385
cannot establish to endpoint 385
cannot make 384
cannot make or receive 384
cannot disable adaptative jitter buffer 383
cannot register to IGMP services 382
configuration, configuration source does not work 388
DHCP unreachable 382
fax
poor line condition during transmission 385
specific issues 387
T.38 transmission fails 386
tested fax models 387
unable to send in clear channel 386
unable to send in T.38 386
voice does not switch back to codec after clear channel fax 386
LEDs, all off 381
long delay when starting unit 381
PC has limited web access in time 382
PCM traces, enabling 383
SNMP
cannot set a variable 390, 391
network manager cannot access unit 390
no response when trying to access unit 390
SNMPv3 variables contents 391
traps not received by network manager 390
when viewing table, unit does not respond 391
wrong value error message 390
software download
cannot communicate with image server 388
downgrade fails 389
long time to perform 389
path not recognized 389
transfer problems 389
unable to reach unit after changing Ethernet speed 382
value not accepted 383
WINS server not forwarded to the PC 382
trusted sources, in SIP 292

## U

UDP
port range 210
source port behaviour 294
transport type 83, 293
unit, restarting 31
Unit Manager Network product
as management server 3, 160, 162, 363
defined xxiv
using 65, 149, 155, 239
unlocking lines 255
unsupported INFO request, in SIP 307
uri-parameters, in SIP 306
using this manual xxii
UTP. see *cabling*

## V

vendor specific information, DHCP setting 169
verifying the installation 24
viewing statistics and performances 375
VLAN, in QoS 368
substitution values 369
vocal features, special 16
IP address 16
IP address, WAN 16
MAC address 16
voice activity detection 98, 275
volatile parameters, defined 149

## W

wall-mounting the unit 8
WAN
configuration via web interface 35
IP address, in transparent address sharing 204
upstream bandwidth control 208
web interface
access limitation 26
allow audio and image negotiation 84
allow media reactivation in answer 84
automatic call 116
call forward 104
call hold 115
call transfer 109
call waiting 110
choosing suitable web browser 26
codecs 93
conference call 113
configuration file download 54
configuration file upload 39
country 117
custom tone configuration 118
default router 51
digit map 87
DTMF transport type 96, 100
enabling 25
escape pound (#) character in SIP URI username 83
Ethernet connection speed 49
firmware download 65
group port management 48
interface management 48
jitter 96
LAN 37
LAN interface 37
MAC address spoofing 37
local IP address 50
message waiting indicator 125
Monitoring 32
password
modify 40, 43, 45
reset 40, 44, 46

web interface (continued)
primary DNS 51
QoS 129
second call 116
secondary DNS 51
SIP custom NAT traversal 135
SIP outbound proxy server 78
SIP penalty box 81
SIP proxy server 78
SIP publications refresh 81
SIP registrar server 78
SIP registration refresh 80
SIP transport type 83
SIP user agents 79
SNTP 51
status
network parameters 28
system 28
STUN 38, 134
subnet mask 51
system log 42
system management 47
user name, modify 40, 44, 46
WAN 35
connection type 36
default router 36
local IP address 36
PPP password 36
PPP user name 36
primary DNS 36
secondary DNS 36
static information 36
subnet mask 36