CONTENTS

Preface

About this Guide xxv

Conventions Used xxv

Related Documentation xxvi

MIOs and DPCs xxvi

Contacting Customer Support xxvii

CHAPTER 1

System Operation and Configuration 1

System Management Overview 1

Terminology 3

Contexts 3

Ports 3

Logical Interfaces 4

Management Interface 4

Bindings 4

Services 4

AAA Servers 5

Subscribers 5

Trusted Builds 6

How the System Selects Contexts 7

Context Selection for Context-level Administrative User Sessions 7

Context Selection for Subscriber Sessions 9

Understanding the ASR 5500 Boot Process 9

Understanding Configuration Files 11

IP Address Notation 12

IPv4 Dotted-Decimal Notation 12

IPv6 Colon-Separated-Hexadecimal Notation 13

CIDR Notation 13
Contents

CHAPTER 2

Getting Started 17

ASR 5500 Configuration 17
Using the ASR 5500 Quick Setup Wizard 17
Using the CLI for Initial Configuration 23
Configuring the System for Remote Access 25
Configuring SSH Options 27

SSH Keys 28

Setting SSH Key Size 28
Configuring SSH Key Generation Wait Time 28
Specifying SSH Encryption Ciphers 29
Generating SSH Keys 30
Setting SSH Key Pair 30

Authorized SSH User Access 30
Authorizing SSH User Access 31

SSH User Login Restrictions 31

Creating an Allowed Users List 31
SSH User Login Authentication 32

Enabling NETCONF 33

Configuring the Management Interface with a Second IP Address 33

CHAPTER 3

System Settings 35

Configuring a Second Management Interface 35

Verifying and Saving Your Interface and Port Configuration 36

Configuring System Timing 37

Setting the System Clock and Time Zone 37
Verifying and Saving Your Clock and Time Zone Configuration 37

Configuring Network Time Protocol Support 38

Configuring NTP Servers with Local Sources 39

Using a Load Balancer 39

Verifying the NTP Configuration 39

Enabling CLI Timestamping 41
Configuring CLI Confirmation Prompts 41
   Enabling Automatic Confirmation 41
   Requiring Confirmation for autoconfirm and configure Commands 42
   Requiring Confirmation for Specific Exec Mode Commands 42
Configuring System Administrative Users 43
   Configuring Context-level Administrative Users 44
      Configuring Context-level Security Administrators 44
      Configuring Context-level Administrators 44
      Configuring Context-level Operators 45
      Configuring Context-level Inspectors 45
      Verifying Context-level Administrative User Configuration 45
   Configuring Local-User Administrative Users 46
      Verifying Local-User Configuration 46
      Updating Local-User Database 47
      Updating and Downgrading the local-user Database 47
  Restricting User Access to a Specified Root Directory 48
      Configuring an SFTP root Directory 48
      Associating an SFTP root Directory with a Local User 48
      Associating an SFTP root Directory with an Administrator 49
      Associating an SFTP root Directory with a Config Administrator 49
  Configuring TACACS+ for System Administrative Users 49
     Operation 49
     User Account Requirements 50
        TACACS+ User Account Requirements 50
        StarOS User Account Requirements 50
     Configuring TACACS+ AAA Services 51
     Configuring TACACS+ for Non-local VPN Authentication 52
     Verifying the TACACS+ Configuration 52
  Configuring a Chassis Key 53
     Overview 53
     Configuring a New Chassis Key Value 53
        CLI Commands 53
        Quick Setup Wizard 54
     Configuring MIO/UMIO Port Redundancy 55
        Configuring MIO/UMIO Port Redundancy Auto-Recovery 57
Verifying Port Redundancy Auto-Recovery 58
Configuring Data Processing Card Availability 58
Verifying Card Configurations 59
Enabling Automatic Reset of FSC Fabric 59
Configuring ASR 5500 Link Aggregation 59
  LAG and Master Port 60
  LAG and Port Redundancy 60
  LAG and Multiple Switches 60
    Multiple Switches with L2 Redundancy 60
  Port States for Auto-Switch 61
  Hold Time 61
  Preferred Slot 61
  Auto-Switch Criteria 62
Link Aggregation Control 62
Minimum Links 63
Redundancy Options 64
Horizontal Link Aggregation with Two Ethernet Switches 64
Non-Redundant (Active-Active) LAG 64
Link Aggregation Status 65
Configuring a Demux Card 66
  Overview 66
  MIO Demux Restrictions 66
  Configuration 67

CHAPTER 4 Management Settings 69

ORBEM 69
  Configuring ORBEM Client and Port Parameters 70
  Configuring IIOP Transport Parameters 70
  Verifying ORBEM Parameters 71
SNMP MIB Browser 71
SNMP Support 74
  Configuring SNMP and Alarm Server Parameters 74
  Verifying SNMP Parameters 75
  Controlling SNMP Trap Generation 76
CHAPTER 5

Verifying and Saving Your Configuration 77

Verifying the Configuration 77

Feature Configuration 77

Service Configuration 78

Context Configuration 78

System Configuration 78

Finding Configuration Errors 78

Synchronizing File Systems 79

Saving the Configuration 79

CHAPTER 6

System Interfaces and Ports 81

Contexts 81

Creating Contexts 81

Viewing and Verifying Contexts 82

Ethernet Interfaces and Ports 82

Creating an Interface 83

Configuring a Port and Binding It to an Interface 83

Configuring a Static Route for an Interface 83

Viewing and Verifying Port Configuration 84

CHAPTER 7

System Security 87

Per-Chassis Key Identifier 87

MIO Synchronization 88

Protection of Passwords 88

Secure Password Encryption 88

Support for Non-Current Encryptions and Decryptions 89

Support for ICSR Configurations 89

Encrypted SNMP Community Strings 90

Lawful Intercept Restrictions 90

LI Server Addresses 90

Modifying Intercepts 91

Adding, Modifying and Removing Users 91

Notification of Users Being Added or Deleted 91

Notification of Changes in Privilege Levels 91
User Access to Operating System Shell 91

Test-Commands 92
  Enabling cli test-commands Mode 92
  Enabling Password for Access to CLI-test commands 92
  Exec Mode cli test-commands 93
  Configuration Mode cli test-commands 93

CHAPTER 8
Software Management Operations 95

Understanding the Local File System 95
  File Types Used by the Local File System 95
  Understanding the boot.sys File 96
Maintaining the Local File System 96
  File System Management Commands 96
    Synchronizing the File System 97
    Creating Directories 97
    Renaming Files and Directories 97
    Copying Files on the ASR 5500 Chassis 98
    Deleting Files 98
    Removing Directories 98
    Formatting Local Devices 98
Applying Pre-existing CLI Configuration Files 99
Viewing Files on the Local File System 99
  Viewing the Contents of a Local Device 99
  Viewing CLI Configuration and boot.sys Files 100
  Validating an Operating System File 100

Configuring the Boot Stack 101
  System Boot Methods 101
  Viewing the Current Boot Stack 101
  Adding a New Boot Stack Entry 103
  Deleting a Boot Stack Entry 103
  Network Booting Configuration Requirements 103
    Configuring the Boot Interface 103
    Configuring the Boot Network 104
    Configuring Boot Network Delay Time 105
    Configuring a Boot Nameserver 105
Upgrading the Operating System Software 105
Identifying OS Release Version and Build Number 106
Verify Free Space on the /flash Device 106
Download the Software Image from the Support Site 107
Transfer StarOS Image to /flash on the Chassis 107
Saving a Copy of the Current Configuration File 107
Downgrading from Release 15.0 to 14.0 108
Downgrading from Release 20.0 108
Off-line Software Upgrade 108
  Configure a Newcall Policy 109
  Configure a Message of the Day Banner 109
  Back up the Current CLI Configuration File 109
  Create a New Boot Stack Entry 110
  Synchronize File Systems 110
  Save the Running Configuration 110
  Reboot the Chassis 110
Verify the Running Software Version 111
Restoring the Previous Software Image 111
Upgrading ICSR Chassis 111
Performing Dynamic Software Updates 111
Managing License Keys 112
  New System License Keys 112
  Session Use and Feature Use Licenses 112
  Installing New License Keys 112
    Cutting and Pasting the Key 113
    Adding License Keys to Configuration Files 113
License Expiration Behavior 114
Requesting License Keys 114
Viewing License Information 115
Deleting a License Key 115
Management Card Replacement and License Keys 115
Managing Local-User Administrative Accounts 115
  Configuring Local-User Password Properties 115
  Configuring Local-User Account Management Properties 116
    Local-User Account Lockouts 116
Local-User Account Suspensions 116
Changing Local-User Passwords 116

CHAPTER 9
Monitoring the System 119
SNMP Notifications 119
Monitoring System Status and Performance 119
Clearing Statistics and Counters 121
Monitoring ASR 5500 Hardware Status 121

CHAPTER 10
Bulk Statistics 123
Configuring Communication with the Collection Server 123
Configuring Standard Settings 123
Configuring Optional Settings 124
Configuring Bulk Statistic Schemas 124
Configuring a Separate Bulkstats Config File 124
Using show bulkstats Commands 125
Verifying Your Configuration 125
Saving Your Configuration 126
Viewing Collected Bulk Statistics Data 126
Manually Gathering and Transferring Bulk Statistics 127
Clearing Bulk Statistics Counters and Information 127
Bulkstats Schema Nomenclature 127
Statistic Types 127
Data Types 128
Key Variables 128
Bulk Statistics Event Log Messages 130

CHAPTER 11
System Logs 131
System Log Types 131
Configuring Event Logging Parameters 132
Configuring Event Log Filters 132
Exec Mode Filtering 132
Global Configuration Mode Filtering 133
Configuring syslog Servers 134
Configuring Active Logs 134
Specifying Facilities 135
Configuring Trace Logging 143
Configuring Monitor Logs 144
   Enabling Monitor Logs 144
   Disabling Monitor Logs 144
Viewing Logging Configuration and Statistics 144
Viewing Event Logs Using the CLI 145
Configuring and Viewing Crash Logs 146
   Crash Logging Architecture 146
   Configuring Software Crash Log Destinations 147
   Viewing Abridged Crash Log Information Using the CLI 148
Checkpointing Logs 149
Saving Log Files 150
Event ID Overview 150
   Event Severities 158
   Understanding Event ID Information in Logged Output 159

CHAPTER 12

Troubleshooting 161
Detecting Faulty Hardware 161
   Licensing Issues 162
   Using the CLI to View Status LEDs 162
Checking the LEDs on the PFU 162
Checking the LEDs on the MIO Card 164
   MIO Run/Fail LED States 164
   MIO Active LED States 165
   MIO Redundancy LED States 166
   MIO Master LED States 166
   MIO Busy LED States 167
   MIO – Interface Link LED States 167
   MIO – Interface Activity LED States 168
Checking the LEDs on the DPC 168
   DPC Run/Fail LED States 169
   DPC Active LED States 170
   DPC Redundancy LED States 171
Checking the LEDs on the FSC 171
FSC Run/Fail LED States 172
FSC Active LED States 173
FSC Redundancy LED States 173
FSC Drive n Activity LED States 174

Checking the LEDs on the SSC 175
SSC Run/Fail LED States 175
SSC Active LED States 176
SSC Redundancy LED States 177
SSC System Status LED States 177
SSC System Service LED States 178

Testing System Alarm Outputs 178

Taking Corrective Action 178
Switching MIOs 179
Busying Out a DPC 179
Migrating a DPC 180
Halting Cards 180
  Initiate a Card Halt 180
  Restore a Previously Halted Card 181

Verifying Network Connectivity 181
Using the ping or ping6 Command 182
Syntax 182
Troubleshooting 182
Using the traceroute or traceroute6 Command 183
  traceroute – IPv4 183
  traceroute6 – IPv6 183

Viewing IP Routes 183
Viewing the Address Resolution Protocol Table 184

Using the System Diagnostic Utilities 184
Using the Monitor Utility 184
Using the Protocol Monitor 185
  Using the Protocol Monitor for a Specific Subscriber 186

Generating an SSD 187
Configuring and Using the Support Data Collector 188

CHAPTER 13

System Recovery 189
Prerequisites 189
   Console Access 189
   Boot Image 189
Accessing the boot CLI 190
   Initiate a Reboot 190
   Interrupt the Boot Sequence 190
   Enter CLI Mode 191
   boot Command Syntax 191
Booting from a Selected Image 191
   Boot Using No Configuration File 191
   Boot Using A Specified Configuration File 192

CHAPTER 14

Access Control Lists 193
   Overview 193
   Understanding ACLs 194
      Rule(s) 194
         Actions 194
         Criteria 194
      Rule Order 196
   Configuring ACLs on the System 196
      Creating ACLs 196
      Configuring Action and Criteria for Subscriber Traffic 197
      Configuring an Undefined ACL 197
      Verifying the ACL Configuration 198
   Applying IP ACLs 198
      Applying the ACL to an Interface 200
         Applying an ACL to an Individual Interface 200
         Verifying the ACL Configuration on an Interface 201
      Applying the ACL to a Context 201
         Applying an ACL to All Traffic Within a Context 202
         Verifying the ACL Configuration in a Context 202
      Applying an ACL to a RADIUS-based Subscriber 203
      Applying an ACL to an Individual Subscriber 204
         Verifying the ACL Configuration to an Individual Subscriber 204
      Applying an ACL to the Subscriber Named default 205
Applying an ACL to the Subscriber Named default 205
Verifying the ACL Configuration to the Subscriber Named default 206
Applying an ACL to Service-specified Default Subscriber 206
Applying an ACL to Service-specified Default Subscriber 207
Verifying the ACL Configuration to Service-specified Default Subscriber 207
Applying a Single ACL to Multiple Subscribers 208
Applying an ACL to Multiple Subscriber via APNs 209
Applying an ACL to Multiple Subscriber via APNs 209
Verifying the ACL Configuration to APNs 210

CHAPTER 15
Congestion Control 211
Overview 211
Configuring Congestion Control 212
Configuring the Congestion Control Threshold 212
Configuring Service Congestion Policies 213
Configuring Overload Reporting on the MME 213
Enabling Congestion Control Redirect Overload Policy 214
Verify the Service Overload Policies 214
Verify the Congestion Control Configuration 214
Verify MME Congestion Action Profiles 214
Disconnecting Subscribers Based on Call or Inactivity Time 214

CHAPTER 16
Routing 217
Routing Policies 217
Creating IP Prefix Lists 218
Creating Route Access Lists 218
Creating AS Path Access Lists 218
Creating Route Maps 219
Sample Configuration 219
Static Routing 219
Adding Static Routes to a Context 220
Deleting Static Routes From a Context 220
OSPF Routing 220
OSPF Version 2 Overview 221
Basic OSPFv2 Configuration 222
Configuring IPv4 BFD for Static Routes 234
Configuring IPv6 BFD for Static Routes 234
Configuring BFD for Single Hop 235
Configuring Multihop BFD 235
Scaling of BFD 236
Associating BGP Neighbors with the Context 236
Associating OSPF Neighbors with the Context 236
Associating BFD Neighbor Groups with the BFD Protocol 236
Enabling BFD on OSPF Interfaces 237
   All OSPF Interfaces 237
   Specific OSPF Interface 237
Monitoring BFD Connection for ICSR 237
Saving the Configuration 237
Chassis-to-Chassis BFD Monitoring for ICSR 237
   Enable Primary Chassis BFD Monitoring 238
Set BFD to Ignore ICSR Dead Interval 238
Configure ICSR Switchover Guard Timer 238
Enable BFD Multihop Fall-over 239
   ip route Command 239
   ip routev6 Command 240
Adjust BFD Interval 240
Enable Advertising BGP Routes from Standby ICSR Chassis 240
Saving the Configuration 240
BFD Support for Link Aggregation Member Links 240
   Overview 241
   Configuring Support for BFD Linkagg Member-links 241
   Saving the Configuration 242
Viewing Routing Information 242

CHAPTER 17

VLANs 245
   Overview 245
   Overlapping IP Address Pool Support – GGSN 246
   RADIUS VLAN Support – Enhanced Charging Services 246
   APN Support – PDN Gateway (P-GW) 246
   Creating VLAN Tags 247
CHAPTER 21

Interchassis Session Recovery 279

Overview 279
Interchassis Communication 281
Checkpoint Messages 281
SRP CLI Commands 281
  Exec Mode CLI Commands 281
  show Commands 282
AAA Monitor 283
BGP Interaction 283
Requirements 283
ICSR Operation 285
  Chassis Initialization 288
Chassis Operation 288
  Chassis Communication 288
  Chassis Switchover 288
Configuring Interchassis Session Recovery (ICSR) 289
  Configuring the Service Redundancy Protocol (SRP) Context 290
    Creating and Binding the SRP Context 290
    Configuring SRP Context Parameters 291
      Basic Parameters 291
      SRP Redundancy, AAA and Diameter Guard Timers 292
      DSCP Marking of SRP Messages 293
  Optimizing Switchover Transitions 293
    Allow Non-VoLTE Traffic During ICSR Switchover 294
    Allow All Data Traffic 296
    Graceful Cleanup of ICSR After Audit of Failed Calls 296
    Optimization of Switchover Control Outage Time 296
  Configuring the SRP Context Interface Parameters 297
Configuring NACK Generation for SRP Checkpoint Messaging Failures 297
   Enabling NACK Messaging from the Standby Chassis 297
   Selective Disabling of NACK Messaging 298
Configuring LZ4 Compression Algorithm 299
Reducing Sync-Up Time with Standby ICSR Chassis 299
Verifying SRP Configuration 300
Modifying the Source Context for ICSR 300
   Configuring BGP Router and Gateway Address 300
   Configuring the SRP Context for BGP 301
   Verifying BGP Configuration 301
Modifying the Destination Context for ICSR 302
   Configuring BGP Router and Gateway Address in Destination Context 302
   Configuring SRP Context for BGP for Destination Context 302
   Setting Subscriber to Default Mode 302
   Verifying BGP Configuration in Destination Context 303
Disabling Bulk Statistics Collection on a Standby System 303
Verifying the Primary and Backup Chassis Configuration 303
Configuring Subscriber State Management Audit Process 304
Troubleshooting ICSR Operation 304
Updating the Operating System 305
   Both ICSR Chassis 310
      Downloading and Transferring the StarOS Build 310
   Standby Backup Chassis 311
      Performing Health Checks 311
      Performing SRP Checks 311
      Performing BGP Checks 312
      Updating the Boot Record 312
      Synchronizing File Systems 312
      Reloading the Chassis 312
      Updating the Configuration File 313
      Verifying the Software Version 313
      Saving the Configuration File 313
      Completing the Update Process 313
      Waiting for Session Synchronization 314
   Primary Chassis 314
Primary Task Subsystems 332
Controllers and Managers 333
Subsystem Tasks 334
  System Initiation Subsystem 334
  High Availability Subsystem 335
  Resource Manager Subsystem 336
  Virtual Private Networking Subsystem 336
  Network Processing Unit Subsystem 338
Session Subsystem 340
Platform Processes 350
Management Processes 354

**APPENDIX C**

**NETCONF and Confd** 357
Overview 357
Configuring Confd 358
  SSH Key Requirement 358
  NETCONF Protocol Configuration Mode 358
    autosave-config 359
    confd-user 359
  Seeding and Synchronizing the CDB 359
Sample Configuration 360
Verifying the Configuration 360
  show confdmgr Command 360
  show configuration confd Command 363
  clear confdmgr statistics 364
Show Support Details (SSD) 364
CDB Maintenance 364
  clear confdmgr confd cdb 364
  configure confd <url> 365
  save configuration <url> confd 365
Supported StarOS ECS Configuration Commands 365

**APPENDIX D**

**ICSR Checkpointing** 367
Overview of Checkpointing 367
Macro-checkpoints 367
GGSN_APN ID MAPPING 368
INSTANCE LEVEL CHECKPOINT 368
SERVICE_ID MAPPING 368
VPNMGR_ID MAPPING 369
Micro-checkpoints 369

Uncategorized 370
SESS_UCHKPT_CMD_INVALIDATE_CRR 370
SESS_UCKKPT_CMD_UPDATE_CLPSTATS 370
SESS_UCHKPT_CMD_UPDATE_IDLESECS 370

DCCA Category 371
SESS_UCHKPT_CMD_DCCA_SESS_INFO 371

ECS Category 371
SESS_UCHKPT_CMD_ACS_CALL_INFO 371
SESS_UCHKPT_CMD_ACS_GX_LI_INFO 372
SESS_UCHKPT_CMD_ACS_SESS_INFO 372
SESS_UCHKPT_CMD_DEL_ACS_CALL_INFO 372
SESS_UCHKPT_CMD_DEL_ACS_SESS_INFO 373
SESS_UCHKPT_CMD_DYNAMIC_CHRG_CA_INFO 373
SESS_UCHKPT_CMD_DYNAMIC_CHRG_DEL_CA_INFO 373
SESS_UCHKPT_CMD_DYNAMIC_CHRG_DEL_QG_INFO 374
SESS_UCHKPT_CMD_DYNAMIC_CHRG_QG_INFO 374
SESS_UCHKPT_CMD_DYNAMIC_RULE_DEL_INFO 374
SESS_UCHKPT_CMD_DYNAMIC_RULE_INFO 375

ePDG Category 375
SESS_UCHKPT_CMD_DELETE_EPG_BEARER 375
SESS_UCHKPT_CMD_UPDATE_EPDG_BEARER 375
SESS_UCHKPT_CMD_UPDATE_EPDG_PEER_ADDR 376
SESS_UCHKPT_CMD_UPDATE_EPDG_REKEY 376
SESS_UCHKPT_CMD_UPDATE_EPDG_STATS 376

Firewall/ECS Category 377
SESS_UCHKPT_CMD_SFW_DEL_RULE_INFO 377
SESS_UCHKPT_CMD_SFW_RULE_INFO 377

GGSN Category 378
SESS_UCHKPT_CMD_GGSN_DELETE_SUB_SESS 378
SESS_UCHKPT_CMD_GGSN_UPDATE_RPR 378
SESS_UCHKPT_CMD_GGSN_UPDATE_SESSION 378
SESS_UCHKPT_CMD_GGSN_UPDATE_STATS 379
SESS_UCHKPT_CMD_UPDATE_COA_PARAMS 379

Gx Interface Category 380
SESS_UCHKPT_CMD_ACS_VOLUME_USAGE 380
SESS_UCHKPT_CMD_UPDATE_SGX_INFO 380

NAT Category 380
SESS_UCHKPT_CMD_GR_UPDATE_NAT_REALM_PORT_INFO1 380
SESS_UCHKPT_CMD_GR_UPDATE_NAT_REALMS 381
SESS_UCHKPT_CMD_NAT_SIP_ALG_CALL_INFO 381
SESS_UCHKPT_CMD_NAT_SIP_ALG_CONTACT_PH_INFO 382
SESS_UCHKPT_CMD_UPDATE_DSK_FLOW_CHKPT_INFO 382
SESS_UCHKPT_CMD_UPDATE_NAT_BYPASS_FLOW_INFO 382

P-GW Category 383
SESS_UCHKPT_CMD_PGW_DELETE_SUB_SESS 383
SESS_UCHKPT_CMD_PGW_OVRCHRG_PRTCTN_INFO 383
SESS_UCHKPT_CMD_PGW_SGWRESTORATION_INFO 383
SESS_UCHKPT_CMD_PGW_UBR_MBR_INFO 384
SESS_UCHKPT_CMD_PGW_UPDATE_APN_AMBR 384
SESS_UCHKPT_CMD_PGW_UPDATE_INFO 384
SESS_UCHKPT_CMD_PGW_UPDATE_LI_PARAM 384
SESS_UCHKPT_CMD_PGW_UPDATE_PDN_COMMON_PARAM 385
SESS_UCHKPT_CMD_PGW_UPDATE_QOS 385
SESS_UCHKPT_CMD_PGW_UPDATE_SGW_CHANGE 385
SESS_UCHKPT_CMD_PGW_UPDATE_STATS 385

Rf Interface Category 385
SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_QCI_RF 385
SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_QCI_RF_WITH_FC 386
SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_RATING_GROUP_RF 386
SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_RATING_GROUP_RF_WITH_FC 386

S6b Interface Category 387
SESS_UCHKPT_CMD_S6B_INFO 387

SaMOG Category 387
SESS_UCHKPT_CMD_CGW_DELETE_BEARER 387
SESS_UCHKPT_CMD_CGW_DELETE_PDN 387
APPENDIX E

ASR 5500 SDR CLI Command Strings 393
About this Guide

This preface describes the System Administration Guide, how it is organized and its document conventions. The System Administration Guide describes how to generally configure and maintain StarOS running on an ASR 5500 platform. It also includes information on monitoring system performance and troubleshooting.

- Conventions Used, page xxv
- Related Documentation, page xxvi
- MIOs and DPCs, page xxvi
- Contacting Customer Support, page xxvii

Conventions Used

The following tables describe the conventions used throughout this documentation.

<table>
<thead>
<tr>
<th>Notice Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Note</td>
<td>Provides information about important features or instructions.</td>
</tr>
<tr>
<td>Caution</td>
<td>Alerts you of potential damage to a program, device, or system.</td>
</tr>
<tr>
<td>Warning</td>
<td>Alerts you of potential personal injury or fatality. May also alert you of potential electrical hazards.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Typeface Conventions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text represented as a screen display</td>
<td>This typeface represents displays that appear on your terminal screen, for example: Login:</td>
</tr>
</tbody>
</table>
### Typeface Conventions

<table>
<thead>
<tr>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text represented as commands</td>
<td>This typeface represents commands that you enter, for example: show ip access-list. This document always gives the full form of a command in lowercase letters. Commands are not case sensitive.</td>
</tr>
<tr>
<td>Text represented as a command variable</td>
<td>This typeface represents a variable that is part of a command, for example: show card slot_number. slot_number is a variable representing the desired chassis slot number.</td>
</tr>
<tr>
<td>Text represented as menu or sub-menu names</td>
<td>This typeface represents menus and sub-menus that you access within a software application, for example: Click the File menu, then click New</td>
</tr>
</tbody>
</table>

## Related Documentation

The most up-to-date information for this product is available in the product Release Notes provided with each software release.

The following user documents are available on www.cisco.com:

- ASR 5500 Installation Guide
- AAA Interface Administration and Reference
- Command Line Interface Reference
- GTPP Interface Administration and Reference
- IPSec Reference
- Release Change Reference
- SNMP MIB Reference
- Statistics and Counters Reference
- Thresholding Configuration Guide
- Product-specific and feature-specific Administration guides

## MIOs and DPCs

The ASR 5500 supports a variety of Management Input/Output and Data Processing Card types. The currently supported Management Input/Output card types include:
MIO and UMIO card types differ only by the UMIO requirement for a Universal chassis license. The currently supported Data Processing Card types include:

- Data Processing Card (DPC)
- Universal Data Processing Card (UDPC)
- Data Processing Card version 2 (DPC2)
- Universal Data Processing Card version 2 (UDPC2)

DPC and UDPC card types differ only by the UDPC requirement for a Universal chassis license. DPC2 and UDPC2 card types differ only by the UDPC2 requirement for a Universal chassis license. The DPC2/UDPC2 is only supported on ASR 5500 running StarOS release 18.2+.

When reference is made to an MIO card or DPC in this guide, it is presumed to apply to all types of these cards as identified above.

**Contacting Customer Support**

Use the information in this section to contact customer support.

Refer to the support area of http://www.cisco.com for up-to-date product documentation or to submit a service request. A valid username and password are required to access this site. Please contact your Cisco sales or service representative for additional information.
System Operation and Configuration

The ASR 5500 is designed to provide subscriber management services for high-capacity 4G wireless networks. Before you connect to the command line interface (CLI) and begin system configuration, you must understand how the system supports these services. This chapter provides terminology and background information to consider before you configure the system.

- System Management Overview, page 1
- Terminology, page 3
- Trusted Builds, page 6
- How the System Selects Contexts, page 7
- Understanding the ASR 5500 Boot Process, page 9
- Understanding Configuration Files, page 11
- IP Address Notation, page 12
- Alphanumeric Strings, page 14

System Management Overview

ASR 5500 management capabilities reflect the requirements of the Telecommunications Management Network (TMN) model for network element (NE) and element management system (EMS) functions. The system also supports external element management applications via standards-based protocols (CORBA and SNMPv1, v2). Wireless operators can readily integrate the ASR 5500 into their overall network, service, and business management systems. All management is performed out-of-band for security and to maintain system performance.
There are multiple ways to manage the system either locally or remotely using its out-of-band management interfaces.

*Figure 1: System Management Interfaces*

Management options include:

- Local login through the Console port on the MIO card using an RS-232 Console connection (RJ45) directly or indirectly via a terminal server

- Remote login using Telnet, and Secure Shell (SSH) access to the CLI through the MIO card's Ethernet management interfaces:
  - Two autosensing RJ45 10/100/1000Base-T (IEEE 802.3ab) shielded twisted-pair (STP) ports

**Important** In release 20.0 and higher Trusted StarOS builds, the Telnet and FTP options are not available.

- Support for Common Object Request Broker Architecture (CORBA) via an Object Request Broker Element Manager (ORBEM) interface and Simple Network Management Protocol version 1 (SNMPv1) and version 2 (SNMPv2) for fault management

- Authentication via RADIUS/Diameter or TACACS+

The StarOS CLI provides complete Fault, Configuration, Accounting, Performance, and Security (FCAPS) capabilities as described in the remaining chapters of this guide.
By default StarOS supports local Console access to the CLI via the RS-232 Console port for initial system configuration.

## Terminology

This section defines important terms used throughout this guide.

### Contexts

A context is a logical grouping or mapping of configuration parameters that pertain to various physical ports, logical IP interfaces, and services. A context can be thought of as a virtual private network (VPN).

The system supports the configuration of multiple contexts. Each context is configured and operates independently of the others. Once a context has been created, administrative users can configure services, logical IP interfaces, and subscribers for that context and then bind the logical interfaces to physical ports.

You can also assign a domain alias to a context; if a subscriber's domain name matches one of the configured alias names for a context, that context is used.

### Ports

Ports are the physical connectors on line cards that support remote access and subscriber traffic. Port configuration includes traffic profiles, data encapsulation methods, media type, and other information for physical connectivity between the system and the rest of the network.

Ports are identified by the chassis slot number for the Management Input/Output (MIO/UMIO) card, followed by the physical connector number. For example, Port 5/10 identifies connector number 10 on the MIO/UMIO card in slot 5.

Associate ports with contexts through bindings. For additional information on bindings, refer to the Bindings section below. You can configure each physical port to support multiple logical IP interfaces, each with up to 17 IP addresses (one primary and up to 16 secondaries).

For complete information on line cards and port assignments, refer to the ASR 5500 Installation Guide.

**Important**

UMIO cards and UDPC/UDPC2s are direct replacements for MIO cards and DPC/DPC2s. However, a special Universal PID license must be purchased and installed on the chassis for each installed UMIO and UDPC/UDPC2. Contact your Cisco account representative for additional licensing information.

**Important**

Throughout this guide, any reference to an MIO card or DPC is assumed to also refer to the UMIO and UDPC/UDPC2 respectively.
Logical Interfaces

You must associate a port with a StarOS virtual circuit or tunnel called a logical interface before the port can allow the flow of user data. Within StarOS, a logical interface is a named interface associated with a virtual router instance that provides higher-layer protocol transport, such as Layer 3 IP addressing. Interfaces are configured as part of VPN contexts and are independent from the physical ports that will be used to bridge the virtual interfaces to the network.

Logical interfaces are associated with ethernet+ppp+tunnel addresses and are bound to a specific port during the configuration process. Logical interfaces are also associated with services through bindings. Services are bound to an IP address that is configured for a particular logical interface. When associated, the interface takes on the characteristics of the functions enabled by the service.

There are several types of logical interfaces to configure to support Simple and Mobile IP data applications. These are briefly defined below.

Management Interface

This interface provides the point of attachment to the management network. The interface supports remote access to the command line interface (CLI). It also supports Common Object Request Broker Architecture (CORBA)-based management via the Web Element Manager application, and event notification via the Simple Network Management Protocol (SNMP).

Define management interfaces in the local context and bind them to the ports on the Management Input/Output (MIO/UMIO) cards.

Bindings

A binding is an association between elements within the system. There are two types of bindings: static and dynamic.

Static binding is accomplished through system configuration. Static bindings associate:

- A specific logical interface (configured within a particular context) to a physical port. Once the interface is bound, traffic can flow through the context as if it were any physically-defined circuit. Static bindings support any encapsulation method over any interface and port type.
- A service to an IP address assigned to a logical interface within the same context. This allows the interface to take on the characteristics (that is, support the protocols) required by the service.

Dynamic binding associates a subscriber to a specific egress context based on the configuration of their profile or system parameters. This provides a higher degree of deployment flexibility, as it allows a wireless carrier to support multiple services and facilitates seamless connections to multiple networks.

Management ports can only be bound in the local context. Traffic or subscriber ports can only be bound in a non-local context.

Services

Configure services within a context to enable certain functionality. The following are examples of services you can configure on the system, subject to licensing availability and platform type:
• Gateway GPRS Support Node (GGSN) services
• Serving GPRS Support Node (SGSN) Services
• Packet Data Serving Node (PDSN) services
• Home Agent (HA) services
• Layer 2 Tunneling Protocol Access Concentrator (LAC) services
• Dynamic Host Control Protocol (DHCP) services
• Mobility Management Entity (MME) Services
• PDN Gateway (P-GW) Services
• Serving Gateway (S-GW) Services
• Intelligent Policy Control Function (IPCF) Services (PCC-Service, PCC-Policy, PCC-AF)

AAA Servers

Authentication, Authorization and Accounting (AAA) servers store profiles, perform authentication, and maintain accounting records for each mobile data subscriber. The AAA servers communicate with the system over an AAA interface. The system supports the configuration of up to 128 interfaces to AAA servers.

It is important to note that for Mobile IP, there can be Foreign AAA (FAAA) and Home AAA (HAAA) servers. FAAA servers typically reside in the carrier's network. HAAA servers could be owned and controlled by either the carrier or the home network. If the HAAA server is owned and controlled by the home network, accounting data is transferred to the carrier via an AAA proxy server.

Important
Mobile IP support depends on the availability and purchase of a standalone license or a license bundle that includes Home Agent (HA).

Subscribers

Subscribers are the end-users of the service; they gain access to the Internet, their home network, or a public network through the system.

There are three primary types of subscribers:

• **RADIUS-based Subscribers**: The most common type of subscriber, these users are identified by their International Mobile Subscriber Identity (IMSI) number, an Electronic Serial Number (ESN), or by their domain name or user name. They are configured on and authenticated by a RADIUS AAA server.

  Upon successful authentication, various attributes that are contained in the subscriber profile are returned. The attributes dictate such things as session parameter settings (for example, protocol settings and IP address assignment method), and what privileges the subscriber has.

Important
Attribute settings received by the system from a RADIUS AAA server take precedence over local-subscriber attributes and parameters configured on the system.
**Local Subscribers**: These are subscribers, primarily used for testing purposes, that are configured and authenticated within a specific context. Unlike RADIUS-based subscribers, the local subscriber's user profile (containing attributes like those used by RADIUS-based subscribers) is configured within the context where they are created.

When local subscriber profiles are first created, attributes for that subscriber are set to the system's default settings. The same default settings are applied to all subscriber profiles, including the subscriber named `default` which is created automatically by the system for each system context. When configuring local profile attributes, the changes are made on a subscriber-by-subscriber basis.

---

**Important** Attributes configured for local subscribers take precedence over context-level parameters. However, they could be over-ridden by attributes returned from a RADIUS AAA server.

---

**Management Subscribers**: A management user is an authorized user who can monitor, control, and configure the system through the CLI or Web Element Manager application. Management is performed either locally, through the system Console port, or remotely through the use of the Telnet or secure shell (SSH) protocols. Management users are typically configured as a local subscriber within the Local context, which is used exclusively for system management and administration. As with a local subscriber, a management subscriber's user profile is configured within the context where the subscriber was created (in this case, the Local context). However, management subscribers may also be authenticated remotely via RADIUS, if an AAA configuration exists within the local context, or TACACS+.

---

**Trusted Builds**

A Trusted build is a starfile image from which non-secure or low security features have been deleted or disabled. However, the binaries in the Trusted starfile image are are identical to those found in other starfiles for a particular StarOS release-build number. In general, a Trusted build is more restrictive than a Normal build image.

You can identify whether your platform is running a Trusted build via the Exec mode `show version` command. The output of the command displays the word "Trusted" as part of the image description text.

The following non-secure programs and features are disabled/removed from a Trusted build:

- Telnet
- FTP (File Transfer Protocol)
- Local user database access
- `tcpdump` utility
- `rlogin` (Remote Login) utility and `rlogind` (Remote Login daemon)
- `rsh` (Remote Shell) and `rcp` (Remote Copy) utilities
How the System Selects Contexts

This section describes the process that determines which context to use for context-level administrative users or subscriber sessions. Understanding this process allows you to better plan your configuration in terms of how many contexts and interfaces you need to configure.

Context Selection for Context-level Administrative User Sessions

The system comes configured with a context called local that you use specifically for management purposes. The context selection process for context-level administrative users (those configured within a context) is simplified because the management ports on the MIO are associated only with the Local context. Therefore, the source and destination contexts for a context-level administrative user responsible for managing the entire system should always be the local context.

A context-level administrative user can also connect through other interfaces on the system and still have full system management privileges.

A context-level administrative user can be created in a non-local context. These management accounts have privileges only in the context in which they are created. This type of management account can connect directly to a port in the context in which they belong, if local connectivity is enabled (SSHD, for example) in that context.

For all FTP or SFTP connections, you must connect through an MIO management interface. If you SFTP or FTP as a non-local context account, you must use the username syntax of username@contextname.

Important

In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

The context selection process becomes more involved if you are configuring the system to provide local authentication or work with a AAA server to authenticate the context-level administrative user.

The system gives you the flexibility to configure context-level administrative users locally (meaning that their profile will be configured and stored in its own memory), or remotely on an AAA server. If a locally-configured user attempts to log onto the system, the system performs the authentication. If you have configured the user profile on an AAA server, the system must determine how to contact the AAA server to perform authentication. It does this by determining the AAA context for the session.
The following table and flowchart describe the process that the system uses to select an AAA context for a context-level administrative user. Items in the table correspond to the circled numbers in the flowchart.

*Figure 2: Context-level Administrative User AAA Context*
Table 1: Context-level Administrative User AAA Context Selection

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>During authentication, the system determines whether local authentication is enabled in the local context. If it is, the system attempts to authenticate the administrative user in the local context. If it is not, proceed to item 2 in this table. If the administrative user's username is configured, authentication is performed by using the AAA configuration within the local context. If not, proceed to item 2 in this table.</td>
</tr>
<tr>
<td>2</td>
<td>If local authentication is disabled on the system or if the administrative user's username is not configured in the local context, the system determines if a domain was received as part of the username. If there is a domain and it matches the name of a configured context or domain, the system uses the AAA configuration within that context. If there is a domain and it does not match the name of a configured context or domain, Go to item 4 in this table. If there is no domain as part of the username, go to item 3 in this table.</td>
</tr>
<tr>
<td>3</td>
<td>If there was no domain specified in the username or the domain is not recognized, the system determines whether an AAA Administrator Default Domain is configured. If the default domain is configured and it matches a configured context, the AAA configuration within the AAA Administrator Default Domain context is used. If the default domain is not configured or does not match a configured context or domain, go to item 4 item below.</td>
</tr>
<tr>
<td>4</td>
<td>If a domain was specified as part of the username but it did not match a configured context, or if a domain was not specified as part of the username, the system determines if the AAA Administrator Last Resort context parameter is configured. If a last resort, context is configured and it matches a configured context, the AAA configuration within that context is used. If a last resort context is not configured or does not match a configured context or domain, the AAA configuration within the local context is used.</td>
</tr>
</tbody>
</table>

Context Selection for Subscriber Sessions

The context selection process for a subscriber session is more involved than that for the administrative users. Subscriber session context selection information for specific products is located in the Administration Guide for the individual product.

Understanding the ASR 5500 Boot Process

Part of the configuration process requires that you allocate hardware resources for processing and redundancy. Therefore, before you configure the system, it is important to understand the boot process which determines how the hardware components are brought on line.
The following steps describe the system's boot process:

**Step 1**  
When power is first applied to the chassis, or after a reboot, only the MIO/UMIOs in slot 5 and slot 6 receive power.

**Step 2**  
During the startup process, the MIO/UMIO performs a series of power-on self tests (POSTs) to ensure that its hardware is operational.

**Step 3**  
If the MIO/UMIO in slot 5 successfully executes all POSTs, it becomes the active MIO. The MIO in slot 6 becomes the standby card. If there is a problem with the MIO in slot 5, the MIO in slot 6 becomes the active MIO.

**Step 4**  
The active MIO/UMIO begins loading the operating system software image designated in the boot stack. The boot stack entries are contained in the boot.sys file that resides on flash memory on the MIO/UMIO. The standby MIO/UMIO observes the active card startup. If the file on the active MIO/UMIO is loads normally, the standby MIO/UMIO boots from the active card image. If the active MIO/UMIO experiences problems during this phase, the standby MIO/UMIO loads its software image designated by its own boot stack entry in its boot.sys file and takes over control of the system as the active MIO/UMIO.

**Step 5**  
After the software image is loaded into its memory, the active MIO/UMIO determines whether other cards are installed in the chassis by applying power to the other chassis slots and signalling them. If the chassis slot contains a card, power is left On to that slot. All empty slots are powered off.

If no MIOs are installed or if both fail to boot successfully, no other card installed in the system will boot.
Step 6 When power is applied to the DPC/UDPCs or DPC2/UDPC2s installed in the system, they each perform their own series of POSTs.

Step 7 After successful POST, each DPC/UDPC or DPC2/UDPC2 enters standby mode.

Step 8 After entering the standby mode, each of the control processors (CPs) on the DPC/UDPCs or DPC2/UDPC2s communicate with the active MIO/UMIO to receive the appropriate code.

Step 9 Upon successful loading of the software image, the system loads a configuration file designated in the boot stack (boot.sys file). If this is the first time the system is powered on and there is no configuration file, the active MIO/UMIO invokes the system's Quick Setup wizard. Use the Quick Setup wizard to configure basic system parameters for communication across the management network.

The wizard creates a configuration file (system.cfg) that you can use as a starting point for subsequent configurations. This allows you to configure the system automatically by applying the configuration file during any subsequent boot. For additional information about system configuration files, refer to the Understanding Configuration Files section.

Understanding Configuration Files

The system supports the use of a file or script to modify configurable parameters. Using a file for offline system configuration reduces the time it takes to configure parameters on multiple systems.

A system configuration file is an ASCII text file that contains commands and configuration parameters. When you apply the configuration file, the system parses through the file line-by-line, testing the syntax and executing the command. If the syntax is incorrect, a message is displayed to the CLI and the system proceeds to the next command. Lines that begin with # are considered remarks and are ignored.

### Important

Pipes ( | ), used with the `grep` and `more` keywords, can potentially cause errors in configuration file processing. Therefore, the system automatically ignores keywords with pipes during processing.

### Important

Always save configuration files in UNIX format. Failure to do so can result in errors that prevent configuration file processing.

The commands and configuration data within the file are organized and formatted just as they would be if they were being entered at the CLI prompt. For example, if you wanted to create a context called `source` in the CLI, you would enter the following commands at their respective prompts:

```plaintext
[local]host_name# config
[local]host_name(config)# context source
[source]host_name(config-ctx)# end
```

To create a context called `source` using a configuration file, you would use a text editor to create a new file that consists of the following:

```plaintext
config  
    context source  
    end
```

There are several important things to consider when using configuration files:
• The system automatically applies a configuration file at the end of the boot process. After the system boots up for the first time, a configuration file that you have created and that is tailored to your network needs, can be applied. To make the system use your configuration file, modify the system's boot parameters according to the instructions located in Software Management Operations.

• In addition to being applied during the boot process, you can also apply configuration files manually at any time by executing the appropriate commands at the CLI prompt. Refer to the instructions in Software Management Operations.

| Important | When you apply a configuration file after the boot process, the file does not delete the configuration loaded as part of the boot process. Only those commands that are duplicated are overwritten. |

• Configuration files can be stored in any of the following locations:
  • **USB Memory Stick**: Supported via a USB port on the active MIO (/usb1).
  • **Network Server**: Any workstation or server on the network that the system can access using the Secure File Transfer Protocol (SFTP). This is recommended for large network deployments in which multiple systems require the same configuration.
  • **/flash**: a solid-state device with limited storage.
  • **/hd-raid**: internal RAID storage.

• Each time you save configuration changes you made during a CLI session, you can save those settings to a file which you can use as a configuration file.

**IP Address Notation**

When configuring a port interface via the CLI you must enter an IP address. The CLI always accepts an IPv4 address, and in some cases accepts an IPv6 address as an alternative.

For some configuration commands, the CLI also accepts CIDR notation. Always view the online Help for the CLI command to verify acceptable forms of IP address notation.

**IPv4 Dotted-Decimal Notation**

An Internet Protocol Version 4 (IPv4) address consists of 32 bits divided into four octets. These four octets are written in decimal numbers, ranging from 0 to 255, and are concatenated as a character string with full stop delimiters (dots) between each number.

For example, the address of the loopback interface, usually assigned the host name localhost, is 127.0.0.1. It consists of the four binary octets 01111111, 00000000, 00000000, and 00000001, forming the full 32-bit address.

IPv4 allows 32 bits for an Internet Protocol address and can, therefore, support $2^{32}$ (4,294,967,296) addresses.
IPv6 Colon-Separated-Hexadecimal Notation

An Internet Protocol Version 6 (IPv6) address has two logical parts: a 64-bit network prefix, and a 64-bit host address part. An IPv6 address is represented by eight groups of 16-bit hexadecimal values separated by colons (:

A typical example of a full IPv6 address is 2001:0db8:85a3:0000:0000:8a2e:0370:7334

The hexadecimal digits are case-insensitive.

The 128-bit IPv6 address can be abbreviated with the following rules:

- Leading zeroes within a 16-bit value may be omitted. For example, the address fe80:0000:0000:0000:0202:b3ff:fe1e:8329 may be written as fe80:0:0:0:202:b3ff:fe1e:8329
- One group of consecutive zeroes within an address may be replaced by a double colon. For example, fe80::0:0:0:0:202:b3ff:fe1e:8329 becomes fe80::202:b3ff:fe1e:8329

IPv6 allows 128 bits for an Internet Protocol address and can support $2^{128}$ (340,282,366,920,938,000,000,000,000,000,000,000,000,000,000) internet addresses.

CIDR Notation

Classless Inter-Domain Routing (CIDR) notation is a compact specification of an Internet Protocol address and its associated routing prefix. It is used for both IPv4 and IPv6 addressing in networking architectures.

CIDR is a bitwise, prefix-based standard for the interpretation of IP addresses. It facilitates routing by allowing blocks of addresses to be grouped into single routing table entries. These groups (CIDR blocks) share an initial sequence of bits in the binary representation of their IP addresses.

CIDR notation is constructed from the IP address and the prefix size, the latter being the number of leading 1 bits of the routing prefix. The IP address is expressed according to the standards of IPv4 or IPv6. It is followed by a separator character, the slash (/) character, and the prefix size expressed as a decimal number.

The address may denote a single, distinct, interface address or the beginning address of an entire network. In the latter case the CIDR notation specifies the address block allocation of the network. The maximum size of the network is given by the number of addresses that are possible with the remaining, least-significant bits below the prefix. This is often called the host identifier.

For example:

- the address specification 192.168.100.1/24 represents the given IPv4 address and its associated routing prefix 192.168.100.0, or equivalently, its subnet mask 255.255.255.0.
- the IPv4 block 192.168.0.0/22 represents the 1024 IPv4 addresses from 192.168.0.0 to 192.168.3.255.
- the IPv6 block 2001:DB8::/48 represents the IPv6 addresses from 2001:DB8:0:0:0:0:0:0 to 2001:DB8:0:FFFF:FFFF:FFFF:FFFF.
- ::1/128 represents the IPv6 loopback address. Its prefix size is 128, the size of the address itself, indicating that this facility consists of only this one address.

The number of addresses of a subnet defined by the mask or prefix can be calculated as $2^{address\ size - mask}$, in which the address size for IPv4 is 32 and for IPv6 is 128. For example, in IPv4, a mask of /29 gives $2^{32-29} = 2^3 = 8$ addresses.
Alphanumeric Strings

Some CLI commands require the entry of an alphanumeric string to define a value. The string is a contiguous collection of alphanumeric characters with a defined minimum and maximum length (number of characters).

Character Set

The alphanumeric character set is a combination of alphabetic (Latin letters) and/or numeric (Arabic digits) characters. The set consists of the numbers 0 to 9, letters A to Z (uppercase) and a to z (lowercase). The underscore character ( _ ) and dash/hyphen (-) are also considered to be members of the alphanumeric set of characters.

Blank spaces (whitespaces or SPACE characters) should mostly be avoided in alphanumeric strings, except in certain ruledef formats, such as time/date stamps.

Do not use any of the following "special" characters in an alphanumeric string except as noted below:

- & (ampersand)
- ' (apostrophe)
- < > (arrow brackets) [see exception below]
- * (asterisk) [see wildcard exception below]
- {} (braces)
- [] (brackets)
- $ (dollar sign) [see wildcard exception below]
- ! (exclamation point) [see exception below]
- () [parentheses]
- % (percent) [see exception below]
- # (pound sign) [see exception below]
- ? (question mark)
- ' (quotation mark – single)
- " (quotation mark – double)
- ; (semicolon)
- \ (slash – backward) [see exception below]
- / (slash – forward) [see exception below]
- ~ (tilde)
- | (vertical bar) [see exception below]

The following characters may appear in strings entered in ruledefs, APNs, license keys and other configuration/display parameters:

- < > (arrow brackets) [less than or greater than]
• * (asterisk) [wildcard]
• : (colon)
• $ (dollar sign) [wildcard]
•  (dot)
• = (equals sign)
• ! (exclamation point)
• % (percent)
• / (slash – forward)
• | (vertical bar)

The following characters may be used to delimit the domain from the user name for global AAA functions:
• @ (at sign)
• - (dash or hyphen)
• # (hash or pound sign)
• % [percent]
• \ (slash – backward) [must be entered as double slash "\\"]
• / (slash – forward)

Quoted Strings

If descriptive text requires the use of spaces between words, the string must be entered within double quotation marks (" "). For example:

interface "Rack 3 Chassis 1 port 5/2"
Getting Started

Following successful installation of the system hardware, you must configure a set of software parameters. You then save these settings in a system configuration file that is launched whenever the system is reloaded. This segment provides instructions for connecting to the console port and creating the initial local context management configuration.

- ASR 5500 Configuration, page 17
- Using the ASR 5500 Quick Setup Wizard, page 17
- Using the CLI for Initial Configuration, page 23
- Configuring the System for Remote Access, page 25
- Configuring SSH Options, page 27
- Configuring the Management Interface with a Second IP Address, page 33

ASR 5500 Configuration

The first time power is applied to the system, the active Management Input/Output (MIO/UMIO/) card (typically the one installed in chassis slot 5) automatically launches a Quick Setup Wizard on its console port. This wizard guides you through the initial configuration of the system.

The serial console port (logical port 3) is located on the front panel of the MIO card.

You can choose not to use the wizard and perform the initial configuration by issuing commands via the command line interface (CLI). You can manually launch the wizard by running the setup command in the Exec mode. Refer to the Command Line Interface Reference for details.

The following sections describe how to configure the system.

Using the ASR 5500 Quick Setup Wizard

The Quick Setup Wizard consists of three parts:

- Configuring a context-level security administrator and hostname
- Configuring the Ethernet interface for out-of-band (OOB) management
• Configuring the system for remote CLI access

The following figure and table provides a flow diagram that shows the run logic of the wizard with supplemental notes.

**Figure 4: ASR 5500 Quick Setup Wizard Logic Diagram**
Table 2: Quick Setup Wizard Logic Diagram Callout Descriptions

<table>
<thead>
<tr>
<th>Item</th>
<th>Task</th>
<th>Description/Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Enter or exit the wizard.</td>
<td>Enter <strong>no</strong> at the prompt to automatically be directed to the command line interface (CLI). Proceed to <strong>Using the CLI for Initial Configuration</strong>, on page 23 for instructions on performing an initial system configuration with the CLI. Enter <strong>setup</strong> at the command prompt to re-invoke the wizard.</td>
</tr>
<tr>
<td>2</td>
<td>For release 19.2 and higher, configure a tech-support password.</td>
<td>The tech-support password is used to access cli test-commands. Configure an administrative username/password and a hostname for the system. The name of the default administrative user configured through the wizard is <strong>admin</strong>. Administrative user name is an alphanumeric string of 1 through 32 characters that is case sensitive. Administrative user password is an alphanumeric string of 1 through 63 characters that is case sensitive. Configure a valid, non-null hostname. The hostname is an alphanumeric string of 1 through 63 characters that is case sensitive.</td>
</tr>
<tr>
<td>3</td>
<td>Change chassis key value.</td>
<td>A unique chassis key is configured at the factory for each system. This key is used to decrypt encrypted passwords found in generated configuration files. The system administrator can create a unique chassis key that will be used to encrypt passwords stored in configuration files. Enter <strong>yes</strong> to set a new chassis key. Refer to the instructions in <strong>System Settings</strong>. Additional information can be found in the <strong>System Security</strong> chapter.</td>
</tr>
<tr>
<td>Item</td>
<td>Task</td>
<td>Description/Notes</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------------</td>
</tr>
<tr>
<td>4</td>
<td>Configure a single Management Input/Output (MIO/UMIO) out-of-band management interface for out-of-band system management.</td>
<td>Traffic on the management LAN is not transferred over the same media as user data and control signaling. For security reasons, it is recommended that management functions be maintained on a separate network from user data and control signaling. MIO port 1 (mio1) is the 1000Base-T default management port. MIO port 2 (mio2) is available as a secondary management port. Use the RJ-45 interfaces to connect the system to the management network with CAT5 Ethernet cable. Configure an IP address, subnet mask, and gateway for the interface. Instructions for configuring the second management interface on the MIO can be found in the <em>System Settings</em> chapter.</td>
</tr>
<tr>
<td>Item</td>
<td>Task</td>
<td>Description/Notes</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------------</td>
</tr>
<tr>
<td>5</td>
<td>In Release 19.0 and prior releases, enable various remote access protocols for accessing the system. In Release 19.2 and higher, enter an SSH key size in bits. The default value is 2048 bits.</td>
<td>Secure Shell (SSH) uses TCP port number 22 by default, if enabled. In Release 19.0 and prior releases, SSH V1 and/or V2 are supported. If SSH is enabled, you can also enable SSH File Transfer Protocol (SFTP) server functionality. In Release 19.2 and higher, you can specify the SSH key size. The SSH v2-RSA key generation uses that key size value.  <strong>Note:</strong> For maximum security, use only SSH v2. In Release 19.2 and higher, only SSH v2 is supported. Secure File Transfer Protocol (SFTP) uses TCP port number 22 by default, if enabled [subsystem sftp].</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> For maximum system security, do not enable telnet protocol. <strong>Note:</strong> in release 20.0 and higher Trusted StarOS builds, telnet is not supported.</td>
<td>Telnet uses TCP port number 23 by default, if enabled. File Transfer Protocol (FTP) uses TCP port number 21 by default, if enabled.  <strong>Note:</strong> For maximum system security, do not enable FTP. <strong>Note:</strong> in release 20.0 and higher Trusted StarOS builds, FTP is not supported.</td>
</tr>
<tr>
<td>6</td>
<td>Review and/or modify the configuration of previous prompts.</td>
<td>Enter the number of the prompt to be modified.  Configure the parameter.  Optional. Repeat step 1 and step 2 to modify additional settings.  Enter &quot;done&quot; when you have completed all changes.</td>
</tr>
<tr>
<td>7</td>
<td>Review the configure script created by the wizard based on your inputs.</td>
<td>An example of a created script is displayed in the example below. Variables are displayed in italics (variable).</td>
</tr>
<tr>
<td>8</td>
<td>Apply the configuration file to the system.</td>
<td>Once applied, the parameter configuration is automatically saved to the system.cfg file stored in MIO/UMIO flash memory.</td>
</tr>
</tbody>
</table>
Figure 5: MIO Interfaces

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Console port [Port 3]</td>
</tr>
<tr>
<td>2</td>
<td>USB port</td>
</tr>
<tr>
<td>3</td>
<td>10 GbE ports, DC-1 [Ports 10 – 19]</td>
</tr>
<tr>
<td>4</td>
<td>10 GbE ports, DC-2 [Ports 20 – 29]</td>
</tr>
<tr>
<td>5</td>
<td>1 GbE ports (1000Base-T) [Ports 1 and 2]</td>
</tr>
</tbody>
</table>
Do you want to view the configuration script created [yes/no]: y

config
  system hostname  hostname
  context local
    administrator  admin_name  password  passwd
    interface mio1
      ip address  ip_address  subnet
      #exit
      ip route 0.0.0.0 0.0.0.0  gw_address  mio1
    ssh key  v1_key
    ssh key  v2_rsa_key
    ssh key  v2_dsa_key
    server sshd
    subsystem sftp
    #exit
    no server telnetd
    no server ftpd
    #exit
    port ethernet 5/1
    bind interface mio1 local
    no shutdown
    #exit
  end
Do you want to apply configuration script created [yes/no]:

Important Once configuration using the wizard is complete, proceed to instructions on how to configure other system parameters.

Using the CLI for Initial Configuration

The initial configuration consists of the following:

- Configuring a context-level security administrator and hostname
- Configuring the Ethernet interface on the MIO/UMIO card
- Configuring the system for remote CLI access via Telnet, SSH, or FTP (secured or unsecured)

Important In release 20.0 and higher Trusted StarOS builds, FTP and telnet are not supported.

This section provides instructions for performing these tasks using the CLI.

Step 1 At the CLI prompt, enter:
[local] host_name# configure
[local] host_name(config)#

Step 2 Enter the context configuration mode by entering the following command:
[local] host_name(config)# context local
[local] host_name(config-ctx)#
The local context is the system's management context. Contexts allow you to logically group services or interfaces. A single context can consist of multiple services and can be bound to multiple interfaces.

**Step 3**
Enter the following command to configure a context-level security administrator for the system:

```
administrator user_name | encrypted | password password [ | ecs | | expiry-date date_time ] | ftp | [ li-administration | [ nocli | [ noecs | [ timeout-absolute timeout_absolute | | timeout-min-absolute timeout_min_absolute | [ timeout-idle timeout_idle | [ timeout-min-idle timeout_min_idle
```

You must configure a context-level security administrator during the initial configuration. After you complete the initial configuration process and end the CLI session, if you have not configured a security administrator, CLI access will be locked. For complete information about the commands in this section, see the Context Configuration Mode Commands chapter of the Command Line Interface Reference.

**Step 4**
Enter the following command at the prompt to exit the context configuration mode:

```
{local}\host_name(config-ctx)# exit
{local}\host_name(config)#
```

**Step 5**
Enter the following command to configure a hostname by which the system will be recognized on the network:

```
{local}\host_name(config)# system hostname host_name
```

`host_name` is the name by which the system will be recognized on the network. The hostname is an alphanumeric string of 1 through 63 characters that is case sensitive.

**Step 6**
Configure the network interfaces on the MIO/UMIO using the following instructions:

a) Enter the context configuration mode by entering the following commands:

```
{local}\host_name(config)# context local
{local}\host_name(config-ctx)#
```

b) Enter the following command to specify a name for the interface:

```
{local}\host_name(config-ctx)# interface interface_name
```

`interface_name` is the name of the interface expressed as an alphanumeric string of 1 through 79 characters that is case sensitive. The following prompt appears as the system enters the Ethernet Interface Configuration mode:

```
{local}\host_name(config-if-eth)#
```

c) Configure an IP address for the interface configured in the previous step by entering the following command:

```
{ ip | ipv6 address } ipaddress subnetmask
```

If you are executing this command to correct an address or subnet that was mis-configured with the Quick Setup Wizard, you must verify the default route and port binding configuration. Use step 11 and step 6 of this procedure. If there are issues, perform steps 7e through 7k to reconfigure the information.

d) Enter the following command to exit the Ethernet interface configuration mode:

```
{local}\host_name(config-if-eth)# exit
{local}\host_name(config-ctx)#
```

e) Configure a static route, if required, to point the system to a default gateway. Entering the following command:

```
{ ip | ipv6 } route gw_address interface_name
```

f) Enter the following to exit from the context configuration mode:

```
{local}\host_name(config-ctx)# exit
{local}\host_name(config)#
```

g) Enter the Ethernet Port Configuration mode:

```
port ethernet slot#/port#
```
h) Bind the port to the interface that you created in step 7b. Binding associates the port and all of its settings to the interface. Enter the following command:

```
(host_name(config-port-<slot#/port#>)# bind interface interface_name local

no shutdown
```

*interface_name* is the name of the interface that you configured in step 7b.

i) Exit the Ethernet Interface Configuration mode by entering the command:

```
(host_name(config-port-<slot#/port#>)# exit

(host_name(config)#
```

**Important** Refer below for instructions on configuring the MIO/UMIO management interface with a second IP address.

---

**Configuring the System for Remote Access**

Configure the system for remote access. An administrative user may access the system from a remote location over a local area network (LAN) or wide area network (WAN):

- Telnet
- Secure Shell (SSH)
- File Transfer Protocol (FTP) (secured or unsecured)
- Trivial File Transfer Protocol (TFTP)

**Important** If there are two simultaneous telnet sessions, and one administrator deletes the context into which the other administrator is logged, the administrator in the deleted context will not be automatically kicked into the *local* context. Although the deleted context will still appear in the CLI prompt, context specific commands will generate errors.

**Important** For maximum security, use SSH v2.

**Important** In release 20.0 and higher Trusted StarOS builds, FTP and telnet are not supported.

---

**Step 1** Enter the context configuration mode by entering the following command:

```
(host_name(config)# context local

(config-ctx)#
```

**Step 2** If desired, configure the system to allow Telnet access:

```
(host_name(config-ctx)# server telnetd
```
For maximum system security, you should not enable telnet.

**Step 3** Configure the system to allow SSH access:

```bash
[local] host_name(config-ctx)# ssh generate key [ type { v1-rsa | v2-rsa | v2-dsa } ]
```

v2-rsa is the recommended key type.

In StarOS 19.2 and higher, the v1-rsa keyword has been removed from and the v2-dsa keyword has been concealed within the Context Configuration mode ssh generate CLI command. A keyword that was supported in a previous release may be concealed in subsequent releases. StarOS continues to parse concealed keywords in existing scripts and configuration files created in a previous release. But the concealed keyword no longer appears in the command syntax for use in new scripts or configuration files. Entering a question mark (?) will not display a concealed keyword as part of the Help text. A removed keyword generates an error message when parsed.

```bash
[local] host_name(config-ctx)# ssh generate key type v2-rsa
```

**Step 4** Configure the system to support SFTP:

```bash
[local] host_name(config-ctx)# server sshd
[local] host_name(config-sshd)# subsystem sftp
[local] host_name(config-sshd)# exit
```

For additional information about SSH, see Configuring SSH Options, on page 27

**Step 5** Configure the system to allow FTP access, if desired, by entering the following command:

```bash
[local] host_name(config-ctx)# server ftpd
```

For maximum system security, you should not enable FTP. In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

**Step 6** Exit the configuration mode by entering the following command:

```bash
[local] host_name(config-ctx)# end
[local] host_name#
```

**Step 7** Verify the configuration by entering the following command:

```bash
[local] host_name# show configuration
```

The CLI output should be similar to the sample output:

```
context local
  interface interface_name
    ip address ipaddress subnetmask
    exit
  subscriber default
    exit
  administrator admin_name password admin_password
  no server telnetd
  no server ftppd
  ssh generate key
  server sshd
  subsystem sftp
  exit
  port ethernet 5/1
    bind interface interface_name local
    exit
  port ethernet 5/1
    no shutdown
    exit
```
Step 8  Verify the configuration of the IP routes by entering the following command:

```
[local] host_name# show ip route
```

The CLI output should be similar to the sample output:

```
** indicates the Best or Used route.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Nexthop</th>
<th>Protocol</th>
<th>Prec</th>
<th>Cost</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>*0.0.0.0/0</td>
<td>ipaddress</td>
<td>static</td>
<td>1</td>
<td>0</td>
<td>mio1</td>
</tr>
<tr>
<td>*network</td>
<td>0.0.0.0</td>
<td>connected</td>
<td>0</td>
<td>0</td>
<td>mio1</td>
</tr>
</tbody>
</table>
```

Step 9  Verify the interface binding by entering the following command:

```
[local] host_name# show ip interface name interface_name
```

`interface_name` is the name of the interface that was configured in step 7b. The CLI output should be similar to the sample output:

```
Intf Name: mio1
Intf Type: Broadcast
Description:
IP State: UP (Bound to 5/1 untagged, ifIndex 83951617)
IP Address: ipaddress Subnet Mask: subnetmask
Bcast Address: bcastaddress MTU: 1500
Resoln Type: ARP ARP timeout: 3600 secs
Number of Secondary Addresses: 0
```

Step 10  Save your configuration as described in *Verifying and Saving Your Configuration*.

---

### Configuring SSH Options

SSHv2 RSA is the only version of SSH supported under StarOS. Keywords previously supported for SSHv1 RSA and SSHv2 DSA have been removed from or concealed within the StarOS CLI.

**Important**  A keyword that was supported in a previous release may be concealed in subsequent releases. StarOS continues to parse concealed keywords in existing scripts and configuration files created in a previous release. But the concealed keyword no longer appears in the command syntax for use in new scripts or configuration files. Entering a question mark (?) will not display a concealed keyword as part of the Help text. Removed keywords generate an error message when parsed.

Version 1 of the SSH protocol is now obsolete due to security vulnerabilities. The `v1-rsa` keyword has been removed for the Context Configuration mode `ssh` command. Running a script or configuration that uses the SSHv1-RSA key returns an error message and generates an event log. The output of the error message is shown below:

```
CLI print failure Failure: SSH V1 contains multiple structural vulnerabilities and is no longer considered secure. Therefore we don't support v1-rsa SSH key any longer, please generate a new v2-rsa key to replace this old one.
```
If the system boots from a configuration that contains the `v1-rsa` key, you can expect a boot failure when logging in through SSH. The workaround is to log in via the Console port, re-generate a new ssh v2-rsa key, and configure server sshd. It will then be possible to log in via ssh.

The `v2-dsa` keyword is now concealed for the Context Configuration mode `ssh` command.

The `v1-rsa` keyword has been removed from the Exec mode `show ssh key` CLI command.

### SSH Keys

SSH key-based authentication uses two keys, one "public" key that anyone is allowed to see, and another "private" key that only the owner is allowed to see. You create a key pair, securely store the private key on the device you want to log in from, and store the public key on the ASR 5x00 that you wish to log into.

SSH keys are generated within a specified StarOS context. The context is associated with a user interface. You set or remove an administrative user name having authorized keys for access to the sshd server associated with context.

### Setting SSH Key Size

The Global Configuration mode `ssh key-size` CLI command configures the key size for SSH key generation for all contexts (RSA host key only).

**Step 1**
Enter the Global Configuration mode.

```
[local] host_name(config)
```

**Step 2**
Specify the bit size for SSH keys.

```
[local] host_name(config)# ssh key-size {2048 | 3072 | 4096 | 5120 | 6144 | 7168 | 9216 }
```

The default bit size for SSH keys is 2048 bits.

### Configuring SSH Key Generation Wait Time

SSH keys can only be generated after a configurable time interval has expired since the last key generation. The `ssh key-gen wait-time` command specifies this wait time in seconds. The default interval is 300 seconds (5 minutes).

**Step 1**
Enter the context configuration mode.

```
[local] host_name(config)# context context_name
```

**Step 2**
Specify the wait time interval.

```
[local] host_name(config-ctx)# ssh key-gen wait-time seconds
```

Notes:
specifying ssh encryption ciphers

the ssh configuration mode ciphers cli command configures the cipher priority list in sshd for ssh symmetric encryption. it changes the cipher options for that context.

step 1
enter the ssh configuration mode.
[local]host_name(config-ctx)# server sshd

step 2
specify the desired encryption algorithms.
[local]host_name(config-sshd)# ciphers algorithm

notes:

- algorithm is a string of 1 through 511 alphanumeric characters that specifies the algorithm(s) to be used as a single string of comma-separated variables (no spaces) in priority order from those shown below:

  - blowfish-cbc – symmetric-key block cipher, cipher block chaining, (cbc)
  - 3des-cbc – triple data encryption standard, cbc
  - aes128-cbc – advanced encryption standard (aes), 128-bit key size, cbc
  - aes128-ctr – aes, 128-bit key size, counter-mode encryption (ctr)
  - aes192-ctr – aes, 192-bit key size, ctr
  - aes256-ctr – aes, 256-bit key size, ctr
  - aes128-gcm@openssh.com – aes, 128-bit key size, galois counter mode [gcm], openssh
  - aes256-gcm@openssh.com – aes, 256-bit key size, gcm, openssh
  - chacha20-poly1305@openssh.com – chaCha20 symmetric cipher, poly1305 cryptographic message authentication code [mac], openssh

the default string for algorithm in a normal build is:
blowfish-cbc,3des-cbc,aes128-cbc,aes128-ctr,aes192-ctr,aes256-ctr,aes128=gcm@openssh.com,aes256-gcm@openssh.com, chacha20-poly1305@openssh.com

the default string for algorithm in a trusted build is:
aes256-ctr,aes192-ctr,aes128-ctr

step 3
exit the ssh configuration mode.
[local]host_name(config-sshd)# end
[local]host_name#
Generating SSH Keys

The `ssh generate` command generates a public/private key pair which is to be used by the SSH server. The `v1-rsa` keyword has been removed from and the `v2-dsa` keyword concealed within the `ssh generate` CLI command. The only keyword available for generating SSH keys is `v2-rsa`.

> **Important**
The generated key pair remains in use until the command is issued again.

**Step 1**
Enter the context configuration mode:

```
[local] host_name (config) # context context_name
[local] host_name (config-ctx) #
```

**Step 2**
Generate an SSH key pair.

```
[local] host_name (config-ctx) # ssh generate keytype v2-rsa
[local] host_name (config-ctx) #
```

Setting SSH Key Pair

The `ssh key` command sets the public/private key pair to be used by the system. The `v2-dsa` keyword is concealed in the `ssh key` command.

Specify the SSH key pair parameters.

```
[local] host_name (config-ctx) # ssh key data length octets type v2-rsa
```

Notes:

- `data` is the encrypted key expressed as an alphanumeric string of 1 through 1023 characters
- `length octets` is the length of the encrypted key in octets expressed as an integer from 0 through 65535
- `type` specifies the key type; `v2-rsa` is the only supported type.

> **Important** *For releases prior to 20.0, StarOS supports a maximum of 64 configurable authorized SSH keys. For release 20.0 and higher, StarOS supports a maximum of 200 configurable authorized SSH keys.*

Authorized SSH User Access

You must authorize users to access a StarOS context from a specific host with an SSH authentication-key pair.
Authorizing SSH User Access

The SSH Configuration mode **authorized-key** command grants user access to a context from a specified host.

**Step 1**
Go to the SSH Configuration mode.
```
[local]host_name(config-ctx)# server sshd
[local]host_name(config-sshd)#
```

**Step 2**
Specify administrative user access via the **authorized-key** command.
```
[local]host_name(config-sshd)# authorized-key username user_name host host_ip [ type { v2-dsa | v2-rsa } ]
```

Notes:
- **username user_name** specifies an existing StarOS administrator user name as having authorized keys for access to the sshd server. The **user_name** is expressed as an alphanumeric string of 1 through 255 characters. User names should have been previously created via the Context Configuration mode **administrator** command using the **nopassword** option to prevent bypassing of the sshd keys. Refer to the **System Settings** chapter for additional information on creating administrators.
- **host host_ip** specifies the IP address of an SSH host having the authorization keys for this username. The IP address must be in IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.
- **type** specifies the key type; **v2-rsa** is the only supported type.

SSH User Login Restrictions

An administrator can restrict SSH access to the StarOS CLI to a "white list" of allowed users. Access to a service may be restricted to only those users having a legitimate need. Only explicitly allowed users will be able connect to a host via SSH. The user name may optionally include a specific source IP address.

The AllowUsers list consists of user name patterns, separated by space. If the pattern takes the form 'USER' then login is restricted for that user. If pattern is in the format 'USER@IP_ADDRESS' then USER and IP address are separately checked, restricting logins to those users from the specified IP address.

The default is to allow unrestricted access by any user.

Creating an Allowed Users List

The **allowusers add** command allows an administrator to create a list of users who may log into the StarOS CLI.

**Step 1**
Enter the context configuration mode.
```
[local]host_name(config)# context context_name
[local]host_name(config-ctx)#
```
Step 2  Go to the SSH Configuration mode.

    [local] host_name(config-ctx)# server ssd

Step 3  Configure the SSH user list.

    [local] host_name(config-sshd)# allowusers add user_list

  user_list specifies a list of user name patterns, separated by spaces, as an alphanumeric string of 1 through 999 characters. If the pattern takes the form 'USER' then login is restricted for that user.

  If the pattern is in the format 'USER@IP_ADDRESS' then user name and IP address are separately checked, restricting logins to those users from that particular IP address.

  If the pattern is in the format 'USER@<context>@IP_ADDRESS' then user name, StarOS context and IP address are separately checked, restricting logins to those users associated with the specific context from that particular IP address.

  The following limits apply to the user_list:

    • The maximum length of this string is 3000 bytes including spaces.

    • The maximum number of AllowUsers, which is counted by spaces, is 256, which is consistent with the limit from OpenSSH.

Important  If you exceed either of the above limits, an error message is displayed. The message prompts you to use a regular expression pattern to shorten the string, or remove all the allowusers with no allowusers add or default allowusers add and re-configure.

  For additional information, see the SSH Configuration Mode Commands chapter in the Command Line Interface Reference.

Step 4  Exit the SSH Configuration mode.

    [local] host_name(config-sshd)# end

    [local] host_name#

### SSH User Login Authentication

StarOS authenticates SSH user login attempts via authorized-key/user-account pairings for the following scenarios:

  - User tries to login with local context username through local context (VPN) interface with authorized-key configured on local context.

  - User tries to login with non-local context username through non-local context interface with authorized-key configured on non-local context.

  - User tries to login with local context username through non-local context interface with authorized-key configured on local context.

  - User tries to login with non-local context username through local context interface with authorized-key configured on non-local context.

A failure to authenticate based on the current system configuration prevents the login and generates an error message.

StarOS does not permit users with different user IDs but having the same public SSH key to login to an unauthorized context. Authentication of the user takes into account the authorized-key/user-account pairing.
Enabling NETCONF

An SSH key is a requirement before NETCONF protocol and the ConfD engine can be enabled in support of Cisco Network Service Orchestrator (NSO).

Refer to the NETCONF and ConfD appendix in this guide for detailed information on how to enable NETCONF.

Configuring the Management Interface with a Second IP Address

If necessary, you can configure a second IP address on the MIO/UMIO management interface.

Step 1
Enter the configuration mode by entering the following command at the prompt:

```
[local\host_name#  configure
[local\host_name(config)#
```

Step 2
Enter the following to enter the context configuration mode:

```
[local\host_name(config)#  context local
[local\host-name(config-ctx)#
```

Step 3
Enter the interface slot number and port number by entering the following command:

```
[local\host_name(config-ctx)#  5/1
[local\host_name(config-if-eth)#
```

Step 4
Enter the secondary IP address and subnet mask by entering the following command:

```
[local\host_name(config-if-eth)#  {ip | ipv} address ipaddress subnet mask secondary
```

Step 5
Exit the configuration mode by entering the following command:

```
[local\host_name(config-if-eth)#  end
```

Step 6
Confirm the interface IP addresses by entering the following command:

```
[local\host_name#  show config context local
```

The CLI output should look similar to this example:

```
config
  context local
    interface interface_name
      ip address ipaddress subnetmask
      ip address ipaddress subnetmask secondary
    #exit
```

Step 7
Save your configuration as described in Verifying and Saving Your Configuration.
Configuring the Management Interface with a Second IP Address
System Settings

It is assumed that the procedures to initially configure the system as described in Getting Started have been completed.

Important

The commands used in the configuration examples in this section are the most likely-used commands and/or keyword options. In many cases, other optional commands and/or keyword options are available. Refer to the Command Line Interface Reference for complete information.

- Configuring a Second Management Interface, page 35
- Verifying and Saving Your Interface and Port Configuration, page 36
- Configuring System Timing, page 37
- Enabling CLI Timestamping, page 41
- Configuring CLI Confirmation Prompts, page 41
- Configuring System Administrative Users, page 43
- Configuring TACACS+ for System Administrative Users, page 49
- Configuring a Chassis Key, page 53
- Configuring MIO/UMIO Port Redundancy, page 55
- Configuring Data Processing Card Availability, page 58
- Enabling Automatic Reset of FSC Fabric, page 59
- Configuring ASR 5500 Link Aggregation, page 59
- Configuring a Demux Card, page 66

Configuring a Second Management Interface

Refer to Getting Started for instructions on configuring a system management interface on the Management Input/Output (MIO/UMIO) card. This section provides described how to configure a second management interface.
Use the following example to configure a second management interface:

```
configure
context local
  interface interface_name
    ip address ipaddress subnetmask
  exit
  ip route 0.0.0.0 0.0.0.0 gw_address interface_name
  exit
port ethernet slot#/port#
  bind interface interface_name local
  no shutdown
  media rj45
  end
```

Notes:

- For `port ethernet slot#`, use the actual chassis slot in which the active MIO/UMIO resides (slot number 5 or 6).
- Enter IP addresses using IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.
- For `port ethernet port#`, use the physical port on the MIO/UMIO card – port 1 or 2.
- The MIO/UMIO is equipped with RJ-45 (1000Base-T copper) interfaces. The RJ-45 interfaces connect the system to the management network via CAT3 or CAT5 Ethernet cable.
- **Option:** In the Ethernet Port configuration mode, configure the port speed, if needed, by entering the `medium` command. Refer to the *Command Line Interface Reference* for a complete explanation of this command.
- In the `{ip | ipv6} route` command, other keyword options, instead of the gateway IP address, are available and include: `next-hop` IP address, `point-to-point`, and `tunnel`.

## Verifying and Saving Your Interface and Port Configuration

Verify that your interface configuration settings are correct by entering the following command:

```
show ip interface
```

The output from this command should be similar to that shown below. In this example an interface named `mgmt2` was configured in the local context.

```
Intf Name: mgmt2
Intf Type: Broadcast
Description: management2
VRF: None
IP State: UP (Bound to 5/2)
IP Address: 192.168.100.3   Subnet Mask: 255.255.255.0
Bcast Address: 192.168.100.255 MTU: 1500
Resolv Type: ARP    ARP timeout: 60 secs
L3 monitor LC-port switchover: Disabled
Number of Secondary Addresses: 0
```

Verify that the port configuration settings are correct by entering the following command:

```
show configuration port slot#/port#
```

`slot#` is the chassis slot number of the line card where the physical port resides. `slot#` is either 5 or 6. `port#` is the number of the port (either 1 or 2).
This following command produces an output similar to the one shown below. It displays the configuration of port 2 of the MIO/UMIO installed in chassis slot 5. In this example, the port is bound to an interface called mgmt2.

```
config
  port ethernet 5/2
  description management2
  no shutdown
  bind interface mgmt2 local
end
```

Save your configuration as described in the Verifying and Saving Your Configuration chapter.

### Configuring System Timing

The system is equipped with a clock that supplies the timestamp for statistical counters, accounting records, logging, and event notification. After the initial configuration of the system clock, you can configure the system to communicate with one or more Network Time Protocol (NTP) server(s) to ensure that the clock is always accurate.

In the event of a power outage, the clock is maintained with an accuracy of ± one minute per month for up to 10 years. This ensures that when power is restored, the system is ready to process sessions and generate accounting, log, and event data with accurate timestamps.

In addition to configuring the timing source, you must configure the system's time zone.

#### Setting the System Clock and Time Zone

Use the following command example to configure the system clock and time zone:

```
clock set date:time
configure
  clock timezone timezone [ local ]
end
```

**Notes:**

- Enter the date and time in the format YYYY:MM:DD:HH:mm or YYYY:MM:DD:HH:mm:ss.
- Refer to the online Help for the `clock timezone` command for a complete list of supported time zones.
- The optional `local` keyword indicates that the time zone specified is the local timezone.
- Daylight Savings Time is automatically adjusted for time zones supporting it.

Save your configuration as described in the Verifying and Saving Your Configuration chapter.

#### Verifying and Saving Your Clock and Time Zone Configuration

Enter the following command to verify that you configured the time and time zone correctly:

```
show clock
```

The output displays the date, time, and time zone that you configured.
Configuring Network Time Protocol Support

This section provides information and instructions for configuring the system to enable the use of the Network Time Protocol (NTP).

Configure the system clock and time zone prior to implementing NTP support. This greatly reduces the time period that must be corrected by the NTP server.

Important

Many of the services offered by the ASR 5x00 platform require accurate timekeeping derived through NTP. If the time reference(s) used by StarOS are not accurate, the services may be unreliable. For this reason it should be assumed that normal system operation requires that NTP be configured.

The system uses NTP to synchronize internal clocks on the chassis to external time sources (typically GPS NTP sources, or other Stratum 2 or 3 servers, switches or routers).

By default, NTP is not enabled externally and should be configured when the system is initially installed. When enabled, the active MIO/UMIO will synchronize with external sources. If not enabled, the active MIO/UMIO will use its local clock as a time source. In the event of an NTP server or network outage, an already running MIO/UMIO will continue to use NTP to maintain time accuracy, but in a holdover mode.

All cards with CPUs synchronize to the active MIO/UMIO internally. This occurs even if an external NTP server is not configured. In the event of a MIO/UMIO switchover, all other cards will start synchronizing with the newly active MIO/UMIO automatically.

The system should have:

- NTP enabled.
- NTP configured for use in the local context only. Use of other contexts (which can be specified in the enable configurable) will cause issues.
- NTP configured for at least three external NTP servers. With three or more servers, outliers and broken or misconfigured servers can be detected and excluded. Generally, the more servers the better (within reason).

Important

Do not configure any external NTP servers using the prefer keyword. The NTP clock selection algorithms already have the built-in ability to pick the best server. Use of prefer usually results in a poorer choice than NTP can determine for itself.

Important

Do not change the maxpoll, minpoll, or version keyword settings unless instructed to do so by Cisco TAC.

Use the following example to configure the necessary NTP association parameters:

```
configure
ntp
   enable
   server ip_address1
   server ip_address2
```
server ip_address3
end

Notes:

• By default context_name is set to local. This is the recommended configuration.

• A number of options exist for the server command. Refer to the NTP Configuration Mode Commands chapter in the Command Line Interface Reference for more information.

• Enter the IP address of NTP servers using IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.

---

Important

Configure the system with at least three (preferably four) NTP servers.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

---

Configuring NTP Servers with Local Sources

NTP can use network peers, local external clocks (such as GPS devices), or a local clock with no external source.

A local clock with no external source is usually a last-resort clock when no better clock is available. It is typically configured on a site's intermediate NTP server so that when a WAN network outage occurs, hosts within the site can continue to synchronize amongst themselves.

You can configure this in ntpd or on many commercially available NTP devices. This local clock should always have a high stratum number (8+) so that under normal conditions (when real sources are available) this local clock will not be used.

Using a Load Balancer

The NTP daemon and protocol assume that each configured server is running NTP. If a NTP client is configured to synchronize to a load balancer that relays and distributes packets to a set of real NTP servers, the load balancer may distribute those packets dynamically and confuse the NTP client. NTP packets are latency and jitter sensitive. Relaying them through a load balancer can confuse the NTP client and is not a supported practice.

Verifying the NTP Configuration

Verify the NTP configuration is correct. Enter the following command at the Exec mode prompt:

```
show ntp associations
```

The output displays information about all NTP servers. See the output below for an example deploying two NTP servers.

```
+----Peer Selection:  ( ) - Rejected / No Response
  | (x) - False Tick
  | (.) - Excess
  | (-) - Outlyer
  | (+) - Candidate
  | (#) - Selected
```
The following table describes the parameters output by the `show ntp associations` command.

Table 3: NTP Parameters

<table>
<thead>
<tr>
<th>Column Title</th>
<th>Description</th>
</tr>
</thead>
</table>
| remote       | List of the current NTP servers. One of these characters precedes each IP address to show the server's current condition:  
  - ( ) Rejected/No response  
  - X False tick  
  - . Excess  
  - - Outlier  
  - + Candidate  
  - # Selected  
  - * System peer  
  - (o) PPS peer |
| refid        | Last reported NTP reference to which the server is synchronizing. |
| st           | NTP server stratum level. |
| t            | Communication type: broadcast, multicast, etc. |
| when         | Number of seconds since the last contact. |
| poll         | Polling interval between the system and the NTP server. |
| reach        | Octal value of the reachability shift register indicating which responses were received for the previous eight polls to this NTP server. |
| delay        | Round-trip delay (in milliseconds) for messages exchanged between the system and the NTP server. |
| offset       | Number of milliseconds by which the system clock must be adjusted to synchronize it with the NTP server. |
| jitter       | Jitter in milliseconds between the system and the NTP server. |
Enabling CLI Timestamping

To display a timestamp (date and time) for every command that is executed on the CLI, enter the following command at the root prompt for the Exec mode:

\texttt{timestamps}

The date and time appear immediately after you execute the command.

Save the configuration as described in the \textit{Verifying and Saving Your Configuration} chapter.

Configuring CLI Confirmation Prompts

A number of Exec mode and Global Configuration mode commands prompt users for a confirmation (Are you sure? [Yes|No]:) prior to executing the command.

This section describes configuration settings that:

\begin{itemize}
  \item Automatically confirm commands for the current CLI session (Exec mode) or for all CLI sessions and users (Global Configuration mode).
  \item Requires confirmation prompting only for the Exec mode \texttt{configure} command and \texttt{autoconfirm} command.
  \item Selectively requires confirmation of Exec mode configuration commands.
\end{itemize}

Enabling Automatic Confirmation

You can use the \texttt{autoconfirm} command to disable confirmation prompting for configuration commands. The \texttt{autoconfirm} command is available in the Exec mode and Global Configuration mode. Enabling the autoconfirm feature automatically supplies a "Yes" response to configuration command prompts, including for critical commands such as \texttt{reload} and \texttt{shutdown}. By default autoconfirm is disabled.

In the Exec mode, autoconfirm applies only to the current interactive CLI session.

In the Global Configuration mode, autoconfirm applies to all CLI sessions for all CLI users:

\begin{verbatim}
configure
  autoconfirm
end
\end{verbatim}

To disable autoconfirm once it has been enabled, use the \texttt{no autoconfirm} command.

\begin{footnotesize}
\begin{itemize}
  \item \textbf{Important} If commandguard is enabled, autoconfirm will disable commandguard.
\end{itemize}
\end{footnotesize}

Autoconfirm is intended as an "ease-of-use" feature. It presumes that the answer to "Are you sure? [Y/N]" prompts will be "Yes", and skips the prompt. Its use implies that the user is an expert who does not need these "safety-net" prompts.
Requiring Confirmation for autoconfirm and configure Commands

You can require confirmation prompting for the **autoconfirm** (Exec mode and Global Configuration mode) and **configure** (Exec mode) commands via the Global Configuration mode **commandguard** command.

---

**Important**

If autoconfirm is enabled, commandguard will not take effect until autoconfirm is disabled in both Exec and Global Configuration modes.

---

The following command sequence enables the commandguard feature:

```
configure
  commandguard
end
```

With commandguard enabled the confirmation prompt appears as shown in the example below:

```
[local] host_name# configure
Are you sure? [Yes|No]: yes
[local] host_name(config)#
```

To disable commandguard once it has been enabled, use the **no commandguard** command.

The status of **commandguard** is output in **show configuration** commands.

---

Requiring Confirmation for Specific Exec Mode Commands

A keyword for the **commandguard** command allows you to apply mandatory prompting for specified categories of Exec mode configuration commands, even when autoconfirm is enabled.

The command syntax is as follows:

```
configure
  commandguard exec-command exec_mode_category
end
```

Notes:

- **exec-command** *exec_mode_category* specifies one of the following categories of Exec mode configuration commands.
  - card
  - clear
  - copy
  - debug
  - delete
  - filesystem
  - hd
  - reload
  - rename
* shutdown
* task
* upgrade

- You can enter multiple `commandguard exec-command exec_mode_category` commands.
- All Exec mode commands beginning with the specified category word will prompt for confirmation, regardless if autoconfirm is enabled.
- You can turn off confirmation prompting for a specific category using `no commandguard exec-command exec_mode_category`.
- If autoconfirm is overridden by `commandguard exec-command` for an Exec mode command, StarOS displays an informational message indicating why autoconfirm is being overridden when you attempt to execute the command.
- Users may selectively override confirmation prompting for any Exec mode configuration command that supports the `-noconfirm` keyword.

For example, with `commandguard exec-command card` enabled, the confirmation prompt appears as shown below:

```
[local]host_name# card busy-out 1
Info: commandguard prevents autoconfirm of this command
Are you sure? [Yes|No]: yes
[local]host_name#
```

## Configuring System Administrative Users

*Getting Started* describes how to configure a context-level security administrator for the system.

This section provides instructions for configuring additional administrative users having the following privileges:

- **Security Administrators**: have read-write privileges and can execute all CLI commands, including those available to Administrators, Operators, and Inspectors
- **Administrators**: have read-write privileges and can execute any command in the CLI except for a few security-related commands that can only be configured by Security Administrators. Administrators can configure or modify system settings and execute all system commands, including those available to the Operators and Inspectors.
- **Operators**: have read-only privileges to a larger subset of the Exec Mode commands. They can execute all commands that are part of the inspector mode, plus some system monitoring, statistic, and fault management functions. Operators do not have the ability to enter the Config Mode.
- **Inspectors**: are limited to a few read-only Exec Mode commands. The bulk of these are `show` commands for viewing a variety of statistics and conditions. An Inspector cannot execute `show configuration` commands and does not have the privilege to enter the Config Mode.

Configuration instructions are categorized according to the type of administrative user: context-level or local-user.
Configuring Context-level Administrative Users

This user type is configured at the context-level and relies on the AAA subsystems for validating user names and passwords during login. This is true for both administrative user accounts configured locally through a configuration file or on an external RADIUS or TACACS+ server. Passwords for these user types are assigned once and are accessible in the configuration file.

This section contains information and instructions for configuring context-level administrative user types.

Configuring Context-level Security Administrators

Use the example below to configure additional security administrators:

```
configure
c   context local
      administrator  user_name  { [ encrypted ] | nopassword | password  password } 
   end
```

Notes:

- Additional keyword options are available that identify active administrators or place time thresholds on the administrator. Refer to the Command Line Interface Reference for more information about the administrator command.

- The nopassword option allows you to create an administrator without an associated password. Enable this option when using ssh public keys (authorized key command in SSH Configuration mode) as a sole means of authentication. When enabled this option prevents someone from using an administrator password to gain access to the user account.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

Configuring Context-level Administrators

Use the example below to configure context-level configuration administrators:

```
configure
c   context local
      config-administrator  user_name  { [ encrypted ] | nopassword | password  password } 
   end
```

Notes:

- Additional keyword options are available that identify active administrators or place time thresholds on the administrator. Refer to the Command Line Interface Reference for more information about the config-administrator command.

- The nopassword option allows you to create a config-administrator without an associated password. Enable this option when using ssh public keys (authorized key command in SSH Configuration mode)
Configuring Context-level Administrative Users

Configuring Context-level Operators

Use the example below to configure context-level operators:

```
configure
  context local
    operator user_name { [ encrypted ] [ nopassword ] password password }
  end
```

Notes:

- Additional keyword options are available that identify active administrators or place time thresholds on the administrator. Refer to the Command Line Interface Reference for more information about the operator command.
- The nopassword option allows you to create an operator without an associated password. Enable this option when using ssh public keys (authorized key command in SSH Configuration mode) as a sole means of authentication. When enabled this option prevents someone from using an operator password to gain access to the user account.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

Configuring Context-level Inspectors

Use the example below to configure context-level inspectors:

```
configure
  context local
    inspector user_name { [ encrypted ] [ nopassword ] password password }
  end
```

Notes:

- Additional keyword options are available that identify active administrators or place time thresholds on the administrator. Refer to the Command Line Interface Reference for more information about the inspector command.
- The nopassword option allows you to create an inspector without an associated password. Enable this option when using ssh public keys (authorized key command in SSH Configuration mode) as a sole means of authentication. When enabled this option prevents someone from using an inspector password to gain access to the user account.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

Verifying Context-level Administrative User Configuration

Verify that the configuration was successful by entering the following command:

```
show configuration context local
```
This command displays all of the configuration parameters you modified within the Local context during this session. The following displays sample output for this command. In this example, a security administrator named testadmin was configured.

```
config
  context local
    interface mgmt1
      ip address 192.168.1.10 255.255.255.0
  #exit
  subscriber default
  #exit
  administrator testadmin encrypted password fd01268373c5da85
  inspector testinspector encrypted password 148661a0bb12cd59
#exit
  port ethernet 5/1
  bind interface mgmt1 local
#exit
```

**Configuring Local-User Administrative Users**

The local user type supports ANSI T1.276-2003 password security protection. Local-user account information, such as passwords, password history, and lockout states, is maintained in /flash. This information is saved immediately in a separate local user database subject to AAA based authentication and is not used by the rest of the system. As such, configured local-user accounts are not visible with the rest of the system configuration.

---

**Important**

In release 20.0 and higher Trusted StarOS builds, the local user database is disabled. The Global Configuration mode `local-user` commands, and Exec mode `show local-user` and `update local-user` commands are unavailable. For additional information on Trusted builds, see the System Operation and Configuration chapter.

Use the example below to configure local-user administrative users:

```
configure
  local-user username name
end
```

Notes:

- Additional keyword options are available identify active administrators or place time thresholds on the administrator. Refer to the Command Line Interface Reference for more information about the `local-user username` command.

For additional information on the local-user database, see Updating and Downgrading the local-user Database, on page 47.

**Verifying Local-User Configuration**

Verify that the configuration was successful by entering the following command:

```
show local-user verbose
```

This command displays information on configured local-user administrative users. A sample output for this command appears below. In this example, a local-user named SAUser was configured.

<table>
<thead>
<tr>
<th>Username:</th>
<th>SAUser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auth Level:</td>
<td>secadmin</td>
</tr>
<tr>
<td>Last Login:</td>
<td>Never</td>
</tr>
<tr>
<td>Login Failures:</td>
<td>0</td>
</tr>
</tbody>
</table>
Password Expired: Yes  
Locked: No  
Suspended: No  
Lockout on Pw Aging: Yes  
Lockout on Login Fail: Yes

### Updating Local-User Database

Update the local-user (administrative) configuration by running the following Exec mode command. This command should be run immediately after creating, removing or editing administrative users.

`update local-user database`

### Updating and Downgrading the local-user Database

Prior to release 20.0, local-user passwords were hashed with the MD5 message digest-algorithm and saved in the local-user database. In release 20.0, PBKDF2 (Password Based Key Derivation Function - Version 2) is now used to derive a key of given length, based on entered data, salt and number of iterations. Local-user account passwords are hashed using the PBKDF2 method with a randomly generated salt coupled with a large number of iterations to make password storage more secure.

When upgrading to release 20.0, existing user passwords in the local-user database are not automatically upgraded from MD5 to PBKDF2 hashing (only hashed password values are stored). Since hash functions are one-way, it is not possible to derive user passwords from the stored hash values. Thus it is not possible to convert existing hashed passwords to strongly hashed passwords automatically.

To update the database, a Security Administrator must run the Exec mode `update local-user database` CLI command. When this command is executed, StarOS reads the database from the `/flash` directory, reconstructs the database in the new format, and writes it back to the disk.

The database upgrade process does not automatically convert MD5 hashed passwords into the PBKDF2 format. StarOS continues to authenticate users using the old encryption algorithm. It flags the users using the old encryption algorithm with a "Weak Hash" flag. This flag appears in the output of the `show local-user [verbose]` Exec mode CLI command. When users re-login with their credentials, StarOS verifies the entered password using the MD5 algorithm, then creates a new hash using the PBKDF2 algorithm and then saves the result in the database. StarOS then clears the "Weak Hash" flag for that user.

---

**Important**

Since hash functions are one-way, it is not possible to convert PBKDF2 hashed passwords to the MD5 format. The local-user database must be downgraded prior to reverting to StarOS releases prior to 20.0.

To downgrade the local-user database to use the MD5 hash algorithm, a Security Administrator must run the Exec mode `downgrade local-user database` command. StarOS prompts for confirmation and requests the Security Administrator to reenter a password. The entered password re-authenticates the user prior to executing the downgrade command. After verification, the password is hashed using the appropriate old/weak encryption algorithm and saved in the database to allow earlier versions of StarOS to authenticate the Security Administrator.

The downgrade process does not convert PBKDF2 hashed passwords to MD5 format. The downgrade process re-reads the database (from the `/flash` directory), reconstructs the database in the older format, and writes it back to the disk. Since the PBKDF2 hashed passwords cannot be converted to the MD5 hash algorithm, and earlier StarOS releases cannot parse the PBKDF2 encryption algorithm, StarOS suspends all those users encrypted via the PBKDF2 algorithm. Users encrypted via the MD5 algorithm ("Weak Hash" flag) can continue
to login with their credentials. After the system comes up with the earlier StarOS release, suspended users can be identified in the output of the `show local-user [verbose]` command.

To reactivate suspended users a Security Administrator can:

- Set temporary passwords for suspended users, using the Exec mode `password change local-user username` command.
- Reset the suspend flag for users, using the Configuration mode `no suspend local-user username` command.

### Restricting User Access to a Specified Root Directory

By default an admin user who has FTP/SFTP access can access and modify any files under the `/mnt/user/` directory. Access is granted on an "all-or-nothing" basis to the following directories: `/flash`, `/cdrom`, `/hd-raid`, `/records`, `/usb1` and `/usb2`.

An administrator or configuration administrator can create a list of SFTP subsystems with a file directory and access privilege. When a local user is created, the administrator assigns an SFTP subsystem. If the user's authorization level is not security admin or admin, the user can only access the subsystem with read-only privilege. This directory is used as the user's root directory. The information is set as environmental variables passed to the openssh_sftp-server.

You must create the SFTP root directory before associating it with local users, administrators and config administrators. You can create multiple SFTP directories; each directory can be assigned to one or more users.

#### Configuring an SFTP root Directory

The `subsystem sftp` command allows the assignment of an SFTP root directory and associated access privilege level.

```conf
configure context local
server sshd
    subsystem sftp | name sftp_name root-dir pathname mode { read-only | readwrite } ]
```

Notes:

- `sftp_name` is an alphanumeric string that uniquely identifies this subsystem.
- `pathname` specifies the root directory to which SFTP files can be transferred. Options include:
  - `/hd-raid/records/cdr`
  - `/flash`

#### Associating an SFTP root Directory with a Local User

The `local-user username` command allows an administrator to associate an SFTP root directory with a specified username.

```conf
configure
    local-user username user_name authorization-level level ftp sftp-server sftp_name password
```
Associating an SFTP root Directory with an Administrator

The administrator command allows an administrator to associate an SFTP root directory for a specified administrator.

configure context local
    administrator user_name password password ftp sftp-server sftp_name
exit

Associating an SFTP root Directory with a Config Administrator

The config-administrator command allows an administrator to associate an SFTP root directory with a specified configuration administrator.

configure context local
    config-administrator user_name password password ftp sftp-server sftp_name
exit

Configuring TACACS+ for System Administrative Users

This section describes TACACS+ (Terminal Access Controller Access Control System+) AAA (Authentication Authorization and Accounting) service functionality and configuration on the ASR 5x00.

Operation

TACACS+ is a secure, encrypted protocol. By remotely accessing TACACS+ servers that are provisioned with the administrative user account database, the ASR 5x00 can provide TACACS+ AAA services for system administrative users. TACACS+ is an enhanced version of the TACACS protocol that uses TCP instead of UDP.

The ASR 5x00 system serves as the TACACS+ Network Access Server (NAS). As the NAS the system requests TACACS+ AAA services on behalf of authorized system administrative users. For the authentication to succeed, the TACACS+ server must be in the same local context and network accessed by the system.

The system supports TACACS+ multiple-connection mode. In multiple-connection mode, a separate and private TCP connection to the TACACS+ server is opened and maintained for each session. When the TACACS+ session ends, the connection to the server is terminated.

TACACS+ is a system-wide function on the ASR 5x00. TACACS+ AAA service configuration is performed in TACACS Configuration Mode. Enabling the TACACS+ function is performed in the Global Configuration Mode. The system supports the configuration of up to three TACACS+ servers.

Once configured and enabled on the system, TACACS+ authentication is attempted first. By default, if TACACS+ authentication fails, the system then attempts to authenticate the user using non-TACACS+ AAA services, such as RADIUS.
For releases after 15.0 MR4, TACACS+ accounting (CLI event logging) will not be generated for Lawful Intercept users with privilege level set to 15 and 13.

## User Account Requirements

Before configuring TACACS+ AAA services, note the following TACACS+ server and StarOS user account provisioning requirements.

### TACACS+ User Account Requirements

The TACACS+ server must be provisioned with the following TACACS+ user account information:

- A list of known administrative users.
- The plain-text or encrypted password for each user.
- The name of the group to which each user belongs.
- A list of user groups.
- TACACS+ privilege levels and commands that are allowed/denied for each group.

TACACS+ privilege levels are stored as Attribute Value Pairs (AVPs) in the network's TACACS+ server database. Users are restricted to the set of commands associated with their privilege level. A mapping of TACACS+ privilege levels to ASR 5x00 CLI administrative roles and responsibilities is provided in the table below.

To display the default mapping of TACACS+ privilege levels to CLI administrative roles, run the Exec mode `show tacacs priv-lvl` command. The default mapping varies based on the StarOS release and build type.

TACACS+ priv-levels can be reconfigured from their default StarOS authorization values via the TACACS+ Configuration mode `priv-lvl` and `user-id` commands. For additional information, see the TACACS+ Configuration Mode Commands chapter of the Command Line Interface Reference.

### Important

In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

### StarOS User Account Requirements

TACACS+ users who are allowed administrative access to the system must have the following user account information defined in StarOS:

- username
- password
- administrative role and privileges
For instructions on defining users and administrative privileges on the system, refer to Configuring System Administrative Users.

Configuring TACACS+ AAA Services

This section provides an example of how to configure TACACS+ AAA services for administrative users on the system.

Caution

When configuring TACACS+ AAA services for the first time, the administrative user must use non-TACACS+ services to log into the ASR 5x00. Failure to do so will result in the TACACS+ user being denied access to the system.

Log in to the system using non-TACACS+ services.

Use the example below to configure TACACS+ AAA services on the system:

```
configure
tacacsmode
  server priority priority_number ip-address tacacs+srvr_ip_address
end
```

Note:

- **server priority priority_number**: Must be an integer from 1 to 3 (releases prior to 18.2) or 1 through 4 (releases 18.2+), that specifies the order in which this TACACS+ server will be tried for TACACS+ authentication. 1 is the highest priority, and 3 or 4 is the lowest. The priority number corresponds to a configured TACACS+ server.

- **ip-address**: Must be the IPv4 address of a valid TACACS+ server that will be used for authenticating administrative users accessing this system via TACACS+ AAA services.

- By default, the TACACS+ configuration will provide authentication, authorization, and accounting services.

Enable TACACS+ on the ASR 5x00:

```
configure
  aaa tacacs+
end
```

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

Important

For complete information on all TACACS+ Configuration Mode commands and options, refer to the TACACS Configuration Mode Commands chapter in the Command Line Reference.
Configuring TACACS+ for Non-local VPN Authentication

By default TACACS+ authentication is associated with login to the local context. TACACS+ authentication can also be configured for non-local context VPN logins. TACACS+ must configured and enabled with the option described below.

A *stop* keyword option is available for the TACACS+ Configuration mode *on-unknown-user* command. If TACACS+ is enabled with the command-keyword option, the VPN context name into which the user is attempting a login must match the VPN name specified in the username string. If the context name does not match, the login fails and exits out.

Without this option the login sequence will attempt to authenticate in another context via an alternative login method. For example, without the *on-unknown-user stop* configuration, an admin account could log into the local context via the non-local VPN context. However, with the *on-unknown-user stop* configuration, the local context login would not be attempted and the admin account login authentication would fail.

```
configure
tacacs mode
    on-unknown-user stop &quest;
end
```

Verifying the TACACS+ Configuration

This section describes how to verify the TACACS+ configuration.

Log out of the system CLI, then log back in using TACACS+ services.

---

**Important**

Once TACACS+ AAA services are configured and enabled on the ASR 5x00, the system first will try to authenticate the administrative user via TACACS+ AAA services. By default, if TACACS+ authentication fails, the system then continues with authentication using non-TACACS+ AAA services.

At the Exec Mode prompt, enter the following command:

```
show tacacs [ client | priv-lvl | session | summary ]
```

The output of the *show tacacs* commands provides summary information for each active TACACS+ session such as username, login time, login status, current session state and privilege level. Optional filter keywords provide additional information.

An example of this command’s output is provided below. In this example, a system administrative user named *asradmin* has successfully logged in to the system via TACACS+ AAA services.

```
active session #1:   
    login username       : asradmin
    login tty            : /dev/pts/1
    login server priority: 1
    current login status : pass
    current session state: user login complete
    current privilege level: 15
    remote client application: ssh
    remote client ip address: 111.11.11.11
    last server reply status : -1
    total TACACS+ sessions : 1
```
Configuring a Chassis Key

A chassis key should be configured for each system. This key is used to decrypt encrypted passwords found in configuration files.

Overview

The chassis key is used to encrypt and decrypt encrypted passwords in the configuration file. If two or more chassis are configured with the same chassis key value, the encrypted passwords can be decrypted by any of the chassis sharing the same chassis key value. As a corollary to this, a given chassis key value will not be able to decrypt passwords that were encrypted with a different chassis key value.

The chassis key is used to generate the chassis ID which is stored in a file and used as the master key for protecting sensitive data (such as passwords and secrets) in configuration files.

For release 15.0 and higher, the chassis ID is an SHA256 hash of the chassis key. The chassis key can be set by users through a CLI command or via the Quick Setup Wizard. If the chassis ID does not exist, a local MAC address is used to generate the chassis ID.

For release 19.2 and higher, the user must explicitly set the chassis key through the Quick Setup Wizard or CLI command. If it is not set, a default chassis ID using the local MAC address will not be generated. In the absence of a chassis key (and hence the chassis ID), sensitive data will not appear in a saved configuration file. The chassis ID is the SHA256 hash (encoded in base36 format) of the user entered chassis key plus a 32-byte secure random number. This assures that the chassis key and chassis ID have 32-byte entropy for key security.

If a chassis ID is not available encryption and decryption for sensitive data in configuration files will not work.

Configuring a New Chassis Key Value

CLI Commands

Only a user with Security Administrator privilege can execute the `chassis key value` and `chassis keycheck` commands.

Use the Exec mode `chassis key value key_string` command to enter a new chassis key.

The `key_string` is an alphanumeric string of 1 through 16 characters. The chassis key is stored as a one-way encrypted value, much like a password. For this reason, the chassis key value is never displayed in plain-text form.
The Exec mode `chassis keycheck key_string` command generates a one-way encrypted key value based on the entered `key_string`. The generated encrypted key value is compared against the encrypted key value of the previously entered chassis key value. If the encrypted values match, the command succeeds and keycheck passes. If the comparison fails, a message is displayed indicating that the key check has failed. If the default chassis key (MAC address) is currently being used, this key check will always fail since there will be no chassis key value to compare against.

Use the `chassis keycheck` command to verify whether multiple chassis share the same chassis key value.

---

**Important**

For release 19.2 and higher, in the absence of an existing chassis ID file the `chassis keycheck` command is hidden.

For additional information, refer to the *Exec Mode Commands* chapter in the *Command Line Interface Reference*.

Beginning with Release 15.0, the chassis ID will be generated from the chassis key using a more secure algorithm. The resulting 44-character chassis ID will be stored in the same file.

Release 14 and Release 15 chassis IDs will be in different formats. Release 15 will recognize a Release 14 chassis ID and consider it as valid. Upgrading from 14.x to 15.0 will not require changing the chassis ID or configuration file.

However, if the chassis key is reset in Release 15 through the Quick Setup Wizard or CLI command, a new chassis ID will be generated in Release 15 format (44 instead of 16 characters). Release 14 builds will not recognize the 44-character chassis ID. If the chassis is subsequently downgraded to Release 14, a new 16-character chassis ID will be generated. To accommodate the old key format, you must save the configuration file in pre-v12.2 format before the downgrade. If you attempt to load a v15 configuration file on the downgraded chassis, StarOS will not be able to decrypt the password/secrets stored in the configuration file.

For release 19.2 and higher, in a chassis where the chassis ID file already exists nothing is changed. However, if the chassis ID file is lost in both management cards, all existing configuration files become invalid. Entering a new chassis key that is the same as the original value will not resolve the issue because of the new method used to generate the chassis ID.

---

**Caution**

After setting a new chassis key, you must save the configuration before initiating a reload. See the *Verifying and Saving Your Configuration* chapter.

---

**Quick Setup Wizard**

The Quick Setup Wizard prompts the user to enter a chassis key value. If a chassis key value is not entered a default chassis is generated using the chassis' MAC address (releases prior to 20.0).

For releases 20.0 and higher, if the chassis ID file does not exist, the Quick Setup Wizard prompts the user to enter a chassis key. A default chassis ID is not generated if a chassis key is not entered.

To run the Quick Setup Wizard, execute the Exec mode `setup` command.

```
[local] host_name> setup
1. Do you wish to continue with the Quick Setup Wizard[yes/no]: y
2. Enable basic configuration[yes/no]: y
3. Change chassis key value[yes/no]: y
4. New chassis key value: key_string
```
Configuring MIO/UMIO Port Redundancy

Port redundancy for MIO cards provides an added level of redundancy that minimizes the impact of network failures that occur external to the system. Examples include switch or router port failures, disconnected or cut cables, or other external faults that cause a link down error.

Caution

To ensure that system card and port-level redundancy mechanisms function properly, disable the Spanning Tree protocol on devices connected directly to any system port. Failure to turn off the Spanning Tree protocol may result in failures in the redundancy mechanisms or service outage.

By default, the system provides port-level redundancy when a failure occurs, or you issue the port switch to command. In this mode, the ports on active and standby MIO/UMIO cards have the same MAC address, but since only one of these ports may be active at any one time there are no conflicts. This eliminates the need to transfer MAC addresses and send gratuitous ARPs in port failover situations. Instead, for Ethernet ports, three Ethernet broadcast packets containing the source MAC address are sent so that the external network equipment (switch, bridge, or other device) can re-learn the information after the topology change. However, if card removal is detected, the system sends out gratuitous ARPs to the network because of the MAC address change that occurred on the specific port.

With port redundancy, if a failover occurs, only the specific port(s) become active. For example, if port 5/1 fails, then port 6/1 becomes active, while all other active ports on the line card in slot 5 remain in the same active state. In port failover situations, use the show port table command to check that ports are active on both cards and that both cards are active.

Take care when administratively disabling a port that is one of a redundant pair. A redundant pair comprises both the active and standby ports—for example 5/1 and 6/1. If 5/1 is active, administratively disabling 5/1 through the CLI does not make 6/1 active. It disables both 5/1 and 6/1 because an action on one port has the same effect on both. Refer to Creating and Configuring Ethernet Interfaces and Ports in System Interface and Port Configuration Procedures.

With automatic card-level redundancy, there is no port-level redundancy in an MIO/UMIO failover. The standby MIO/UMIO becomes active and all ports on that card become active. The system automatically copies all the MAC addresses and configuration parameters used by the failed MIO/UMIO to its redundant counterpart. The ports on MIOs keep their original MAC addresses, and the system automatically copies the failed MIO/UMIO’s configuration parameters to its redundant counterpart.

Port redundancy can be configured to be revertive or non-revertive. With revertive redundancy service is returned to the original port when service is restored.
This feature requires specific network topologies to work properly. The network must have redundant switching components or other devices that the system is connected to. The following diagrams show examples of a redundant switching topologies and how the system reacts to various external network device scenarios.

**Figure 6: Network Topology Example Using MIO/UMIO Port Redundancy**

In the example above, an Ethernet cable is cut or unplugged, causing the link to go down. When this event occurs, the system, with port-mode redundancy enabled, recognizes the link down state and makes port 6/1 the active port. The switching device, using some port redundancy scheme, recognizes the failure and enables
the port on the secondary switch to which the MIO/UMIO in slot 6 is connected, allowing it to redirect and transport data.

**Figure 8: Port Redundancy Failover in External Network Device Failure Scenario**

In the example above, a switch failure causes a link down state on all ports connected to that switch. This failure causes all redundant ports on the line card in slot 6 to move into the active state and utilize the redundant switch.

### Configuring MIO/UMIO Port Redundancy Auto-Recovery

You can configure a port auto-recovery feature. When a port failure occurs and the preferred port is returned to service (link is up), control is automatically returned to that port. By default, ports are in a non-revertive state, meaning that no ports are preferred; a manual port switch is required to return use to the original port.

---

**Important**

This feature is applied on a per port basis (via the `preferred slot` keyword), allowing you to configure specific ports to be used on individual MIO cards. For example, you could configure ports 10 through 19 as preferred on the MIO/UMIO in slot 5, and configure ports 20 through 29 as the preferred ports on the MIO/UMIO in slot 6.

Use the following example to configure a preferred port for revertive, automatic return to service when a problem has cleared:

```
configure
  port ethernet slot#/port#
    preferred slot slot#
end
```

**Notes**

- If you do specify a preference, redundancy is revertive to the specified card. If you do not specify a preference, redundancy is non-revertive.
- Repeat for each additional port that you want to make preferred.

Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.
Verifying Port Redundancy Auto-Recovery

Verify port information by entering the following command

```
show port info slot#/port#
```

*slot#* is the chassis slot number of the MIO/UMIO card on which the physical port resides.

*port#* is the physical port on the MIO/UMIO.

The following shows a sample output of this command for port 1 on the MIO/UMIO in slot 5:

```
[local] host_name# show port info 5/1
Port: 5/1
  Port Type : 1000 Ethernet
  Role : Management Port
  Description : (None Set)
  Redundancy Mode : Port Mode
  Redundant With : 6/1
  Preferred Port : Non-Revertive
  Physical ifIndex : 83951616
  Administrative State : Enabled
  Configured Duplex : Auto
  Configured Speed : Auto
  Configured Flow Control : Enabled
  Interface MAC Address : 02-05-47-B8-2F-41
  Fixed MAC Address : 02-05-47-B8-2F-41
  Link State : Up
  Link Duplex : Full
  Link Speed : 1000 Mb
  Flow Control : Disabled
  Link Aggregation Group : None
  Logical ifIndex : 83951617
  Operational State : Up, Active
```

Configuring Data Processing Card Availability

As discussed in the *Understanding the System Boot Process* section of *Understanding System Operation and Configuration*, when the system initially boots up, all installed DPC/UDPCs or DPC2/UDPC2s are placed into standby mode. You must activate some of these cards in order to configure and use them for session processing. One DPC/UDPC or DPC2/UDPC2 may remain in standby mode for redundancy.

This section describes how to activate DPC/UDPCs or DPC2/UDPC2s and specify their redundancy.

---

**Important**

Refer to the *ASR 5500 Installation Guide* for information about system hardware configurations and redundancy.

Enter the following command to check the operational status of all DPC types:

```
show card table
```

This command lists the DPC types installed in the system by their slot number, their operational status, and whether or not the card is a single point of failure (SPOF).

Use the following example to configure DPC/UDPC or DPC2/UDPC2 availability:

```
configure
card slot#
  mode { active | standby }
end
```
Notes:

- When activating cards, remember to keep at least one DPC/UDPC or DPC2/UDPC2 in standby mode for redundancy.
- Repeat for every other DPC/UDPC or DPC2/UDPC2 in the chassis that you wish to activate.

Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

**Verifying Card Configurations**

Verify that the configuration was successful. Enter the following command:

```bash
show card table
```

Any DPC/UDPC or DPC2/UDPC2 that you made active should now have an operational status of *Active*.

**Enabling Automatic Reset of FSC Fabric**

By default if an excessive number of discarded fabric egress packets occurred in the switch fabric, a manual reset of the Fabric Storage Card(s) is required for fabric recovery.

You can optionally enable automatic resets of FSCs if an excessive number of discarded fabric egress packets is detected.

A Global Configuration mode *fabric fsc-auto-recover* command enables or disables automatic FSC resets upon detection of an excessive number of discarded fabric egress packets.

The following command sequence enables this feature:

```bash
configure
    fabric fsc-auto-recovery { disable | enable } [ max-attempts [ number_attempts | unlimited ] ]
end
```

*max-attempts [ number_attempts | unlimited ]* specifies how many times StarOS will attempt to reset each FSC as an integer from 1 to 99 or unlimited (will not stop until FSC is reset). The default setting is 1.

---

**Important**

To enable this feature, you must first configure the Fabric Egress Drop Threshold via the Global Configuration mode *fabric egress drop-threshold* command.

**Configuring ASR 5500 Link Aggregation**

A Link Aggregation Group (LAG) works by exchanging control packets via Link Aggregation Control Protocol (LACP) over configured physical ports with peers to reach agreement on an aggregation of links as defined in IEEE 802.3ad. The LAG sends and receives the control packets directly on physical ports.

Link aggregation (also called trunking or bonding) provides higher total bandwidth, auto-negotiation, and recovery by combining parallel network links between devices as a single link. A large file is guaranteed to be sent over one of the links, which removes the need to address out-of-order packets.
LAG and Master Port

Logical port configurations (VLAN and binding) are defined in the master port of the LAG. If the master port is removed because of a card removal/failure, another member port becomes the master port (resulting in VPN binding change and outage), unless there is a redundant master port available.

---

**Important**
The master port on which VLAN can be created for VPN binding must always be configured on the active/master MIO/UMIO. The redundancy between the MIO/UMIO in slot 5 and the MIO/UMIO in slot 6 automatically causes both ports to be the master with the same VLANs configured and active.

---

LAG and Port Redundancy

ASR 5500 LAG implementation assumes that:

- LAG ports on MIO/UMIO-slot 5 and MIO/UMIO-slot 6 are connected to two Ethernet switches.
- LAG ports on MIO/UMIO-slot 5 and MIO/UMIO-slot 6 are both active at the same time.
- Ports on MIO/UMIO-slot 5 and MIO/UMIO-slot 6 are redundant with each other.

All ports in a LAG can be auto-switched to another MIO/UMIO when certain active port counts or bandwidth thresholds are crossed.

LAG and Multiple Switches

This feature connects subscriber traffic ports on MIOs to ports on Ethernet switches. A port failure/switch forces all ports in a LAG to switch to the other MIO/UMIO when a specified threshold is crossed. This works in a way similar to the auto-switch feature for port redundancy. LACP runs between the ASR 5500 and the Ethernet switch, exchanging relevant pieces of information, such as health status.

The following table summarizes typical LAG functionality on an MIO/UMIO card.

**Table 4: MIO/UMIO LAG Functionality**

<table>
<thead>
<tr>
<th>ASR 5500</th>
<th>LAGID</th>
<th>Ethernet Switch A</th>
<th>Ethernet Switch B</th>
</tr>
</thead>
<tbody>
<tr>
<td>MIO/UMIO Port 11</td>
<td>1</td>
<td>Port 1</td>
<td>----</td>
</tr>
<tr>
<td>MIO/UMIO Port 12</td>
<td>1</td>
<td>Port 2</td>
<td>----</td>
</tr>
<tr>
<td>MIO/UMIO Port 13</td>
<td>1</td>
<td>----</td>
<td>Port 1</td>
</tr>
</tbody>
</table>

Multiplex Switches with L2 Redundancy

To handle the implementation of LACP without requiring standby ports to pass LACP packets, two separate instances of LACP are started on redundant cards. The two LACP instances and port link state are monitored to determine whether to initiate an auto-switch (including automatic L2 port switch).
The figure below shows an LAG established across two MIO/UMIO daughter card ports with L2 redundancy.

Figure 9: LAG with L2 Redundancy, Two Ethernet Switches

An LACP implementation with L2 redundancy cannot pass traffic even though standby ports have link up. For example, with two MIO/UMIO cards connected to two different Ethernet switches and all ports in the same LAG, failure of ports would not trigger a LAG switch until the active port number ratio flipped (more ports down than up).

Port States for Auto-Switch

Ports are classified in one of four states to determine whether to start auto-switching. See the table below. For counters, State(x) represents the number of ports on a card in that state.

Table 5: Auto-Switch Port States

<table>
<thead>
<tr>
<th>State</th>
<th>Counter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link</td>
<td>L(x)</td>
<td>Physical link up</td>
</tr>
<tr>
<td>Standby</td>
<td>S(x)</td>
<td>Link up but in standby mode</td>
</tr>
<tr>
<td>Waiting</td>
<td>W(x)</td>
<td>Waiting for Link Aggregation Control Protocol negotiation</td>
</tr>
<tr>
<td>Aggregated</td>
<td>A(x)</td>
<td>Aggregation formed</td>
</tr>
</tbody>
</table>

Hold Time

Once the LAG manager switches to another LACP instance, it does not consider another change for a short period to let link and LACP negotiation settle down. This "hold time" is configurable.

The LAG manager also enters/extends the hold period when an administrator manually switches ports to trigger a card switch.

Preferred Slot

You can define which card is preferred per LAG group as a preferred slot. When a preferred MIO/UMIO slot is specified, it is selected for the initial timeout period to make the selection of a switch less random.
Port preference is not allowed in this mode.

**Auto-Switch Criteria**

The following criteria determine the switching of card x to card y to provide better bandwidth while allowing manual intervention. The evaluation of the criteria occurs outside of the hold period.

Ports are automatically switched from card x to card y when \( A(y) = 1 \), at least one port is in aggregated state on card y, and one of the following conditions is true (in order of precedence):

- \( L(x) \leq L(y) \): Less ports with link Up on card x than card y
- \( S(x) \geq S(y) \): More ports in Standby state on card x than card y
- \( W(x) \geq W(y) \): More ports in Waiting state on card x than card y
- \( A(x) \leq A(y) \): Fewer ports in Aggregated state on card x than card y
- Card y is preferred
- Card y is selected.

**Link Aggregation Control**

One port in an aggregation group is configured as a master so that all traffic (except control traffic) in the aggregation group logically passes through this port. It is recommended that you configure link-aggregation on the master port first when enabling LAG, and unconfigure the master port last when disabling LAG.

The following command creates link aggregation group \( N \) with port \( \text{slot#/port#} \) as master. Only one master port is allowed for a group. \( N \) must be in the range of \([1–255]\).

```
configure
  port ethernet \text{slot#/port#}
  link-aggregation master group \( N \)
exit
```

**Important**

Link Aggregation Control Protocol (LACP) starts running only when the master port is enabled.

Use the following command to add a port as member of link aggregation group number \( N \) only if the master port is assigned. Otherwise, it is added to the group when the master port is assigned:

```
port ethernet \text{slot#/port#}
  link-aggregation member group \( N \)
exit
```

**Important**

The VPN can only bind the master port, and a VLAN can only be created on the master port. A failure message is generated if you attempt to bind to a link aggregation member port.

Each system that participates in link aggregation has a unique system ID that consists of a two-byte priority (where the lowest number [0] has the highest priority) and a six-byte MAC address derived from the first
port's MAC address. The following command sets the system priority used to form the system ID. $P$ is a hex in the range $[0x0000..0xFFFF]$. The default is $0x8000$.

```bash
card slot#
    link-aggregation system-priority $P$
```

Ports in a system are assigned keys. The group number maps directly to the key, whereby only ports with the same key can be aggregated. Ports on each side of the link use a different aggregation key.

The system ID, port key and port ID of two peers form the Link Aggregation Group Identifier (LAGID). You can aggregate links having the same LAGID. Systems are often configured initially with each port in its own aggregation (requiring a separate key per port), or with all ports in the same aggregation (a single key for all ports). Negotiation via LACP would qualify the actual aggregation.

Systems exchange information about system ID, port key and port ID with peers across the physical links using LACP.

LACP packets are defined with the Slow Protocol format. Each system sends out its own ("actor") information and its last received information about its peer ("partner") over the physical link.

Use the following commands to set the LACP parameters. LACP can run in active mode to send LACP packets periodically, or in passive mode, in which it only responds to LACP packets it receives.

LACP can send packets at either a auto (30s) or fast (1s) rate. The defaults for this release are Active and Auto; see the sample configuration below:

```bash
config
    port ethernet slot#/port#
        link-aggregation lacp { active | passive } [ rate { auto | fast } | timeout { long | short } ]
```

Peers send out LACP packets when the state changes or if a difference is found from a received LACP packet about its own state.

Corresponding ports on an MIO/UMIO redundant pair cannot be active at the same time. Redundant ports share the same MAC address, so after a failover is resolved, the original port rejoins the link aggregation group.

### Minimum Links

A minimum links option specifies that a Link Aggregation Group (LAG) is up (usable) only when a minimum number of links are available for aggregation. This guarantees that a minimum amount of bandwidth is available for use.

When this feature is enabled, a LAG is not usable when the number of links in a LAG goes below the configured min-link value. Switchover to another LAG bundle (if available) automatically occurs when the number of links in the current active bundle goes below the configured min-link value.

Use the `min-link` keyword option in the Global Configuration mode `link-aggregation` command to enable this feature.

```bash
configure
    port ethernet slot/port
        link-aggregation master { global | group } number
            min-link number_links
        end
```
Redundancy Options

For L2 redundancy set the following option on the master port for use with the whole group:

```
link-aggregation redundancy standard [hold-time sec ] [preferred slot { card_number | none }] 
```

Standard redundancy treats all cards in the group as one group.

Horizontal Link Aggregation with Two Ethernet Switches

When a LAG contains two sets of ports each connecting to a different switch, the operator has the ability to specify the slot/port (connected to the destination switch) when switching ports.

The Exec mode `link-aggregation port switch to slot/port` command configures this option. The `slot/port` is any valid port connected to the destination switch. The following criteria apply to the setting of this option:

- `slot/port` must support LAG.
- `slot/port` must be configured with LAG.
- `slot/port` must not be already actively distributing
- `slot/port` must have negotiated a link aggregation partner in standard mode.
- `slot/port`'s partner must have an equal or higher in standard mode.
- `slot/port`'s partner bundle must have equal or higher bandwidth in standard mode.
- Switching to `slot/port` must not violate preference within hold-time in standard mode.

Non-Redundant (Active-Active) LAG

LAG can be deployed in a non-redundant mode in which the ports from both MIO/UMIO cards are connected to the same switch.

**Figure 10: Non-Redundant LAG Configuration with Single LAG Group**

In the above configuration, there is a single, primary LAG. All ports work as a single bundle of ports that distribute the traffic.
With this mode operation, automatic ASR 5500 port redundancy is lost.  

To achieve redundancy you must configure a second non-redundant LAG. You can use a higher layer load balancing mechanisms such as ECMP (Equal Cost Multiple Path) routing to uniformly distribute the traffic across two LAG groups.

When one MIO/UMIO fails, half the ports from both the LAG groups will be available for distribution of the traffic from the other MIO/UMIO.

Figure 11: Non-Redundant LAG Configuration with ECMP

Configuring a second LAG group is not mandatory, but is the usual approach for achieving redundancy with this mode of LAG.

However, if the aggregating ports are loaded with more than 50% of their capacity and an MIO/UMIO failure/switchover occurs, the ASR 5500 configured port capacity is oversubscribed and an indeterminate amount of sessions are dropped and traffic lost.

**Link Aggregation Status**

To check the status of link aggregation, use the following commands:

- `show port table`
- `show port info slot/port`

A single character is used to display LAG physical port status in the output of the `show port table` command. See the table below.

<table>
<thead>
<tr>
<th>Display</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LA+</td>
<td>Port is actively used for distributing (transmit nd recieve data).</td>
</tr>
<tr>
<td>LA-</td>
<td>Port failed to negotiate LACP.</td>
</tr>
<tr>
<td>LA~ (tilde)</td>
<td>Port negotiated LACP but another peer was selected.</td>
</tr>
<tr>
<td>LA*</td>
<td>Port is (re)negotiating LACP.</td>
</tr>
</tbody>
</table>
You can dedicate a DPC/UDPC or DPC2/UDPC2, or MIO/UMIO to function as a demux card. Demux is a generic term for signal demultiplexing tasks. These are the tasks responsible for parsing call setup (signaling packets) and distributing the calls internally. For this reason there almost as many tasks running on a demux card as there are services.

The vpnmgr tasks responsible for each context also run on the demux card. The number of vpnmgr tasks correspond to the number of contexts. A vpnmgr is responsible for IP address assignment to mobile equipment, IP routing (such as BGP, OSPF), as well as a variety of associated tasks.

Overview

Designating a DPC/UDPC or DPC2/UDPC2, or MIO/UMIO as a demux card frees up resources for session handling, which has the potential to increase system throughput. However, there is no increased support in total subscriber capacity due to other system resource restrictions.

This feature is disabled by default and can be enabled via the Global Configuration mode `require demux` command. It is only supported for a limited number of products. Refer to the product Administration Guide for additional information.

To support this feature session recovery must also be enabled via the Global Configuration mode `require session recovery` command.

Important

After enabling demux card and session recovery, you must save the configuration and reboot the ASR 5500 to enable this feature.

Caution

Enabling the Demux on MIO/UMIO feature changes resource allocations within the system. This directly impacts an upgrade or downgrade between StarOS versions in ICSR configurations. Contact Cisco TAC for procedural assistance prior to upgrading or downgrading your ICSR deployment.

MIO Demux Restrictions

The following restrictions apply when enabling an MIO/UMIO as a demux card:

- The `require demux management-card` command must be configured before any service or contexts have been created on the system. The command will not execute after a mode of operation has been selected for the chassis.
• Only the following services currently support the designation of an MIO/UMIO card for demux functions: GGSN, SGW, PGW, HA, SAE-GW and L2TP LNS. These services are supported only when they are deployed as consumer gateways.

• SGSN, MME, HNBGW, HeNBGW, SaMOG, PDG, PDIF, ePDG, IPSG, PDSN, HSGW, L2TP LAC, NEMO, FA, and WSG are not supported. Enterprise or corporate gateways (GGSN, HA, PGW, etc.) are also not supported.

• You should not enable demux functionality on MIO/UMIO for configurations that require a large number of tunnels.

• After the ASR 5500 has booted with demux functions running on an MIO/UMIO, you cannot configure non-supported services. A maximum of eight Demux Managers are supported. Any attempt to add more than eight Demux Managers will be blocked.

• Service/products requiring a large number of VPN Managers, VRFs and/or Demux Managers must not enable demux functions on an MIO/UMIO.

• With demux functions running on an MIO/UMIO, the ASR 5500 supports a maximum of 10 contexts, 64 interfaces per context, and 250 VRFs per system.

• ICSR upgrades require compatible configurations and Methods of Procedure (MOPs).

Implementation of this feature assumes that CEPS (Call Events Per Second) and the number of subscribers will remain constant, and only the data rate will increase. This ensures that the CPU demand will not increase on the MIO/UMIO.

Important Contact Cisco TAC for additional assistance when assessing the impact to system configurations when enabling the Demux on MIO/UMIO feature.

---

**Configuration**

*For releases prior to 15.0*, to configure a DPC/UDPC as a demux card enter the following CLI commands:
```plaintext
configure
  require demux card
end
```

*For release 15.0+*, to configure a DPC/UDPC as a demux card enter the following CLI commands:
```plaintext
configure
  require demux processing-card
end
```

*For release 18.0+*, to configure a DPC/2UDPC2 as a demux card enter the following CLI commands:
```plaintext
configure
  require demux processing-card
end
```

To configure an MIO/UMIO as a demux card enter the following CLI commands:
```plaintext
configure
  require demux management-card
end
```
Management Settings

This chapter provides instructions for configuring Object Request Broker Element Management (ORBEM) and Simple Network Management Protocol (SNMP) options.

This chapter includes the following sections:

• ORBEM, page 69
• SNMP MIB Browser, page 71
• SNMP Support, page 74

ORBEM

The system can be managed by a Common Object Broker Request Architecture (CORBA)-based, Element Management System (EMS).

You must configure the ORBEM settings on the ASR 5x00 that allow the system to communicate with the server running the EMS application.

Important

Commands used in the configuration samples in this section provide base functionality. The most common commands and keyword options are presented. In many cases, other optional commands and keyword options are available. Refer to the Command Line Interface Reference for detailed information about all commands.

To configure the system to communicate with an EMS:

Step 1 Set client ID parameters and configure the STOP/TCP port settings by applying the example configuration in Configuring ORBEM Client and Port Parameters, on page 70
Step 2 Configure Internet Inter-ORB Protocol (IIOP) transport parameters by applying the example configuration in Configuring IIOP Transport Parameters, on page 70
Step 3 View your new ORBEM configuration by following the steps in Verifying ORBEM Parameters, on page 71
Step 4 Save the configuration as described in Verifying and Saving Your Configuration.
Configuring ORBEM Client and Port Parameters

Use the following example to set client ID parameters and configure the SIOP/TCP port settings:

```
configure
  orbem
    client_id  encrypted_password  password
    max-attempt  number
    session-timeout  time
    siop-port  port_number
    event-notif-siop-port  siop_notif_port
    event-notif-service
end
```

Notes:

- You can issue the client id command multiple times to configure multiple clients.
- If a client ID is de-activated due to reaching the configured maximum number of attempts, use the `activate client_id` command to reactivate it.
- If a firewall exists between the system and the EMS, open the SIOP port number and the TCP port number 15011.
- If the ORB Notification Service is enabled via the `event-notif-service` command, you can set filters to determine which events are to be sent. By default, the Service sends all error and higher level events, "info" level events for the ORBS facility, CLI command logs, and license change logs. Optionally, configure a filter by including the `event-notif-service filter` command. Enter this command for each filter you need to configure.

Configuring IIOP Transport Parameters

Use the following example to configure Internet Inter-ORB Protocol (IIOP) transport parameters that enable ORB-based management to be performed over the network:

```
configure
  orbem
    iiop-transport
    iiop-port  iiop_port_number
    event-notif-iiop-port  iiop_notif_port
end
```

Notes:

- If you are using the Secure Sockets Layer (SSL) option, do not enable the IIOP transport parameter.
- You configure the ORBEM interface to use SSL by specifying a certificate and private key.
Verifying ORBEM Parameters

Step 1  Run the `show orbem client table` command to verify that the client was configured properly. This command lists the configured ORBEM clients and displays their state and privileges.

Step 2  Run the `show orbem status` command to verify the ORBEM parameter configuration. The following displays a sample of this command's output.

```
Service State : On
Management Functions : FCAPS
IOP Address : 192.168.1.150
SSL Port : 14131
TCP Port : 14132
Notification SSL Port : 7777
Notification TCP Port : 7778
Session Timeout : 86400 secs
Max Login Attempts : 5
IIOP Transport : On
Notification : On
Debug Level : Off
IDL Version Check : On
Number of Current Sessions : 1
Number of Event Channels Open : 0
Number of Operations Completed : 2895
Number of Events Processed : 0
Avg Operation Processing time : 87214 usecs
    (last 1000) : 87950 usecs
```

SNMP MIB Browser

This section provides instructions to access the latest Cisco Starent MIB files using a MIB Browser. An updated MIB file accompanies every StarOS release. For assistance to set up an account and access files, please contact your Cisco sales or service representative for additional information.

A MIB Browser allows the user to pull out data from SNMP enabled devices. You can load standard and propriety MIBs. The tool allows the user to see the MIB data in a readable format and also offers the ability to search for a specific OID. The Browser displays all of the MIBs in a MIB tree which makes it easy to find and identify all Objects, Traps or Conformances.
Use the following procedure to view the SNMP MIBs for a specific StarOS build:

**Step 1** Contact Cisco sales or a service representative, to obtain access to the MIB files for a specific StarOS release.

**Step 2** Download the compressed companion file to a folder on your desktop. The file name follows the convention:

```
companion_xx.x.x.tgz
```

**Step 3** Open the companion file, unzip it and extract it to the same folder.

**Step 4** Double click on the new `companion-xx.x.x.xxxxx` file folder.

**Step 5** Unzip and extract the `companion-xx.x.x.xxxxx.tar` file.

**Step 6** From your MIB browser, search for and open the `starent.my` file within the `.tar` file. You can use any SNMP MIB Browser that allows you to compile a MIB `.my` file before viewing it.

**Step 7** To compile the MIB file, click on the STARENT-MIB file and select **File > Open**.

The STARENT-MIB.vosmi file opens.
In the example below the MIB Browser presents a tree diagram that allows you to display details for each Object, Trap and Conformance. The example below includes the OID number and trap details for the starCardPACMigrateFailed trap.

The SNMP MIB browser allows you to search for specific MIBs. You can search for a specific OID (object identifier) to find a specific MIB entry.

**Important** For information on SNMP MIBs changes for a specific release, refer to the *SNMP MIB Changes in Release xx* chapter of the appropriate version of the to the *Release Change Reference*. 
SNMP Support

The system uses the SNMP to send traps or events to the EMS server or an alarm server on the network. You must configure SNMP settings to communicate with those devices.

Important

Commands used in the configuration samples in this section provide base functionality. The most common commands and keyword options are presented. In many cases, other optional commands and keyword options are available. Refer to the Command Line Interface Reference for complete information.

The SNMP MIB Reference describes the MIBs and SNMP traps supported by the ASR 5x00 platform.

To configure the system to communicate with the EMS server or an alarm server:

Step 1 Set SNMP parameters such as UDP port, and alarm server target by applying the example configuration in Configuring SNMP and Alarm Server Parameters, on page 74

Step 2 To view your new SNMP configuration, follow the steps in Verifying SNMP Parameters, on page 75

Step 3 Save the configuration as described in Verifying and Saving Your Configuration.

Configuring SNMP and Alarm Server Parameters

Use the following example to set SNMP and alarm server parameters:

```
configure
    system contact contact_name
    system location location_name
    snmp authentication-failure-trap
    snmp community community_string
    snmp server port port_number
    snmp target name ip_address
    snmp engine-id local id_string
    snmp notif-threshold value low low_value period time_period
    snmp user user_name
    snmp mib mib_name
    snmp runtime-debug [ debug-tokens token_id token_id token_id ... token_id ]
end
```

Notes:

- The system contact is the name of the person to contact when traps are generated that indicate an error condition.

- An snmp community string is a password that allows access to system management information bases (MIBs).
The system can send SNMPv1, SNMPv2c, or SNMPv3 traps to numerous target devices. However, an EMS may only process SNMP version 1 (SNMPv1) and SNMP version 2c (SNMPv2c) traps. If the SNMP target you are configuring is the EMS application, use the `snmp target` command to configure use of version 1 or version 2c. Issue this command as many times as you need to configure multiple targets. If you configure multiple targets, generated alarms are sent to every configured target.

The `snmp notif-threshold` command configures the number of SNMP notifications that need to be generated for a given event and the number of seconds in the monitoring window size (default = 300), before the notification is propagated to the SNMP users (default = 300).

The `snmp engine-id local` command is optional. It is only required if your network requires SNMP v3 support. The engine ID uniquely identifies the SNMP engine and associated SNMP entities, thus providing a security association between the two for the sending and receiving of data.

The `snmp user` name is for SNMP v3 and is optional. There are numerous keyword options associated with this command.

Use the `snmp mib` command to enable other industry standard and Cisco MIBs. By default only the STARENT-MIB is enabled.

By default SNMP runtime debugging always runs and consumes CPU cycles for event logging. To control CPU usage you can set `no snmp runtime-debug` to disable runtime debugging. An option to this command allows you to specify SNMP token values that will locate and parse specified MIBs.

---

**Important** SNMPv3 traps may not be supported by some EMS applications.

---

### Verifying SNMP Parameters

**Step 1** Run the `show snmp server` command to verify that the SNMP server information is correctly configured. The following displays a sample output of this command.

**SNMP Server Configuration:**
- `Server State`: enabled
- `SNMP Port`: 161
- `sysLocation`: chicago
- `sysContact`: admin
- `authenticationFail traps`: Enabled
- `EngineID`: 123456789
- `Alert Threshold`: 100 alerts in 300 seconds
- `Alert Low Threshold`: 20 alerts in 300 seconds

**SNMP Agent Mib Configuration:**
- `STARENT-MIB`: Enabled
- `IF-MIB`: Disabled
- `ENTITY-MIB`: Disabled
- `ENTITY-STATE-MIB`: Disabled
- `ENTITY-SENSORE-MIB`: Disabled
- `HOST-RESOURCES-MIB`: Disabled
- `CISCO-MOBILE-WIRELESS-SERVICE-MIB`: Disabled
- `CISCO-ENTITY-DISPLAY-MIB`: Disabled
CISCO-PROCESS-MIB : Disabled
CISCO-ENTITY-FRU-CONTROL-MIB : Disabled

Step 2 Verify that the SNMP community(ies) were configured properly by entering the following command:
show snmp communities
The output of this command lists the configured SNMP communities and their corresponding access levels.

Step 3 Verify that the SNMP transports are configured properly by entering the following command:
show snmp transports
The following displays a sample output:

Target Name: rms1
IP Address: 192.168.1.200
Port: 162
Default: Default
Security Name: public
Version: 1
Security: 
View: 
Notif Type: traps

Controlling SNMP Trap Generation

The system uses SNMP traps (notifications) to indicate that certain events have occurred. By default, the system enables the generation of all traps. However, you can disable individual traps to allow only traps of a certain type or alarm level to be generated. This section provides instructions for disabling/enabling SNMP traps.

Important

Commands used in the configuration samples in this section provide base functionality. The most common commands and keyword options are presented. In many cases, other optional commands and keyword options are available. Refer to the Command Line Interface Reference for complete information regarding all commands.

To configure SNMP trap generation:

Step 1 Set parameters by applying the following example configuration:

```
configure
snmp trap suppress
snmp trap suppress trap_name1 trap_name2 ... trap_nameN
```

If at a later time you wish to re-enable a trap that was previously suppressed, use the `snmp trap enable` command.

Step 2 Save the configuration as described in Verifying and Saving Your Configuration.
Verifying and Saving Your Configuration

This chapter describes how to save your system configuration.

- Verifying the Configuration, page 77
- Synchronizing File Systems, page 79
- Saving the Configuration, page 79

Verifying the Configuration

You can use a number of commands to verify the configuration of your feature, service, or system. Many are hierarchical in their implementation and some are specific to portions of or specific lines in the configuration file.

Feature Configuration

In many configurations, you have to set and verify specific features. An example includes IP address pool configuration. Using the example below, enter the listed commands to verify proper feature configuration.

Enter the `show ip pool` command to display the IP address pool configuration. The output from this command should look similar to the sample shown below. In this example, all IP pools were configured in the `isp1` context.

```
context : isp1:
+----Type: (P) - Public    (R) - Private
    |    (S) - Static    (E) - Resource
    | +----State: (G) - Good    (D) - Pending Delete    (R) - Resizing
    || +----Priority: 0..10 (Highest (0) .. Lowest (10))
    ||| +----Busyout: (B) - Busyout configured
    ||vvvvvv Pool Name Start Address Mask/End Address Used Avail
    ----- --------------------- -------------- --------------- ----------------
    FG00 ipsec 12.12.12.0 255.255.255.0 0 254
    FG00 pool1 10.10.0.0 255.255.0.0 0 65534
    SG00 vpopool 192.168.1.250 92.168.1.254 0 5

Total Pool Count: 5
```
To configure features on the system, use the `show` commands specifically for these features. Refer to the *Exec Mode show Commands* chapter in the *Command Line Interface Reference* for complete information.

### Service Configuration

Verify that your service was created and configured properly by entering the following command:

```
show service_type service_name
```

The output is a concise listing of the service parameter settings similar to the sample displayed below. In this example, a P-GW service called `pgw` is configured.

```
Service name : pgw1
Service-Id : 1
Context : test1
Status : STARTED
Restart Counter : 8
EGTP Service : egtp1
LMA Service : Not defined
Session-Delete-Delay Timer : Enabled
Session-Delete-Delay timeout : 10000 (msecs)
PLMN ID List : MCC: 100, MNC: 99
Newcall Policy : None
```

### Context Configuration

Verify that your context was created and configured properly by entering the `show context name name` command.

The output shows the active context. Its ID is similar to the sample displayed below. In this example, a context named `test1` is configured.

```
Context Name ContextID State
---------- -------- -----
test1 2 Active
```

### System Configuration

Verify that your entire configuration file was created and configured properly by entering the `show configuration` command.

This command displays the entire configuration including the context and service configurations defined above.

### Finding Configuration Errors

Identify errors in your configuration file by entering the `show configuration errors` command.

This command displays errors it finds within the configuration. For example, if you have created a service named "service1", but entered it as "srv1" in another part of the configuration, the system displays this error.
You must refine this command to specify particular sections of the configuration. Add the `section` keyword and choose a section from the help menu as shown in the examples below.

`show configuration errors section ggsn-service`

or

`show configuration errors section aaa-config`

If the configuration contains no errors, an output similar to the following is displayed:

```
##############################################################################
Displaying Global AAA-configuration errors
##############################################################################
Total 0 error(s) in this section !
```

## Synchronizing File Systems

Whenever changes are made to a configuration or StarOS version boot order in a system with redundant management cards, the file systems must be synchronized across the management cards. This assures that the changes are identically maintained across the management cards.

Enter the following Exec mode command to synchronize the local file systems:

```
[local]host_name# filesystem synchronize all
```

The `filesystem` command supports multiple keywords that allow you to check for and repair file system corruption, as well as synchronize a file system with a specific storage device. For additional information, see the *Exec Mode Commands* chapter in the *Command Line Interface Reference*.

## Saving the Configuration

These instructions assume that you are at the root prompt for the Exec mode:

```
[local]host_name#
```

To save your current configuration, enter the following command:

```
save configuration url obsolete-encryption showsecrets verbose [-redundant] [-noconfirm] url
```

`url` specifies the location in which to store the configuration file. It may refer to a local or a remote file.

**Important**

The `obsolete-encryption` and `showsecrets` keywords have been removed from the `save configuration` command in StarOS 19.2 and higher. If you run a script or configuration that contains the removed keyword, a warning message is generated.

For complete information about the above command, see the *Exec Mode Commands* chapter of the *Command Line Interface Reference*.

**Important**

Do not use the "/" (forward slash), ":" (colon) or "@" (at sign) characters when entering a string for the following URL fields: directory, filename, username, password, host or port#.
The **-redundant** keyword is only applicable when saving a configuration file to a local device (usb1 or usb2) that is installed on both MIO/UMIO cards. This command does **not** synchronize the local file system. If you have added, modified, or deleted other files or directories to or from a local device for the active MIO/UMIO, you must synchronize the local file system on both MIO/UMIOs. See [Synchronizing File Systems](#) on page 79.

To save a configuration file called `system.cfg` to a directory that was previously created called `cfgfiles` to the flash memory on the active MIO/UMIO, enter the following command:

```
save configuration /flash/cfgfiles/system.cfg
```

```
save configuration sftp://administrator:secure@192.168.34.156/host_name_configs/ simple_ip.cfg
```
System Interfaces and Ports

This chapter describes how to create a context and configure system interfaces and ports within the context. Before beginning these procedures, refer to your product-specific administration guide for configuration information for your product.

Important

Make sure at least one Data Processing Card (DPC/UDPC or DPC2/UDPC2) is active before you configure system interfaces and ports. Refer to System Settings in this guide for information and instructions on activating DPCs.

- Contexts, page 81
- Ethernet Interfaces and Ports, page 82

Contexts

Even though multiple contexts can be configured to perform specific functions, they are all created using the same procedure.

Creating Contexts

Important

Commands used in the configuration examples in this section represent the most common or likely commands and/or keyword options. In many cases, other commands and/or keyword options are available. Refer to the Command Line Interface Reference for complete information regarding all commands.

To create a context, apply the following example configuration:

```
configure context name
end
```

Repeat to configure additional contexts.
Viewing and Verifying Contexts

**Step 1** Verify that your contexts were successfully created by entering the following command:

```
[local] host_name# show context all
```

The output is a two-column table similar to the example below. This example shows that two contexts were created: one named `source` and one named `destination`.

<table>
<thead>
<tr>
<th>Context Name</th>
<th>ContextID</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>local</td>
<td>1</td>
<td>Active</td>
</tr>
<tr>
<td>source</td>
<td>2</td>
<td>Active</td>
</tr>
<tr>
<td>destination</td>
<td>3</td>
<td>Active</td>
</tr>
</tbody>
</table>

The left column lists the contexts that are currently configured. The center column lists the corresponding context ID for each of the configured contexts. The third column lists the current state of the context.

**Step 2** Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

**Step 3** Now that the context has been created, interfaces and specific functionality can be configured within the context. Proceed to other sections for instructions on configuring specific services and options.

---

**Ethernet Interfaces and Ports**

Regardless of the type of application interface, the procedure to create and configure it consists of the following:

**Step 1** Create an interface and assign an IP address and subnet mask to it by applying the example configuration in *Creating an Interface*, on page 83.

**Step 2** Assign a physical port for use by the interface and bind the port to the interface by applying the example configuration in *Configuring a Port and Binding It to an Interface*, on page 83.

**Step 3** Optionally configure a static route for the interface by applying the example configuration in *Configuring a Static Route for an Interface*, on page 83.

**Step 4** Repeat the above steps for each interface to be configured.

This section provides the minimum instructions for configuring interfaces and ports to allow the system to communicate on the network. Commands that configure additional interface or port properties are described in the *Ethernet Port Configuration Mode Commands* and *Ethernet Interface Configuration Mode Commands* chapters of the *Command Line Interface Reference*.

To ensure that system line card and port-level redundancy mechanisms function properly, the Spanning Tree protocol must be disabled on devices connected directly to any system port. Failure to turn off the Spanning Tree protocol may result in failures in the redundancy mechanisms or service outage.
Creating an Interface

Use the following example to create a new interface in a context:

```
configure
    context name
    interface name
        { ip | ipv6 } address address subnetmask [ secondary ]
    end
end
```

Notes:

- *Optional:* Add the `loopback` keyword option to the `interface name` command, to set the interface type as "loopback" which is always UP and not bound to any physical port.

- *Optional:* Add the `secondary` keyword to the `{ ip | ipv6 } address` command, to assign multiple IP addresses to the interface. IP addresses can be entered using IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.

- *Optional:* In the interface config mode, add the `port-switch-on-L3-fail address` command, to configure the interface for switchover to the port on the redundant line card if connectivity to a specified IP address is lost. This IP address can be entered using IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.

Configuring a Port and Binding It to an Interface

Use the following example configuration to configure and assign a port to an interface:

```
configure
    port ethernet slot#/port#
    description description
    no shutdown
    bind interface interface_name context_name
    end
```

Notes:

- For `port ethernet slot#`, use the actual chassis slot in which the MIO/UMIO card is installed; this could be 5 or 6.

- For `port ethernet port#`, use ports 10 to 19 (DC1) or 20 to 29 (DC2).

- *Optional:* In the Ethernet Port configuration mode, add the preferred `slot slot#` command if you want to specify a port preference.

- Binding associates the port and all of its settings to the named interface.

Configuring a Static Route for an Interface

Use the following example to configure a static route for an interface:

```
configure
    context name
```


### Viewing and Verifying Port Configuration

**Step 1** Verify that your interface configuration settings are correct by entering the following commands:

```
(host_name) context context_name
show ip interface
```

*context_name* represents the name of the context in which the interface was created. The output from these commands should be similar to the following example.

In this example an interface named `mgmt1` was configured in the local context.

```
Intf Name: mgmt1
Intf Type: Broadcast
IP State: UP (Bound to 5/11 untagged, ifIndex 285278209)
IP Address: 192.168.100.3 Subnet Mask: 255.255.255.0
Bcast Address: 192.168.100.255 MTU: 1500
Resoln Type: ARP ARP timeout: 3600 secs
Number of Secondary Addresses: 0
Total interface count: 1
```

**Step 2** Verify that your port configuration settings are correct by entering the following command:

```
show configuration port slot#/port#
```

*slot#* is the chassis slot number of the MIO/UMIO on which the physical port resides. *slot#* can be 5 or 6.

This command produces an output similar to that displayed in the following example that shows the configuration for port 11 on the MIO/UMIO installed in chassis slot 5.

In this example, the port is bound to an interface called `rp1` configured in a context called `source`.

```
config
port ethernet 5/11
    description MIO5/11_RP1
    no shutdown
    bind interface rp1 source
#end
```

**Step 3** Verify that your static route(s) was configured properly by entering the following command:

```
show ip static-route
```
This command produces an output similar to that displayed in the following example that shows a static route to a gateway with an IP address of 192.168.250.1.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Nexthop</th>
<th>Protocol</th>
<th>Prec</th>
<th>Cost</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0/0</td>
<td>192.168.250.1</td>
<td>Static</td>
<td>0</td>
<td>0</td>
<td>MIO1</td>
</tr>
<tr>
<td>0.0.0.0/0</td>
<td>192.168.250.1</td>
<td>Static</td>
<td>0</td>
<td>0</td>
<td>rpl source</td>
</tr>
</tbody>
</table>

**Step 4**  
Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.
System Security

This chapter describes the security features supported on the ASR 5500 platform.

This chapter explores the following topics:

- Per-Chassis Key Identifier, page 87
- Encrypted SNMP Community Strings, page 90
- Lawful Intercept Restrictions, page 90
- Adding, Modifying and Removing Users, page 91
- Test-Commands, page 92

Per-Chassis Key Identifier

A user can set a unique chassis key which will work only for a chassis or for any set of chassis that will share the same configuration information.

The chassis key consists of 1 to 16 alphanumeric ASCII characters. The chassis key plain-text value is never displayed to the user; it is entered interactively and not echoed to the user.

On the ASR5500 the encrypted chassis key is stored in the midplane EEPROM and shared by both MIO/UMIOs.

If the chassis key identifier stored in the header comment line of the configuration file does not match the chassis key, an error message is displayed to the user. The user can change the chassis key value simply by entering the chassis key again. The previous chassis key is replaced by a new chassis key. The user is not required to enter a chassis key.

If the user does not configure a chassis key, the system generates a unique value for that chassis.

---

Important Changing a chassis key may invalidate previously generated configurations. This is because any secret portions of the earlier generated configuration will have used a different encryption key. For this reason the configuration needs to be recreated and restored.
To make password configuration easier for administrators, the chassis key should be set during the initial chassis set-up.

The configuration file contains a one-way encrypted value of the chassis key (the chassis key identifier) and the version number in a comment header line. These two pieces of data determine if the encrypted passwords stored within the configuration will be properly decrypted.

While a configuration file is being loaded, the chassis key used to generate the configuration is compared with the stored chassis key. If they do not match the configuration is not loaded.

The user can remove the chassis key identifier value and the version number header from the configuration file. Also, the user may elect to create a configuration file manually. In both of these cases, the system will assume that the same chassis key will be used to encrypt the encrypted passwords. If this is not the case, the passwords will not be decrypted due to resulting non-printable characters or memory size checks. This situation is only recoverable by setting the chassis key back to the previous value, editing the configuration to have the encrypted values which match the current chassis key, or by moving the configuration header line lower in the configuration file.

Beginning with Release 15.0, the chassis ID will be generated from an input chassis key using the SHA2-256 algorithm followed by base36 encoding. The resulting 44-character chassis ID will be stored in the same chassisid file in flash.

Release 14 and Release 15 chassis IDs will be in different formats. Release 15 will recognize a Release 14 chassis ID and consider it as valid. Upgrading from 14.x to 15.0 will not require changing the chassis ID or configuration file.

However, if the chassis-key is reset in Release 15 through the setup wizard or chassis-key CLI command, a new chassis ID will be generated in Release 15 format (44 instead of 16 characters). Release 14 builds will not recognize the 44-character chassis ID. If the chassis is subsequently downgraded to Release 14, a new 16-character chassis ID will be generated. To accommodate the old key format, you must save the configuration file in pre-v12.2 format before the downgrade. If you attempt to load a v15 configuration file on the downgraded chassis, StarOS will not be able to decrypt the password/secrets stored in the configuration file.

MIO Synchronization

On boot up both MIO/UMIOs automatically read the chassis key configured on the ASR 5500 midplane.

Protection of Passwords

Users with privilege levels of Inspector and Operator cannot display decrypted passwords in the configuration file via the command line interface (CLI).

Secure Password Encryption

By default the system encrypts passwords using an MD5-based cipher (option A). These passwords also have a random 64-bit (8-byte) salt added to the password. The chassis key is used as the encryption key.

Setting a chassis key supports an encryption method where the decryption requires the knowledge of a "shared secret". Only a chassis with knowledge of this shared secret can access the passwords. To decipher passwords,
a hacker who knew the chassis key would still need to identify the location of the 64-bit random salt value within the encryption.

Passwords encrypted with MD-5 will have "+A" prefixes in the configuration file to identify the methodology used for encrypting.

For release 15.0 and higher, another type of encryption algorithm can be specified. The Global Configuration mode cli-encrypt-algorithm command allows an operator to configure the password/secret encryption algorithm. The default encryption/password algorithm is MD-5 as described above (option A). A second password encryption algorithm (option B) uses AES-CBC-128 for encryption and HMAC-SHA1 for authentication. The encryption key protects the confidentiality of passwords, while the authentication key protects their integrity. Passwords encrypted with this key will have "+B" prefixes in the configuration file.

For release 19.2 and higher, a third type of encryption algorithm can be specified (option C). This algorithm specifies the use of the HMAC-SHA512 cipher algorithm for encryption and authentication. Passwords encrypted with this key will have "+C" prefixes in the configuration file.

Also for release 19.2 and higher, the encryption key is hashed from the chassis ID and a 16-byte Initialization Vector (IV) obtained from an internal random number generator. No two passwords are encrypted using the same encryption key/IV pair. The Security Administrator must set a chassis key in order to generate the chassis ID and resulting encryption key. A default chassis key based on a local MAC address is no longer supported.

The syntax for the cli-encrypt-algorithm command is:

```
config
  cli-encrypt-algorithm {A | B | C}
```

---

### Support for Non-Current Encryptions and Decryptions

The system supports previously formatted encrypted passwords. The syntax of the encrypted passwords indicates which methodology was used for encryption. If the system does not see a prefix before the encrypted password, the earlier encryption method using a fixed key will be used. If the encrypted password includes the "+A" prefix, the decryption method uses the chassis key and random salt.

If the user saves a new configuration, the generated file will always contain passwords encrypted by the most recent method. The user cannot generate the earlier DES-based encryption values. However, all future StarOS releases will continue to support plain-text password entry for all two-way encryptable passwords.

The recommended process for changing the chassis key without causing a "lock-out" state is as follows:

- Load the configuration file of the last good configuration using the previous chassis key.
- Change the chassis key to the new desired value.
- Save the configuration with this new chassis key.

Refer to Configuring a Chassis Key in System Settings for additional information.

---

### Support for ICSR Configurations

Inter-Chassis Session Recovery (ICSR) is a redundancy configuration that employs two identically configured ASR 5500 chassis as a redundant pair.

ICSR chassis share the same chassis key. If the ISCR detects that the two chassis have incompatible chassis keys, an error message is logged but the ICSR system will continue to run. Without the matching chassis key,
the standby ICSR chassis can recover services if the active chassis goes out of service; the standby chassis will still have access to the passwords in their decrypted form.

ICSR chassis use Service Redundancy Protocol (SRP) to periodically check to see if the redundancy configuration matches with either decrypted passwords or DES-based two-way encryption strings. Since the configuration is generated internally to the software, users are not able to access the configuration used to check ICSR compatibility.

**Encrypted SNMP Community Strings**

Simple Network Management Protocol (SNMP) uses community strings as passwords for network elements. Although these community strings are sent in clear-text in the SNMP PDUs, the values can be encrypted in the configuration file.

The `snmp community encrypted name` command enables the encryption of SNMP community strings. For additional information, see the Global Configuration Mode Commands chapter in the Command Line Interface Reference.

**Lawful Intercept Restrictions**

This section describes some of the security features associated with the provisioning of Lawful Intercept (LI). For additional information, refer to the Lawful Intercept Configuration Guide.

**LI Server Addresses**

An external authenticating agent (such as RADIUS or Diameter) sends a list of LI server addresses as part of access-accept. For any intercept that was already installed or will be installed for that subscriber, a security check is performed to match the LI server address with any of the LI-addresses that were received from the authenticating agent. Only those addresses that pass this criteria will get the intercepted information for that subscriber.

While configuring a campon trigger, the user will not be required to enter the destination LI server addresses. When a matching call for that campon trigger is detected, a security check is done with the list received from the authentication agent. The LI-related information is only forwarded if a matching address is found.

When an active-only intercept is configured, if a matching call is found, a security check is made for the LI address received from the authentication agent and the intercept configuration will be rejected.

If no information related to LI server addresses is received for that subscriber, LI server addresses will not be restricted.

---

**Important**

A maximum of five LI server addresses are supported via an authenticating agent.

**Important**

The ability to restrict destination addresses for LI content and event delivery using RADIUS attributes is supported only for PDSN and HA gateways.
Modifying Intercepts

One LI administrator can access and/or modify the intercepts created by another LI administrator. Whenever an intercept is added, removed or modified, an event log is displayed across LI administrators about the change. An SNMP trap is also generated.

Adding, Modifying and Removing Users

It is considered uncommon for a user to be added or removed from the ASR 5500. Likewise, it is considered uncommon for a user's privileges to modified. However, if the system is compromised, it is common for attackers to add or remove a privileged user, raise their privileges or lower the privileges of others.

As a general rule, lower privileged users should not be allowed to increase their privileges or gain access to sensitive data, such as passwords, which were entered by higher privileged users.

Important

The ASR 5500 can only detect changes in users and user attributes, such as privilege level, when these users are configured through the ASR 5500.

Notification of Users Being Added or Deleted

Users with low level authorization should not be able to create users with high level authorization. However, if a malicious actor were to be able to create a high level authorized user, they could then delete the other high level authorized users, thereby locking them out of the system.

The following SNMP traps notify an administrator when users are added or removed:

- `starLocalUserAdded` – indicates that a new local user account has been added to the system.
- `starLocalUserRemoved` – indicates that a local user account has been removed from the system.

Notification of Changes in Privilege Levels

Whenever a user's privilege level is increased or decreased, an SNMP notification will be sent out. A malicious actor may gain access to more privileged commands by somehow promoting" their privileges. Once this is done, they could then "demote" the privileges of all the other users, thereby locking the proper administrators out of the system.

The `starLocalUserPrivilegeChanged` trap indicates that a local user's privilege level has been changed.

User Access to Operating System Shell

The `starOsShellAccessed` trap indicates that a user has accessed the operating system shell.
Test-Commands

Users with Security Administrator or Administrator privilege can enable the display of previously hidden test-commands. The CLI test-commands mode displays new command keywords for existing commands, as well as new commands.

Caution

CLI test-commands are intended for diagnostic use only. Access to these commands is not required during normal system operation. These commands are intended for use by Cisco TAC personnel only. Some of these commands can slow system performance, drop subscribers, and/or render the system inoperable.

Enabling cli test-commands Mode

To enable access to test-commands, a Security Administrator must log into the Global Configuration mode and enter cli hidden.

This command sequence is shown below.

```
[local] host_name(config)# cli hidden
```

By default cli hidden is disabled.

Important

Low-level diagnostic and test commands/keywords will now be visible to a user with Administrator or higher privilege. There is no visual indication on the CLI that the test-commands mode has been enabled.

Enabling Password for Access to CLI-test commands

A Security Administrator can set a plain-text or encrypted password for access to CLI test commands. The password value is stored in /flash along with the boot configuration information. The show configuration and save configuration commands will never output this value in plain text.

The Global Configuration mode command `tech-support test-commands [encrypted] password new_password [ old-password old_password ]` sets an encrypted or plain-text password for access to CLI test-commands.

This command sequence is shown below.

```
[local] host_name(config)# tech-support test-commands password new_password [ old-password old_password ]
```

If the new password replaces an existing password, you must enter the old password for the change to be accepted.

If the old password is not entered or does not match the existing configured value, the following error message appears: "tech-support password is already configured". A prompt then appears to accept entry of the old password: "Enter old tech-support password:".
Entering **old-password old_password** allows you to replace the existing password without being prompted to enter the old password. If you incorrectly enter the old password or do not enter the old password, an error message appears: "Failure: Must enter matching tech-support password to replace existing password".

The Quick Setup Wizard (Exec mode **setup** command) also prompts for entry of a tech-support test-commands password. If you have forgotten the old tech-support password, you can run **setup** directly from the Console port to enter a new tech-support password.

When a test-commands password is configured, the Global Configuration mode command **cli test-commands** 
  [encrypted] **password** requires the entry of the password keyword. If the **encrypted** keyword is specified, the **password** argument is interpreted as an encrypted string containing the password value. If the **encrypted** keyword is not specified, the **password** argument is interpreted as the actual plain text value.

<table>
<thead>
<tr>
<th>Important</th>
</tr>
</thead>
<tbody>
<tr>
<td>If <strong>tech-support test-commands password</strong> is never configured, StarOS will create a new password. If the <strong>password</strong> keyword is not entered for <strong>cli test-commands</strong>, the user is prompted (no-echo) to enter the password. Also, <strong>cli hidden</strong> must be enabled by an administrator to access the CLI test-commands.</td>
</tr>
</tbody>
</table>

### Exec Mode cli test-commands

Exec mode commands are available to a privileged user who enters the command **cli test-commands** from Exec mode.

```
[local]host_name# cli test-commands [encrypted] password password
```

**Warning**: Test commands enables internal testing and debugging commands

**Important**: An SNMP trap (starTestModeEntered) is generated whenever a user enters CLI test-commands mode.

### Configuration Mode cli test-commands

Configuration commands which provided access to low-level software parameters are accessible only after a privileged user enters the command **cli test-commands** from Global Configuration mode.

```
[local]host_name# config
[local]host_name(config)# cli test-commands [encrypted] password password
```

**Warning**: Test commands enables internal testing and debugging commands

**Important**: An SNMP trap (starTestModeEntered) is generated whenever a user enters CLI test-commands mode.
Software Management Operations

This chapter provides information about software management operations on the system.

- Understanding the Local File System, page 95
- Maintaining the Local File System, page 96
- Configuring the Boot Stack, page 101
- Upgrading the Operating System Software, page 105
- Performing Dynamic Software Updates, page 111
- Managing License Keys, page 112
- Managing Local-User Administrative Accounts, page 115

Understanding the Local File System

The Management Input/Output (MIO/UMIO) card provides control and management for the system.

The local file system is made up of files that are stored on one or more of the following:

- /flash - Flash memory located on the circuit board of the MIO/UMIO, is the default storage media for
  the operating system software image, CLI configuration, and crash log files used by the system.

- /usb1 - This device is available when a USB memory stick is inserted on the front panel of the active
  MIO/UMIO.

- /hd-raid - This is the solid state hard disk array supported by the Fabric and Storage Cards (FSCs) and
  accessed via the active MIO/UMIO.

File Types Used by the Local File System

The following file types can be located in the local file system:

- Operating System Software Image File: This binary file type is identified by its .bin extension. The
  file is the operating system that is loaded by the system upon startup or reloading. This is an executable,
  read-only file that cannot be modified by end users.
• **CLI Configuration File**: This file type is identified by its .cfg extension. These are text files that contain CLI commands that work in conjunction with the operating system software image. These files determine services to be provided, hardware and software configurations, and other functions performed by the system. The files are typically created by the end user. You can modify the files both on and off-line and use descriptive long filenames.

• **System File**: Only one file identified by a .sys extension is used by the system. The boot.sys file contains system-specific information, which describes how the system locates, and in what priority it loads, file groups (paired .bin and .cfg files) from its boot stack.

• **Abridged Crash Log**: The abridged crash log, identified by its crashlog filename, contains summary information about software or hardware failures that occur on the system. This file is located in the /flash/crsh2/ directory on the device. You can view the contents of this file through the CLI, but you cannot modify the file.

---

### Understanding the boot.sys File

The system uses the boot.sys file to store the prioritized boot stack parameters and file groups the system uses during startup. Modify this file only through system CLI commands and not through external means. Boot parameters contain information the system needs to locate the operating system image file, including:

- **bootmode**: This setting is typically configured to normal, and identifies how the system starts.

- **network interface configuration**: Use these optional boot method settings when you configure the system to obtain its operating system image from an external network server that is using one of the management LAN interfaces on the MIO/UMIO card.

- **boot stack information**: The boot stack is made up of prioritized file group entries that designate the operating system image file and the CLI configuration file to load.

When a system is unpacked and started for the first time, the boot.sys file is configured to use the normal boot mode and load the operating system software image from the /flash directory.

There is no CLI configuration file contained on the local file system. This causes the system to automatically start its CLI-based Quick Setup Wizard upon the first successful boot. Refer to *Getting Started* for more information on using the Quick Setup Wizard.

---

### Maintaining the Local File System

Use CLI commands to manage and maintain the devices that make up the local file system. Execute all the commands described in this section in the Exec Mode. Unless otherwise specified, you must have security administrator or administrator privileges to execute these commands.

---

### File System Management Commands

Use the commands in this section to manage and organize the local file system.
Important

For complete information on the commands listed below, see the Exec Mode Commands chapter of the Command Line Interface Reference.

Synchronizing the File System

Commands are supported for mirroring the local file systems from the active MIO/UMIO to the standby MIO/UMIO in systems containing two cards. Use these commands to synchronize any or all of the local devices.

Important

Crash log files are not synchronized when these commands are executed.

The following Exec mode command synchronizes the file systems between two MIO/UMIOs:

```
[local] host_name# filesystem synchronize [/flash | /usb1 | all ] [ checkonly ] [ from card_num ] [ to card_num | -noconfirm ]
```

Only filesystems on matching local devices are synchronized. For example, if the active MIO/UMIO contains two local devices (/flash and /usb1) and the standby MIO/UMIO contains only one local device (/flash), then synchronization only occurs on the matching local device (/flash).

The following command synchronizes the file systems on two MIO/UMIO flash devices.

```
[local] host_name# filesystem synchronize /flash
```

Creating Directories

Use the `mkdir` command to create a new directory on the specific local device. This directory can then be incorporated as part of the path name for any file located in the local file system.

```
[local] host_name# mkdir { /flash | /usb1 | /hd-raid } /dir_name
```

Use the following command to create a directory named `configs`:

```
[local] host_name# mkdir /flash/configs
```

Renaming Files and Directories

Use the `rename` command to change the name of a file from its original name to a different name. Remember to use the same file extension, if applicable, to ensure that the file type remains unchanged.

```
[local] host_name# rename { /flash | /usb1 | /hd-raid } /src_filename { /flash | /usb1 | /hd-raid } /dst_filename | -noconfirm ]
```

Use the following command to rename a file named `iot_test.cfg` to `iot_accept.cfg` on the /flash local device.

```
[local] host_name# rename /flash/iot_test.cfg /flash/iot_accept.cfg -noconfirm
```

Important

Use the `rename` command only within the same local device. You cannot rename a file and place it onto another local device at the same time. To move a renamed file, you must use the `copy` command.
Copying Files on the ASR 5500 Chassis

These instructions assume that you are at the root prompt for the Exec mode. To save your current configuration, enter the following command:

```bash
[local]<host_name> copy from_url to_url [-noconfirm]
```

To copy a configuration file called `system.cfg` from a directory that was called `cfgfiles` to a directory named `configs_old` on the flash memory in the MIO/UMIO, enter the following command:

```bash
[local]<host_name> copy /flash/cfgfiles/system.cfg /flash/configs_old/system_2011.cfg
```

To copy a configuration file called `init_config.cfg` to the root directory of a TFTP server with a hostname of `config_server`, enter the following command:

```bash
[local]<host_name> copy /flash/cfgfiles/init_config.cfg tftp://config_server/init_config.cfg
```

Deleting Files

The `delete` command removes a designated file from its specified location on the local file system.

**Important**

This command does not support wildcard entries; each filename must be specified in its entirety.

**Caution**

Do not delete the `boot.sys` file. If deleted, the system will not reboot on command and will be rendered inoperable.

```bash
[local]<host_name> delete{/flash | /usb1 | /hd-raid}/filename [-noconfirm]
```

The following command deletes a file named `test.cfg` from the `/flash` directory.

```bash
[local]<host_name> delete /flash/test.cfg
```

Removing Directories

The `rmdir` command deletes a current directory on the specific local device. This directory can then be incorporated as part of the path name for any file located in the local file system.

**Important**

The directory you want to remove (delete) must be empty before executing the `rmdir` command. If the directory is not empty, the CLI displays a "Directory not empty" message and will not execute.

```bash
[local]<host_name> rmdir url /dir_name
```

The following command deletes an empty directory named `configs` in the `/flash` directory.

```bash
[local]<host_name> rmdir /flash/configs
```

Formatting Local Devices

The `format` command performs a low-level format of a local device. This operation formats the device to use the FAT16 formatting method, which is required for proper read/write functionality with the operating system.
Local devices that have been formatted using other methods such as NTFS or FAT32 may be used to store various operating system, CLI configuration, and crash log files. However, when placing a new local device into the MIO/UMIO for regular use, you should format the device via the system prior to use. This ensures that the proper file allocation table format is used, preventing any possible discrepancies between other formats used with other operating systems.

### Important

The **filesystem format** command removes all files and information stored on the device.

To format a local device for use by the local file system, enter the following command:

```
[local] host_name# filesystem format { /flash | /usb1 | /hd-raid }
```

### Applying Pre-existing CLI Configuration Files

A pre-existing CLI configuration file is any .cfg file created to provide utility functions (such as clearing all statistics during testing) or created off-line using a text editor. There may be pre-existing configuration files stored on the local file system that can be applied to a running system at any time.

### Caution

If a configuration file is applied to a system currently running another CLI configuration, any like contexts, services, logical interfaces, physical ports, IP address pools, or other configured items will be overwritten if the same command exists in the configuration file being applied. Take caution to ensure that you are knowledgeable of the contents of the file being applied and understand what the service ramifications are if a currently running command is overwritten. Also note that changes will not be saved automatically.

A CLI configuration file, or script containing CLI commands, can be applied to a running system by entering the following command at the Exec mode prompt:

```
[local] host_name# configure url [ verbose ]
```

`url` specifies the location of the CLI configuration file to be applied. It may refer to a local or a remote file.

The following command applies a pre-existing CLI configuration file named `clearcmds.cfg` in the `/flash` directory.

```
[local] host_name# configure /flash/clearcmds.cfg
```

### Viewing Files on the Local File System

This section describes how to view a variety of files.

#### Viewing the Contents of a Local Device

The contents, usage information, and file system directory structure of any local device can be viewed by entering the following command at the Exec mode prompt:

```
directory { /flash | /usb1 | /hd-raid }
```
**Viewing CLI Configuration and boot.sys Files**

The contents of CLI configuration and boot.sys files, contained on the local file system, can be viewed off-line (without loading them into the OS) by entering the following command at the Exec mode prompt:

```
[local] host_name# show file url {/flash | /usb1 | /hd-raid} filename
```

Where: *url* is the path name for the location of the file and *filename* is the name of the file, including any extension.

---

**Important**

Operator and inspector-level users can execute the `show file` command but cannot execute the `directory` command.

---

**Validating an Operating System File**

The operating system software image file, identified by its .bin extension, is a non-readable, non-editable file that executes on the system, creating its runtime operating system (OS).

It is important to verify a new operating system image file before attempting to load it. To accomplish this, a proprietary checksum algorithm is used to create checksum values for each portion of the application stored within the .bin file during program compilation.

This information can be used to validate the actual file against the checksum values stored within the file during its compilation. If any portion of the image file has become corrupted (for example, the file was truncated or was transferred using ASCII mode instead of binary mode), then this information is reported and the file is deemed unusable.

To validate an operating system software image file, enter the following command at the Exec mode prompt:

```
[local] host_name# show version {/flash | /usb1 | /hd-raid}[/directory]filename [all]
```

The output of this command displays the following information:

- Version number
- Description
- Date
- Boot Image
- Size
- Flags
- Platform – ASR5500

If an invalid file is found, the system displays a failure message similar to these:

```
Failure: Image /flash/image_version.bin CRC check failed!
Failure: /flash/image_version.bin, has a bad magic number
```
Configuring the Boot Stack

The boot stack consists of a prioritized listing of operating system software image-to-CLI configuration file associations. These associations determine the software image and configuration file that gets loaded during system startup or upon a reload/reboot. Though multiple associations can be configured, the system uses the association with the highest priority. In the event that there is an error processing this association (for example, one of the files cannot be located), the system attempts to use the association with the next highest priority. Priorities range from 1 to 100, with 1 being the highest priority. The maximum number of boot stack entries that may be configured in the boot.sys file is 10.

Boot stack information is contained in the boot.sys file, described in Understanding the boot.sys File, on page 96. In addition to boot stack entries, the boot.sys file contains any configuration commands required to define the system boot method as explained in the section that follows.

System Boot Methods

The local-boot method uses software image and configuration files stored locally on the system. Upon system startup or reboot, the system looks on one of its local devices or /hd-raid located on the active MIO/UMIO for the specific software image and accompanying configuration text file. When using the local-booting method, you only need to configure boot stack parameters.

The system can also be configured to obtain its software image from a specific external network server while it is paired with a configuration text file that resides on the system. When using network booting, you need to configure the following:

- Boot stack parameters, which define the files to use and in what priority to use them
- Boot interface and network parameters defining the MIO/UMIO remote management LAN interface and the methods to use to reach the external network server
- Network booting delay time and optional name server parameters defining the delay period (in seconds) to allow for network communications to be established, and the IP address of any Domain Name Service (DNS) name server that may be used

Detailed information on how to configure the system to use the network booting method appears in Network Booting Configuration Requirements, on page 103

Viewing the Current Boot Stack

To view the boot stack entries contained in the boot.sys file run the Exec mode show boot command.

Important: Operator and inspector-level users can execute the show boot command.

The examples below show the command output for a local booting configuration. Notice that in these examples both the image file (operating system software) and configuration file (CLI commands) are located on the /flash device.
The StarOS image filename scheme changed with release 16.1. Pre-16.1, format = "production.image.bin". For 16.1 onwards, format = "asr5500-image_number.bin". This change is reflected in the examples provided below.

**Example 1 – StarOS releases prior to 16.1:**

```
boot system priority 18 \
    image /flash/15-0-builds/production.45666.bin \
    config /flash/general_config.cfg

boot system priority 19 \
    image /flash/15-0-builds/production.45717.bin \
    config /flash/general_config_3819.cfg

boot system priority 20 \
    image /flash/15-0-builds/production.45069.bin \
    config /flash/general_config_3665.cfg
```

**Example 2 – StarOS release 16.1 onwards:**

```
boot system priority 18 \
    image /flash/16-1-builds/asr5500-16.1.3.bin \
    config /flash/general_config.cfg

boot system priority 19 \
    image /flash/16-1-builds/asr5500-16.1.1.bin \
    config /flash/general_config_3819.cfg

boot system priority 20 \
    image /flash/16-1-builds/asr5500-16.1.0.bin \
    config /flash/general_config_3665.cfg
```

The example below shows the output for a combination network booting and local booting configuration. Notice in this example that the first two boot stack entries (Priorities 18 and 19) load the image file (operating system software) from an external network server using the Trivial File Transfer Protocol (TFTP), while all configuration files are located on the /flash device.

Also notice the boot network interface and boot network configuration commands located at the top of the boot stack. These commands define what MIO/UMIO remote management LAN interface(s) to use and information about communicating with the external network server that hosts the operating system software image file.

```
boot networkconfig static ip address mio1 192.168.1.150 netmask 255.255.255.0
boot delay 15

boot system priority 18 image tftp://192.168.1.161/tftpboot/image_version.bin \config /flash/general_config.cfg

boot system priority 19 image tftp://192.168.1.161/tftpboot/image_version.bin \config /flash/general_config.cfg

boot system priority 20 image /flash/image_version.bin \config /flash/general_config.cfg
```

To identify the boot image priority that was loaded at the initial boot time enter:

```
show boot initial-config
```

The example below displays the output:

```
[local] host_names show boot initial-config
Initial (boot time) configuration:
    image tftp://192.168.1.161/tftpboot/image_version.bin \config /flash/config_name.cfg
    priority 1
```
Adding a New Boot Stack Entry

**Important**

Before performing this procedure, verify that there are less than 10 entries in the boot.sys file and that a higher priority entry is available (i.e. that minimally there is no priority 1 entry in the boot stack). Refer to *Viewing the Current Boot Stack* for more information.

If priority 1 is in use, then you must renumber the existing entry(ies) to ensure that at least that priority is available. The maximum number of boot stack entries that can be contained in the boot.sys file is 10. If there are already 10 entries in the boot stack, you must delete at least one of these entries (typically, the lowest priority) and, if necessary, renumber some or all of the other entries before proceeding. Refer to *Deleting a Boot Stack Entry*, on page 103 for more information.

This procedure details how to add new boot stack entries to the boot.sys file. Make sure you are at the Exec mode prompt and enter the following commands:

```plaintext
configure
  boot system priority number image image_url config cfg_url
```

The following command creates a new boot stack entry, using a boot priority of 3.

```plaintext
boot system priority 3 image /flash/image_filename.bin config /flash/config_name.cfg
```

**Important**

Boot stack changes saved to the boot.sys file are not executed until the system is rebooted.

Synchronize the local file systems on the MIO/UMIOs with the following command:

```plaintext
filesystem synchronize all
```

Deleting a Boot Stack Entry

This procedure details how to remove an individual boot stack entry from the boot.sys file. Make sure you are at the Exec mode prompt and enter the following commands:

```plaintext
configure
  no boot system priority number
```

Where *number* specifies the boot priority used for the boot stack entry. This command removes that specific entry from the boot stack, causing the boot.sys file to be overwritten.

Network Booting Configuration Requirements

Configuring the Boot Interface

Boot interface parameters define the MIO/UMIO management LAN interface that the system will use to communicate with the management network when using the network booting method.
This procedure details how to configure the boot interface for reliable communications with your network server. Make sure you are at the Exec mode prompt.

**Step 1** Enter the Global Configuration mode by entering the following command:

```
[local]host_name(config) configure
```

The following prompt appears:

```
[local]host_name(config)#
```

**Step 2** Enter the following command:

```
[local]host_name(config)# boot interface { local-eth1 | local-eth2 } medium { auto | speed { 10 | 100 | 1000 } duplex { full | half } } media { rj45 | sfp }
```

For complete information about the above command, see the *Global Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

Use port 1 for network booting.

If the speed is manually configured, you must also configure the duplex mode. In addition, you must ensure that the network server configuration supports the speed and duplex configuration.

Network speed for MIO/UMIO is fixed at 1000.

Ethernet networking rules dictate that if a device's interface is configured for auto-negotiation is communicating with a device that is manually configured to support full duplex, the first device will negotiate to the manually configured speed of the second device, but will only communicate in half duplex mode.

The media for MIO/UMIO port 1 is fixed at rj45.

**Step 3** Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

---

**Configuring the Boot Network**

Boot network parameters define the protocols and IP address information for MIO/UMIO interfaces used to reach the external network server that hosts the operating system software image file. To configure boot network parameters, make sure you are at the Exec mode prompt.

**Step 1** Enter the Global Configuration mode by entering the following command:

```
[local]host_name(config) configure
```

The following prompt appears:

```
[local]host_name(config)#
```

**Step 2** Enter the following command:

```
[local]host_name(config)# boot networkconfig { dhcp | { dhcp-static-fallback | static } ip address mio5 ip_address5 mio6 ip_address6 netmask subnet_mask gateway gw_ip_address }
```

For complete information about the above command, see the *Global Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

The following command configures the boot network to communicate using DHCP, with a static-fallback IP address for MIO/UMIO in slot 5 of 192.168.206.101 and a Class C netmask.

```
[local]host_name(config)# boot networkconfig dhcp-static-fallback ip address mio5 192.168.206.101 netmask 255.255.255.0
```
The next example uses static IP addresses for MIO/UMIO in slot 5, which can access the external network server through a gateway whose IP address is 135.212.10.2.

```
[local]host_name(config)# boot networkconfig static ip address mio5 192.168.206.101 netmask 255.255.255.0 gateway 135.212.10.2
```

**Step 3** Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

---

**Configuring Boot Network Delay Time**

An optional delay period, in seconds, can be configured for systems booting from a network. The purpose of this parameter is to allow time for external devices, such as switches, that use the Spanning Tree Protocol (STP) to determine the network route to a specified IP address.

To configure a boot network delay, enter the following command from the Global Configuration mode prompt.

```
[local]host_name(config)# boot delay time
```

Where *time* is an integer from 1 to 300 seconds before attempting to contact the external network server. If your network uses STP, a typical delay time of 30 seconds should suffice.

---

**Important** Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

---

**Configuring a Boot Nameserver**

To enter the hostname of the network server that hosts the operating system software image, first configure the IP address of the Domain Name Service (DNS) server, referred to as a name server, that can resolve the host name for the machine.

To configure a boot nameserver address, enter the following command from the Global Configuration mode prompt.

```
[local]host_name(config)# boot nameserver ip_address
```

Where *ip_address* is the IP address of the DNS server entered in IPv4 dotted-decimal notation.

---

**Important** Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

---

**Upgrading the Operating System Software**

The following information is required prior to performing a software upgrade:

- Current operating system version
- New operating system version
- Upgrade method
Identifying OS Release Version and Build Number

The operating system can be configured to provide services and perform pre-defined functions through commands issued from the CLI or through the Web Element Manager application.

The operating system software is delivered as a single binary file (.bin file extension) and is loaded as a single instance for the entire system.

- **For StarOS releases prior to 16.1**, the image filename is identified by its release type, build number and platform type. For example, `production.build_number.asr5500.bin`. For example, `production.54029.asr5500.bin`.

- **For StarOS release 16.1 onwards**, the image filename is identified by a suffix specifying its platform type and release number. For example, `asr5500-release_number.bin`. For example, `asr5500-16.1.0.bin`.

*For StarOS releases 20.0 and higher*, a starfile image must be signed with an REL key before being released. A deployable image will be signed with an REL key having a ".bin.SPA" extension, where "A" identifies the revision level of the signing key. For example, `asr5500-20.0.0.bin.SPA`. If a signing key becomes compromised, a new key is created and the revision level increments to "B".

*For StarOS releases 20.0 and higher* Trusted images have been introduced. The difference between a Trusted build and a Normal build is the absence of unsecure programs `ftpd`, `telnet` and `tcpdump`, as well as the addition of a `staros.conf` file for security options. Trusted images are identifiable by the presence of "_T" in the platform name. For example, `asr5500_T-20.0.0.bin.SPA`.

The software version information can be viewed from the CLI in the Exec mode by entering the `show version` command.

```
[local] host_name# show version
```

Verify Free Space on the /flash Device

Verify that there is enough free space on the /flash device to accommodate the new StarOS image file by entering the following Exec mode command:

```
[local] host_name# directory /flash
```

The following is an example of the type of directory information displayed:

```
-rwxrwxr-x 1 root root 7334 May 5 17:29 asr-config.cfg
-rwxrwxr-x 1 root root 399 Jun 7 18:32 system.cfg
-rwxrwxr-x 1 root root 10667 May 14 16:24 testconfig.cfg
-rwxrwxr-x 1 root root 10667 Jun 1 11:21 testconfig_4.cfg
-rwxrwxr-x 1 root root 5926 Apr 7 16:27 tworcontext.cfg
-rwxrwxr-x 1 root root 15534 Aug 4 13:31 test_vlan.cfg
-rwxrwxr-x 1 root root 2482 Nov 18 11:09 gateway2.cfg
-rwxrwxr-x 1 root root 159106048 Dec 31 2011 image_filename

1136352 /flash
```

Note the "Available" blocks in the last line of the display. After displaying the directory information, the CLI returns to root and the following prompt appears:

```
[local] host_name#
```
Download the Software Image from the Support Site

Access to the Cisco support site and download facility is username and password controlled. You must have an active customer account to access the site and download the StarOS image.

Download the software image to a network location or physical device (USB stick) from which it can be uploaded to the /flash device.

Contact your Cisco representative or Cisco TAC for additional information.

Transfer StarOS Image to /flash on the Chassis

Transfer the new operating system image file to the /flash device on the MIO/UMIO using one of the following methods:

• Copy the file from a network location or local device plugged in into the MIO/UMIO by entering the following command:

  [local]<host_name> copy from_url to_url [ -noconfirm ]

• Transfer the file to the /flash device using an FTP client with access to the system.

  (Important) Whenever transferring a operating system software image file using the file transfer protocol (FTP), the FTP client must be configured to transfer the file using binary mode. Failure to use binary transfer mode will make the transferred operating system image file unusable. In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

  • Transfer the file to the /flash device using an SFTP client with access to the system.

Verify that the image file was successfully transferred to the /flash device by running the following Exec mode command:

  [local]<host_name> directory /flash

The image filename should appear in the displayed output.

Run the show version /flash/image_filename command to verify the build information.

  [local]<host_name> show version /flash/image_filename.bin

Saving a Copy of the Current Configuration File

Prior to upgrading to a new software release, you should copy and rename the current configuration file to the /flash device and to an off-chassis location (external memory device or network URL). This renamed copy assures that you will have a fallback, loadable configuration file should a problem be encountered during the upgrade.
Downgrading from Release 15.0 to 14.0

Release 14 and Release 15 chassis IDs use different encryption formats. Release 15 will recognize a Release 14 chassis ID and consider it as valid. Upgrading from 14.x to 15.0 will not require changing the chassis ID or configuration file.

However, if the chassis key is reset in Release 15 through the setup wizard or `chassis-key` CLI command, a new chassis ID will be generated in Release 15 format (44 instead of 16 characters). Release 14 builds will not recognize the 44-character chassis ID. If the chassis is subsequently downgraded to Release 14, a new 16-character chassis ID will be generated. To accommodate the old key format, you must save the configuration file in pre-v12.2 format before the downgrade. If you attempt to load a v15 configuration file on the downgraded chassis, StarOS will not be able to decrypt the password/secrets stored in the configuration file.

Downgrading from Release 20.0

Prior to release 20.0, local-user passwords were hashed with the MD5 message digest-algorithm and saved in the database. In release 20.0, PBKDF2 (Password Based Key Derivation Function - Version 2) is now used to derive a key of given length, based on entered data, salt and number of iterations. Local-user account passwords are hashed using the PBKDF2 method with a randomly generated salt coupled with a large number of iterations to make password storage more secure.

Since hash functions are one-way, it is not possible to convert PBKDF2 hashed passwords to the MD5 format. The local-user database must be downgraded prior to reverting to StarOS releases prior to 20.0.

To downgrade the local-user database to use the MD5 hash algorithm, a Security Administrator must run the Exec mode `downgrade local-user database` command. StarOS prompts for confirmation and requests the Security Administrator to reenter a password. The entered password re-authenticates the user prior to executing the downgrade command. After verification, the password is hashed using the appropriate old/weak encryption algorithm and saved in the database to allow earlier versions of StarOS to authenticate the Security Administrator.

The downgrade process does not convert PBKDF2 hashed passwords to MD5 format. The downgrade process re-reads the database (from the /flash directory), reconstructs the database in the older format, and writes it back to the disk. Since the PBKDF2 hashed passwords cannot be converted to the MD5 hash algorithm, and earlier StarOS releases cannot parse the PBKDF2 encryption algorithm, StarOS suspends all those users encrypted via the PBKDF2 algorithm. Users encrypted via the MD5 algorithm (“Weak Hash” flag) can continue to login with their credentials. After the system comes up with the earlier StarOS release, suspended users can be identified in the output of the `show local-user [verbose]` command.

To reactivate suspended users a Security Administrator can:

- Set temporary passwords for suspended users, using the Exec mode `password change local-user username` command.
- Reset the suspend flag for users, using the Configuration mode `no suspend local-user username` command.

Off-line Software Upgrade

An off-line software upgrade can be performed for any system, upgrading from any version of operating system software to any version, regardless of version number. This process is considered off-line because...
while many of the steps can be performed while the system is currently supporting sessions, the last step of this process requires a reboot to actually apply the software upgrade.

This procedure assumes that you have a CLI session established and are placing the new operating system image file onto the local file system. To begin, make sure you are at the Exec mode prompt:

```
[local] host_name#
```

**Configure a Newcall Policy**

Configure a newcall policy from the Exec mode to meet your service requirements. When enabled the policy redirects or rejects new calls in anticipation of the chassis reload that completes the upgrade process. This reduces the amount of service disruption to subscribers caused by the system reload that completes the upgrade.

**Important**

Newcall policies are created on a per-service basis. If you have multiple services running on the chassis, you can configure multiple newcall policies.

The syntax for newcall policies is described below:

```plaintext
[local] host_name# newcall policy { asngw-service | asnpc-service | sgsn-service } { all | name service_name } reject
[local] host_name# newcall policy { fa-service | lns-service | mipv6ha-service } { all | name service_name } reject
[local] host_name# newcall policy { ha-service | pdsn-service | pdsnclosedrp-service } { all | name service_name } { redirect target_ip_address [ weight weight_num ] | target_ipaddress2 [ weight weight_num ] | ... target_ip_address16 [ weight weight_num ] | reject } [weight weight_num]...
[local] host_name# newcall policy ggsn-service { apn_name apn_name | all | name service_name } reject
[local] host_name# newcall policy hnbgw-service { all | name service_name } reject
[local] host_name# newcall policy { pcc-af-service | pcc-policy-service } { all | name service_name } reject
[local] host_name# newcall policy { pcc-af-service | pcc-policy-service } { all | name service_name } reject
[local] host_name# newcall policy mme-service { all | name service_name } reject
```

For complete information about the above commands, see the *Exec Mode Commands* chapter of the *Command Line Interface Reference*.

**Configure a Message of the Day Banner**

Optional: Configure a "Message of the Day" banner informing other management users that the system will be rebooted by entering the following command from the Global Configuration mode prompt.

```
[local] host_name#(config)# banner motd "banner_text"
```

*banner_text* is the message that you would like to be displayed and can be up to 2048 alphanumeric characters. Note that *banner_text* must begin with and end in quotation marks (" "). For more information in entering CLI banner information, see the *CLI Reference*. The banner is displayed when an administrative user logs onto the CLI.

**Back up the Current CLI Configuration File**

Back up the current CLI configuration file by entering the following command:

```
[local] host_name# copy from_url to_url [-noconfirm]
```

---

**Software Management Operations**

Off-line Software Upgrade
This creates a mirror-image of the CLI configuration file linked to the operating system defined in the current boot stack entry.

The following command example creates a backup copy of a file called `general.cfg` located on the `/flash` device to a file called `general_3652.cfg`:

```
[local] host_name copy /flash/general.cfg /flash/general_3652.cfg
```

### Create a New Boot Stack Entry

Create a new boot stack entry for the new file group, consisting of the new operating system image file and the currently used CLI configuration file by entering the following Global Configuration command:

```
[local] host_name(config)# boot system priority number image image_url /flash filename config cfg_url /flash/filename
```

Assign the next highest priority to this entry, by using the `<N-1>` method, wherein you assign a priority number that is one number less than your current highest priority.

---

**Important**

Run the Exec mode `show boot` command to verify that there are less than 10 entries in the `boot.sys` file and that a higher priority entry is available (minimally there is no priority 1 entry in the boot stack).

If priority 1 is in use, you must renumber the existing entries to ensure that at least that priority is available.

The maximum number of boot stack entries that can be contained in the `boot.sys` file is 10. If there are already 10 entries in the boot stack, you must delete at least one of these entries (typically, the lowest priority) and, if necessary, renumber some or all of the other entries before proceeding. Use the no boot system priority command to delete a boot stack entry.

```
[local] host_name(config)# no boot system priority number
```

To add new boot stack entries to the `boot.sys` file enter the following commands:

```
[local] host_name config
[local] host_name(config)# boot system priority number image image_url config cfg_url
```

For information on using the `boot system priority` command, refer to the Adding a New Boot Stack Entry, on page 103.

### Synchronize File Systems

Synchronize the local file systems on the management cards by entering the following command:

```
[local] host_name filesystem synchronize all
```

### Save the Running Configuration

Save the currently running, upgraded configuration prior to rebooting the chassis.

To save the running configuration to the current configuration file, enter the following command:

```
[local] host_name save configuration /flash
```

### Reboot the Chassis

Reboot the chassis by entering the following command:

```
[local] host_name reload [-noconfirm]
```
As the system reboots, it loads the new operating system software image and its corresponding CLI configuration file using the new boot stack entry configured earlier.

After the system reboots, establish a CLI session and enter the `show version` command to verify that the active software version is correct.

*Optional for PDSN:* If you are using the IP Pool Sharing Protocol during your upgrade, refer to Configuring IPSP Before the Software Upgrade in the PDSN Administration Guide.

### Verify the Running Software Version

After the system has successfully booted, verify that the new StarOS version is running by executing the Exec mode `show version` command.

```
[localhost_name] show version
```

### Restoring the Previous Software Image

If for some reason you need to undo the upgrade, perform the upgrade again except:

- Specify the locations of the upgrade software image and configuration files.

then

- Specify the locations of the original software image and configuration files.

### Upgrading ICSR Chassis

The procedure for upgrading primary and backup ICSR chassis is described in *Interchassis Session Recovery*. Essentially the procedure requires upgrading the primary and standby chassis using the off-line method while each is in standby mode.

### Performing Dynamic Software Updates

StarOS allows the runtime loading of plugins. All StarOS builds include a "default" baseline plugin.

This feature is currently used to dynamically update the detection logic used to filter P2P applications via the Application Detection and Control (ADC) feature.

Patching is the process used to install a plugin as an incremental update to a StarOS release. One plugin can be provided to multiple, compatible, concurrent product releases. A plugin is distributed in the form of a compressed distribution kit via the internet or by other means (USB stick, CD, etc.).

A plugin is a functional software entity that provides incremental updates to a pre-existing StarOS software component. Plugins have the characteristic of being dynamically loadable at runtime and do not require a system restart. A plugin has a name and one or more versions. All plugin names are known to the system at product release.

For complete information on the Dynamic Software Update process, refer to the *ADC Administration Guide*. 
Managing License Keys

License keys define capacity limits (number of allowed subscriber sessions) and available features on your system. Adding new license keys allows you to increase capacity and add new features as your subscriber base grows.

New System License Keys

New systems are delivered with no license keys installed. In most cases, you receive the license key in electronic format (usually through e-mail).

When a system boots with no license key installed a default set of restricted session use and feature licenses is installed. The following Exec Mode command lists the license information:

```
[local] host_name> show license information
```

| Important | With no license key installed, the session use licenses for PDSN, HA, GGSN, and L2TP LNS are limited to 10,000 sessions. |

The license keys on the ASR 5500 are stored in EEPROM on the chassis midplane. Both MIO/UMIOs access this EEPROM when booting.

Session Use and Feature Use Licenses

Session use and feature use licenses are software mechanisms that provide session limit controls and enable special features within the system. These electronic licenses are stored in the system's configuration file that is loaded as part of the system software each time the system is powered on or restarted.

- Session use licenses limit the number of concurrent sessions that a system is capable of supporting per service type and are acquired on an as-needed basis. This allows carriers to pay only for what they are using and easily increase capacity as their subscriber base grows.
- Feature use licenses enable specific features/functionality within the system and are distributed based on the total number of sessions supported by the system.

Installing New License Keys

Use the instructions below to install a new license key.
Cutting and Pasting the Key

If you have a copy of the license, use the following configuration to cut and paste just the license key part:

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
</table>
| Step 1 | From the Exec mode, enter the following:  
  ```
  configure
  license key license
  exit
  ```
  Where `license` is the license key string. The license can be an alphanumeric string of 1 through 1023 characters that is case sensitive. Copy the license key as shown in the example below, including the " (double-quote slash). Please note: this is not a functional license.
  ```
  VER=1|C1M=000-0000-00|C1S=03290231803|C2M=11-1111-11-1|C2S=
  STCB21M82003R80411A4|DOI=00000000000|DOE=000000000000000|ISS=1|NUM=13459|0000000000000000
  LSP=000000|LSH=000000|LSG=500000|LSL=5000000|FIS=Y|FR4=Y|FPP=Y|FCS=Y|FTC=Y|FMG=Y|FCR=Y|FSR=Y|FPM=Y|FID=Y|SIG=MCwCF\Esnsq6Ba/
  XdmyfLe7rHcD4sVP2bzAbQ31eHDoyyd6388jHsHD99sg36SGG267ghssja77
  ```
  End
| Step 2 | Verify that the license key just entered was accepted by entering the following command at the Exec mode prompt:  
  ```
  [local]host_name# show license key
  ```
  The new license key should be displayed. If it is not, return to the Global configuration mode and re-enter the key using the `license key` command.
  **Important** An invalid license will not be accepted. A Failure error will appear in the output of the `license key` command when you attempt to configure an invalid license key. If you use the `-force` option to install an invalid license key, the license will be placed into a 30-day grace period. StarOS will generate daily syslog error messages and SNMP traps during the grace period. The output of the `show license information` command will indicate "License State" as "Not Valid".
| Step 3 | Verify that the license key enabled the correct functionality by entering the following command:  
  ```
  [local]host_name# show license information
  ```
  All license keys and the new session capacity or functionality enabled should be listed. If the functionality or session capacity enabled by the new key is incorrect, please contact your service representative.
| Step 4 | Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.
  **Caution** Failure to save the new license key configuration in the current CLI configuration file will result in the loss of any of the new features enabled by the license key once the system is reloaded.

Adding License Keys to Configuration Files

License keys can be added to a new or existing configuration file.
License key information is maintained as part of the CLI configuration. Each time a key is installed or updated, you must re-save the configuration file.

**Important**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Open the configuration file to which the new license key commands are to be copied.</th>
</tr>
</thead>
</table>
| Step 2 | Copy the license as shown in the example, including the "\" (double-quote slash). Please note: this is not a functional license.

```
"\nVER=1|C1M=000-0000-00|C1S=03290231803|C2M=11-1111-11-1|C2S=\STCB21M82003R80411A4| 
DOI=0000000000|DOE=0000000000|ISS=1|NUM=13459|0000000000000000|LSP=000000|LSH=000000|
LSG=500000|LSL=500000|FIS=Y|FR4=Y|FPP=Y|FCS=Y|FTC=Y|FMR=Y|FMR=Y|FID=Y|
SIG=MCwCF\Esaq6Bs/XdmyfhE7rHcD4aVp2bzAhQ3eHDcyyd6388jHsHd99ag36SG267gshssja77
end
```

| Step 3 | Paste the license key into the configuration |
|        | **Important** Paste the license key information at the beginning of the configuration file to ensure the system has the expected capacity and features before it configures contexts. |
|        | Step 4 | Save your configuration as described in the *Verifying and Saving Your Configuration* chapter. |

**License Expiration Behavior**

When a license expires, there is a built-in grace period of 30 days that allows normal use of the licensed session use and feature use licenses. This allows you to obtain a new license without any interruption of service.

The following Exec mode command lists the license information including the date the grace period is set to expire:

```
show license information
```

**Requesting License Keys**

License keys for the system can be obtained through your Cisco account representative. Specific information is required before a license key may be generated:

- Sales Order or Purchase Order information
- Desired session capacity
- Desired functionality
- Midplane (chassis) serial number

To obtain the ASR 5500 chassis serial number, at the Exec mode prompt enter the `show card hardware 5` command. Look under the "MEC" heading for the "UDI Serial Number" as shown in the example below:

```
MEC:
  Description : MEC
  Cisco Part Number : 73-14501-01 A0
  UDI Serial Number : FLM154300D8
  UDI Product ID : ASR55-MEC
  UDI Version ID : V01
```
The ICSR license key for Active and Standby chassis are uniquely coded to each chassis. Two separate license keys are required.

Viewing License Information

To see the license detail, enter the following command from the Exec mode:

```
[local]host_name# show license information [ full | key [ full ] ]
```

Deleting a License Key

Use the procedure below to delete the session and feature use license key from a configuration. You must be a security administrator or administrator.

```
configure
    no license key
exit
show license key
```

The output of this command should display: "No license key installed".

Management Card Replacement and License Keys

License keys are stored on a midplane EEPROM in the ASR 5500 chassis. The MIO/UMIOs share these license keys. There is no need to swap memory cards into replacement MIO/UMIOs.

Managing Local-User Administrative Accounts

Unlike context-level administrative accounts which are configured via a configuration file, information for local-user administrative accounts is maintained in a separate file in flash memory and managed through the software's Shared Configuration Task (SCT). Because local-user accounts were designed to be compliant with ANSI T1.276-2003, the system provides a number of mechanisms for managing these types of administrative user accounts.

Configuring Local-User Password Properties

Local-user account password properties are configured globally and apply to all local-user accounts. The system supports the configuration of the following password properties:

- **Complexity**: Password complexity can be forced to be compliant with ANSI T1.276-2003.
- **History length**: How many previous password versions should be tracked by the system.
- **Maximum age**: How long a user can use the same password.
- **Minimum number of characters to change**: How many characters must be changed in the password during a reset.
• **Minimum change interval:** How often a user can change their password.
• **Minimum length:** The minimum number of characters a valid password must contain.

Refer to the `local-user password` command in the *Global Configuration Mode Commands* chapter of the *Command Line Interface Reference* for details on each of the above parameters.

## Configuring Local-User Account Management Properties

Local-user account management includes configuring account lockouts and user suspensions.

### Local-User Account Lockouts

Local-user accounts can be administratively locked for the following reasons:

- **Login failures:** The configured maximum login failure threshold has been reached. Refer to the `local-user max-failed-logins` command in the *Global Configuration Mode Commands* chapter of the *Command Line Interface Reference* for details.

- **Password Aging:** The configured maximum password age has been reached. Refer to the `local-user password` command in the *Global Configuration Mode Commands* chapter of the *Command Line Interface Reference* for details.

Accounts that are locked out are inaccessible to the user until either the configured lockout time is reached (refer to the `local-user lockout-time` command in the *Global Configuration Mode Commands* chapter of the *Command Line Interface Reference*) or a security administrator clears the lockout (refer to the `clear local-user` command in the *Exec Mode Commands* chapter of the *Command Line Interface Reference*).

**Important**

Local-user administrative user accounts could be configured to enforce or reject lockouts. Refer to the `local-user username` command in the *Global Configuration Mode Commands* chapter of the *Command Line Interface Reference* for details.

### Local-User Account Suspensions

Local-user accounts can be suspended as follows:

```
configure
  suspend local-user  name
```

A suspension can be removed by entering:

```
configure
  no suspend local-user  name
```

### Changing Local-User Passwords

Local-user administrative users can change their passwords using the `password change` command in the Exec mode. Users are prompted to enter their current and new passwords.
Security administrators can reset passwords for local-users by entering the following command from the root prompt in the Exec mode:

```
[local] host_name# password change username  name
```

`name` is the name of the local-user account for which the password is to be changed. When a security administrator resets a local-user's password, the system prompts the user to change their password the next time they login.

All new passwords must adhere to the password properties configured for the system.
Monitoring the System

This chapter provides information for monitoring system status and performance using the show commands found in the Command Line Interface (CLI). These command have many related keywords that allow them to provide useful information on all aspects of the system ranging from current software configuration through call activity and status.

The selection of keywords described in this chapter is intended to provide the most useful and in-depth information for monitoring the system. For additional information on these and other show command keywords, refer to the Exec Mode show Commands chapter of the Command Line Interface Reference.

- SNMP Notifications, page 119
- Monitoring System Status and Performance, page 119
- Clearing Statistics and Counters, page 121
- Monitoring ASR 5500 Hardware Status, page 121

SNMP Notifications

In addition to the CLI, the system supports Simple Network Management Protocol (SNMP) notifications that indicate status and alarm conditions. Refer to the SNMP MIB Reference for a detailed listing of these notifications.

Monitoring System Status and Performance

This section contains commands used to monitor the status of tasks, managers, applications and other software components in the system. Output descriptions for most of the commands are located in the Statistics and Counters Reference.

Table 7: System Status and Performance Monitoring Commands

<table>
<thead>
<tr>
<th>To do this:</th>
<th>Enter this command:</th>
</tr>
</thead>
<tbody>
<tr>
<td>View Administrative Information</td>
<td></td>
</tr>
<tr>
<td>Display Current Administrative User Access</td>
<td></td>
</tr>
</tbody>
</table>
### To do this:

<table>
<thead>
<tr>
<th>Enter this command:</th>
<th>View information about all currently outstanding alarms</th>
<th>show alarm outstanding all verbose</th>
</tr>
</thead>
<tbody>
<tr>
<td>View system uptime (time since last reboot)</td>
<td>show system uptime</td>
<td>show system uptime</td>
</tr>
<tr>
<td>View NTP servers status</td>
<td>show ntp status</td>
<td>show ntp status</td>
</tr>
<tr>
<td>View all system resources such as CPU resources and number of managers created</td>
<td>show resources [ cpu ]</td>
<td>show resources [ cpu ]</td>
</tr>
<tr>
<td>View information pertaining to local-user administrative accounts configured for the system</td>
<td>show local-user verbose</td>
<td>show local-user verbose</td>
</tr>
<tr>
<td>View statistics for local-user administrative accounts</td>
<td>show local-user statistics verbose</td>
<td>show local-user statistics verbose</td>
</tr>
<tr>
<td>View information pertaining to your CLI session</td>
<td>show cli</td>
<td>show cli</td>
</tr>
<tr>
<td>Determining System Uptime</td>
<td>View NTP Server Status</td>
<td>Display SNMP Notification Statistics</td>
</tr>
<tr>
<td>View System Resources</td>
<td>View NTP servers status</td>
<td>View SNMP notification statistics</td>
</tr>
<tr>
<td></td>
<td>View all system resources such as CPU resources and number of managers created</td>
<td>show snmp notifies</td>
</tr>
<tr>
<td></td>
<td>View information pertaining to local-user administrative accounts</td>
<td>show snmp accesses</td>
</tr>
<tr>
<td></td>
<td>Show System Alarms</td>
<td>Display SNMP Trap History</td>
</tr>
<tr>
<td></td>
<td>View information about all currently outstanding alarms</td>
<td>View SNMP trap history</td>
</tr>
<tr>
<td></td>
<td>View system alarm statistics</td>
<td>show snmp trap history</td>
</tr>
<tr>
<td></td>
<td>View Congestion-Control Statistics</td>
<td>Display SNMP Trap Statistics</td>
</tr>
<tr>
<td></td>
<td>View Congestion-Control Statistics</td>
<td>View SNMP Trap Statistics</td>
</tr>
<tr>
<td></td>
<td>Display SNMP Access Statistics</td>
<td>show snmp trap statistics</td>
</tr>
<tr>
<td></td>
<td>View SNMP access statistics</td>
<td>Display ORBEM Information</td>
</tr>
<tr>
<td></td>
<td>Display SNMP Trap History</td>
<td>View ORBEM Information</td>
</tr>
<tr>
<td></td>
<td>View SNMP trap history</td>
<td>View ORBEM client status</td>
</tr>
<tr>
<td></td>
<td>Display SNMP Trap Statistics</td>
<td>show orbem client id</td>
</tr>
<tr>
<td></td>
<td>View SNMP Trap Statistics</td>
<td>View ORBEM session information</td>
</tr>
<tr>
<td></td>
<td>Display ORBEM Information</td>
<td>show orbem session table</td>
</tr>
</tbody>
</table>
**Clearing Statistics and Counters**

It may be necessary to periodically clear statistics and counters in order to gather new information. The system provides the ability to clear statistics and counters based on their grouping (PPP, MIPHA, MIPFA, etc.). Statistics and counters can be cleared using the CLI `clear` command. Refer to the `Exec Mode Commands` chapter of the *Command Line Interface Reference* for detailed information on using this command.

**Monitoring ASR 5500 Hardware Status**

Use the commands contained in this section to monitor the status of the hardware components in the chassis. For output descriptions for most of the commands, refer to the *Statistics and Counters Reference*.

---

**Important**
The commands or keywords and variables are dependent on platform type, product version, and installed license(s). Some commands produce different outputs, depending on the platform type.

---

**Table 8: Hardware Monitoring Commands**

<table>
<thead>
<tr>
<th>To do this:</th>
<th>Enter this command:</th>
</tr>
</thead>
<tbody>
<tr>
<td>View the Status of the Power System</td>
<td><code>show power chassis</code></td>
</tr>
<tr>
<td>View the status of the PFUs</td>
<td></td>
</tr>
<tr>
<td>To do this:</td>
<td>Enter this command:</td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>-------------------------------------------</td>
</tr>
<tr>
<td>View the power status of the individual chassis slots</td>
<td>show power all</td>
</tr>
<tr>
<td><strong>View the Status of the Fan Trays</strong></td>
<td></td>
</tr>
<tr>
<td>View the status of the fan trays, including current</td>
<td>show fans</td>
</tr>
<tr>
<td>relative speeds and temperatures.</td>
<td></td>
</tr>
<tr>
<td><strong>Determine the Status of Installed Cards</strong></td>
<td></td>
</tr>
<tr>
<td>View a listing of installed application cards</td>
<td>show card table</td>
</tr>
<tr>
<td><strong>Perform a Hardware Inventory</strong></td>
<td></td>
</tr>
<tr>
<td>View all cards installed in the chassis and their</td>
<td>show hardware inventory</td>
</tr>
<tr>
<td>hardware revision, part, serial, assembly, and</td>
<td></td>
</tr>
<tr>
<td>fabrication numbers</td>
<td></td>
</tr>
<tr>
<td>View details of a specific card. Output contains</td>
<td>show hardware card slot_number</td>
</tr>
<tr>
<td>same information as output of both show hardware</td>
<td></td>
</tr>
<tr>
<td>inventory and show hardware version board</td>
<td></td>
</tr>
<tr>
<td><strong>View Card Diagnostics</strong></td>
<td></td>
</tr>
<tr>
<td>View boot, power and temperature diagnostics</td>
<td>show card diag slot_number</td>
</tr>
<tr>
<td>View runtime, or real time, information</td>
<td>show card info slot_number</td>
</tr>
<tr>
<td><strong>View the LED Status of All Installed Cards</strong></td>
<td></td>
</tr>
<tr>
<td>Note Refer to the descriptions of card-level and</td>
<td></td>
</tr>
<tr>
<td>system-level LEDs in the ASR 5500 Installation Guide</td>
<td></td>
</tr>
<tr>
<td>View the LED status for all installed cards</td>
<td>show leds all</td>
</tr>
<tr>
<td><strong>View Available Physical Ports</strong></td>
<td></td>
</tr>
<tr>
<td>View ports that are available to the system</td>
<td>show port table</td>
</tr>
<tr>
<td>View detailed information for a specific port</td>
<td>show port info slot_number/port_number</td>
</tr>
<tr>
<td><strong>View CPU Resource Information</strong></td>
<td></td>
</tr>
<tr>
<td>View CPU resource information</td>
<td>show resource cpu</td>
</tr>
<tr>
<td>View CPU resources</td>
<td>show resources { cpu</td>
</tr>
<tr>
<td>View CPU usage information</td>
<td>show cpu table; show cpu info</td>
</tr>
<tr>
<td><strong>View Component Temperature Information</strong></td>
<td></td>
</tr>
<tr>
<td>View current component temperatures</td>
<td>show temperature</td>
</tr>
<tr>
<td>View maximum temperatures reached since last</td>
<td>show maximum-temperatures</td>
</tr>
<tr>
<td>timestamp.</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 10

Bulk Statistics

This chapter provides configuration information for:

- Configuring Communication with the Collection Server, page 123
- Viewing Collected Bulk Statistics Data, page 126
- Manually Gathering and Transferring Bulk Statistics, page 127
- Clearing Bulk Statistics Counters and Information, page 127
- Bulkstats Schema Nomenclature, page 127
- Bulk Statistics Event Log Messages, page 130

Configuring Communication with the Collection Server

Two configuration methods are available for defining how bulk statistics are collected and managed. A "standard" configuration allows the system to automatically assign a number to the bulk statistics file. Optionally, a number can be specified by an administrator in the optional configuration method. Command details and descriptions of keywords and variables for commands in this chapter are located in the Bulk Statistics Configuration Mode Commands and Bulk Statistics File Configuration Mode Commands chapters in the Command Line Interface Reference.

Configuring Standard Settings

The configuration example in this section defines basic operation of the bulk statistics feature. Use the following example configuration to set up the system to communicate with the statistic collection server:

```
configure
  bulkstats mode
    schema name format format_string
    sample-interval time interval
    transfer-interval xmit_time_interval
    limit mem_limit
  exit
    bulkstats collection
end
```
Configuring Optional Settings

This section describes optional commands that can be used within the Bulk Statistics Configuration mode. Specifically, you can configure bulk statistic "files" under which to group the bulk statistics. "Files" are used to group bulk statistic schema, delivery options, and receiver configuration. Because multiple "files" can be configured, this functionality provides greater flexibility because it allows you to configure different schemas to go to different receivers.

```
configure
  bulkstats mode
    file number
      receiver ip_address { primary | secondary }
      [ mechanism { { ftp | sftp } login user_name } ]
      [ encrypted ]
      password pwd
    receiver mode { { redundant | secondary-on-failure } }
    remotefile format naming_convention [ both-receivers | primary-receiver | secondary-receiver ]
    header format header_format
    footer format footer_format
    exit
  schema_type schema_format format_string
  sample-interval time_interval
  transfer-interval xmit_time_interval
  limit mem_limit
  exit
bulkstats collection
end
```

In release 20.0 and higher Trusted StarOS builds, FTP is not supported. SFTP is the recommended transfer protocol.

Configuring Bulk Statistic Schemas

In each configuration example described in Configuring Standard Settings, on page 123 and Configuring Optional Settings, on page 124, the following is the primary command used to configure the type of schema and the statistics collected:

```
configure
  bulkstats mode
    schema_type schema_format format_string
```

Refer to the Bulk Statistics Configuration Mode Commands and Bulk Statistics File Configuration Mode Commands chapters in the Command Line Interface Reference for more information regarding supported schemas, available statistics, and proper command syntax.

Configuring a Separate Bulkstats Config File

You can configure a separate destination config file for storing the bulk statistics sub-mode configuration. Run the show configuration bulkstats command to confirm the configuration.
The bulkstats configuration file stores the configuration that was previously stored in the system configuration file under the bulk statistics sub-mode.

The Global Configuration mode `bulkstats config` command creates the separate configuration file in the system configuration.

```
configure
  [no] bulkstats config url
end
```

`url` specifies the location of the bulkstats configuration file. If the destination file already exists, it is replaced with the new file. The new file will only be created if you save the configuration after completing changes. The Exec mode `show configuration bulkstats` command displays the URL for the bulkstats configuration mode destination file if it has been configured.

You can copy the bulkstats configuration file from the `url`, edit it and copy it back to `/flash`. Changes can be applied by using the `no` form of the `bulkstats config` command followed by reconfiguring the `bulkstats` command.

When the new `bulkstats config` command is enabled, StarOS removes the existing bulk statistics sub-mode configuration from the system configuration file. You must save the system configuration to retain the configuration change.

If `no bulkstats config` is used to disable the new destination file after it has been enabled. StarOS does not remove the file. You must save the system configuration to retain the configuration change.

---

**Important**

After completing changes to the bulk statistics configuration, you must save the system configuration to save the changes. If the `bulkstats config` command is enabled, the bulkstats configuration file will be updated.

---

### Using show bulkstats Commands

There are several Exec mode `show bulkstats` commands that display information about defined parameters.

- `show bulkstats data` – displays criteria contained in the statistics gathering scheme for up to four files. See *Viewing Collected Bulk Statistics Data*, on page 126.
- `show bulkstats schemas` – displays the scheme used to gather statistics including collection and transmission statistics. See *Verifying Your Configuration*, on page 125.
- `show bulkstats variables` – displays available bulkstat variables (\%variable\%) by schema type that can be incorporated into a schema format.

### Verifying Your Configuration

After configuring support for bulk statistics on the system, you can check your settings prior to saving them. Follow the instructions in this section to verify your bulk statistic settings. These instructions assume that you are at the root prompt for the Exec mode.

Check your collection server communication and schema settings by entering the following Exec mode command:

```
show bulkstats schemas
```
The following is an example command output:

**Bulk Statistics Server Configuration:**
- **Server State:** Enabled
- **File Limit:** 6000 KB
- **Sample Interval:** 15 minutes (0D 0H 15M)
- **Transfer Interval:** 480 minutes (0D 0H 15M)
- **Collection Mode:** Cumulative
- **Receiver Mode:** Secondary-on-failure
- **Local File Storage:** None

**Bulk Statistics Server Statistics:**
- Records awaiting transmission: 114
- Bytes awaiting transmission: 8032
- Total records collected: 59926
- Total bytes collected: 4190178
- Total records transmitted: 59812
- Total bytes transmitted: 4188512
- Total records discarded: 0
- Total bytes discarded: 0
- Last collection time required: 2 second(s)
- Last transfer time required: 0 second(s)
- Last successful transfer: Wednesday December 7 12:14:30 EDT 2011
- Last successful tx recs: 190
- Last successful tx bytes: 13507
- Last attempted transfer: Wednesday December 7 12:14:30 EDT 2011

**File 1**
- **Remote File Format:** /users/ems/server/data/chicago/bulkstat%date%%time%.txt
- **File Header:** "CHI_test %time%"
- **File Footer:** ""

**Bulkstats Receivers:**
- **Primary:** 192.168.0.100 using FTP with username administrator
- Records awaiting transmission: 0
- Bytes awaiting transmission: 0
- Total records collected: 0
- Total bytes collected: 0
- Total records transmitted: 0
- Total bytes transmitted: 0
- Total records discarded: 0
- Total bytes discarded: 0
- Last transfer time required: 0 second(s)
- No successful data transfers
- No attempted data transfer

File 2 not configured
File 3 not configured
File 4 not configured

---

**Saving Your Configuration**

Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

**Viewing Collected Bulk Statistics Data**

The system provides a mechanism for viewing data that has been collected but has not been transferred. This data is referred to as "pending data".

View pending bulk statistics data per schema by entering the following Exec mode command:

```
show bulkstats data
```

The above command also shows the statistics of remote files, if configured as described in *Configuring Optional Settings*, on page 124.
Manually Gathering and Transferring Bulk Statistics

There may be times where it is necessary to gather and transfer bulk statistics outside of the scheduled intervals. The system provides commands that allow you to manually initiate the gathering and transferring of bulk statistics.

To manually initiate the gathering of bulk statistics outside of the configured sampling interval, enter the following Exec mode command:

`bulkstats force gather`

To manually initiate the transferring of bulk statistics prior to reaching the of the maximum configured storage limit, enter the following Exec mode command:

`bulkstats force transfer`

Clearing Bulk Statistics Counters and Information

It may be necessary to periodically clear counters pertaining to bulk statistics in order to gather new information or to remove bulk statistics information that has already been collected. The following Exec mode command can be used to perform either of these functions:

`clear bulkstats { counters | data }`

The `clear bulkstats data` command clears any accumulated data that has not been transferred. This includes any "completed" files that have not been successfully transferred.

Bulkstats Schema Nomenclature

This section describes the nomenclature associated with configuring and viewing bulkstats.

Statistic Types

The following statistic types are defined in the Statistics and Counters Reference user document (published prior to Release 20.0) and displayed in the output of the Exec mode `show bulkstats variables` command:

- **Counter**: A counter records incremental data cumulatively and rolls over when the counter limit is reached.
  - All counter statistics are cumulative and reset only by one of the following methods: roll-over when the limit is reached, after a system restart, or after a clear command is performed.
  - The limit depends upon the data type.

- **Gauge**: A gauge statistic indicates a single value; a snapshot representation of a single point in time within a defined time frame. The gauge changes to a new value with each snapshot though a value may repeat from one period to the next. The limit depends upon the data type.

- **Information**: This type of statistic provides information, often intended to differentiate sets of statistics; for example, a VPN name or IP address. The type of information provided depends upon the data type.
The following statistic types are included in the Statistics and Counters Reference spreadsheet published after release 20.0 to replace the original user document:

- **Incremental**: An incremental data type records incremental data cumulatively and rolls over when the counter limit is reached.
  - All incremental statistics are cumulative and reset only by one of the following methods: roll-over when the limit is reached, after a system restart, or after a clear command is performed.
  - The limit depends upon the data type.

- **Gauge**: A gauge statistic indicates a single value; a snapshot representation of a single point in time within a defined time frame. The gauge changes to a new value with each snapshot though a value may repeat from one period to the next. The limit depends upon the data type.

- **Primary-key**: This type of statistic provides information, often intended to differentiate sets of statistics; for example, a VPN name or IP address. The type of information provided depends upon the data type.

### Data Types

The data type defines the format of the data for the value provided by the statistic. The following data types appear in the Statistics and Counters Reference and the output of the Exec mode `show bulkstats variables` command:

- **Int32**: A 32-bit integer; the roll-over to zero limit is 4,294,967,295.
- **Int64**: A 64-bit integer; the roll-over to zero limit is 18,446,744,073,709,551,615.
- **Float**: A numeric value that includes decimal points; for example, 1.345.
- **String**: A series of ASCII alphanumeric characters in a single grouping, usually pre-configured.

### Key Variables

Every schema has some variables which are typically referred to as "key variables". These key variables provide index markers to identify to which object the statistics apply. For example, in the card schema the card number (variable %card%) uniquely identifies a card. For an HA service, the keys would be "%vpnname%" plus "%servname%", as the combination uniquely identifies an HA service. So, in a given measurement interval, one row of statistics will be generated per unique key.

There are also a number of common variables shared across schema that identify time, date, place, etc. These common variables are identified in the table below.

<table>
<thead>
<tr>
<th>Variables</th>
<th>Description</th>
<th>Statistic Type</th>
<th>Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>uptime</td>
<td>The total uptime (in seconds) of the system that created the file.</td>
<td>Information</td>
<td>Int32</td>
</tr>
<tr>
<td>host</td>
<td>The system hostname that created the file.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>Variables</td>
<td>Description</td>
<td>Statistic Type</td>
<td>Data Type</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>ipaddr</td>
<td>The default management (local context) IP address in dPv4 dotted-decimal format. An empty string is inserted if no address is available.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>date</td>
<td>The UTC date that the collection file was created in YYYYMMDD format where YYYY represents the year, MM represents the month and DD represents the day.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>date3</td>
<td>The UTC date that the collection file was created in YYMMDD format where YY represents the year, MM represents the month and DD represents the day.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>time</td>
<td>The UTC time that the collection file was created in HHMMSS format where HH represents the hours, MM represents the minutes, and SS represents the seconds.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>time2</td>
<td>The UTC time that the collection file was created in HH:MM:SS format where HH represents the hours, MM represents the minutes, and SS represents the seconds.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>time3</td>
<td>The UTC time that the collection file was created in HH:MM format where HH represents the hours, MM represents the minutes.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>epochtime</td>
<td>The number of seconds since Jan 1, 1970, 00:00:00 GMT.</td>
<td>Information</td>
<td>In32</td>
</tr>
<tr>
<td>schemas</td>
<td>Lists all bulkstat schemas available on this platform.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>schemas-delta</td>
<td>Lists all bulkstats schemas that have changed the schema list was last output.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>localdate</td>
<td>The date (adjusted for the local timezone) that the collection file was created in YYYYMMDD format where YYYY represents the year, MM represents the month and DD represents the day.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>localdate3</td>
<td>The date that the collection file was created in YYYYMMDD format where YYYY represents the year, MM represents the month and DD represents the day. The date displays in local time, not UTC.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>localtime</td>
<td>The time (adjusted for the local timezone) that the collection file was created in HHMMSS format where HH represents the hours, MM represents the minutes, and SS represents the seconds.</td>
<td>Information</td>
<td>String</td>
</tr>
<tr>
<td>localtime2</td>
<td>The time (adjusted for the local timezone) that the collection file was created in HH:MM:SS format where HH represents the hours, MM represents the minutes, and SS represents the seconds.</td>
<td>Information</td>
<td>String</td>
</tr>
</tbody>
</table>
The stat logging facility captures several events that can be useful for diagnosing errors that could occur with either the creation or writing of a bulk statistic data set to a particular location.

The following table displays information pertaining to these events.

**Table 10: Logging Events Pertaining to Bulk Statistics**

<table>
<thead>
<tr>
<th>Event</th>
<th>Event ID</th>
<th>Severity</th>
<th>Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Local File Open Error</td>
<td>31002</td>
<td>Warning</td>
<td>&quot;Unable to open local file filename for storing bulkstats data&quot;</td>
</tr>
<tr>
<td>Receiver Open Error</td>
<td>31018</td>
<td>Warning</td>
<td>&quot;Unable to open url filename for storing bulkstats data&quot;</td>
</tr>
<tr>
<td>Receiver Write Error</td>
<td>31019</td>
<td>Warning</td>
<td>&quot;Unable to write to url filename while storing bulkstats data&quot;</td>
</tr>
<tr>
<td>Receiver Close Error</td>
<td>31020</td>
<td>Warning</td>
<td>&quot;Unable to close url filename while storing bulkstats data&quot;</td>
</tr>
</tbody>
</table>
System Logs

This chapter describes how to configure parameters related to the various types of logging and how to viewing their content. It includes the following sections:

- System Log Types, page 131
- Configuring Event Logging Parameters, page 132
- Configuring Active Logs, page 134
- Specifying Facilities, page 135
- Configuring Trace Logging, page 143
- Configuring Monitor Logs, page 144
- Viewing Logging Configuration and Statistics, page 144
- Viewing Event Logs Using the CLI, page 145
- Configuring and Viewing Crash Logs, page 146
- Checkpointing Logs, page 149
- Saving Log Files, page 150
- Event ID Overview, page 150

System Log Types

There are five types of logs that can be configured and viewed on the system:

- **Important**: Not all Event Logs can be configured on all products. Configurability depends on the hardware platform and licenses in use.

- **Event**: Event logging can be used to determine system status and capture important information pertaining to protocols and tasks in use by the system. This is a global function that will be applied to all contexts, sessions, and processes.
• **Active**: Active logs are operator configurable on a CLI instance-by-CLI instance basis. Active logs configured by an administrative user in one CLI instance cannot be viewed by an administrative user in a different CLI instance. Each active log can be configured with filter and display properties that are independent of those configured globally for the system. Active logs are displayed in real time as events are generated.

• **Trace**: Trace logging can be used to quickly isolate issues that may arise for a particular connected subscriber session. Traces can be taken for a specific call identification (callid) number, IP address, mobile station identification (MSID) number, or username.

• **Monitor**: Monitor logging records all activity associated with a particular session. This functionality is available in order to comply with law enforcement agency requirements for monitoring capabilities of particular subscribers. Monitors can be performed based on a subscriber's MSID or username.

• **Crash**: Crash logging stores useful information pertaining to system software crashes. This information is useful in determining the cause of the crash.

### Configuring Event Logging Parameters

The system can be configured to generate logs based on user-defined filters. The filters specify the facilities (system tasks or protocols) that the system is to monitor and severity levels at which to trigger the generation of the event entries.

Event logs are stored in system memory and can be viewed via the CLI. There are two memory buffers that store event logging information. The first buffer stores the active log information. The second buffer stores inactive logging information. The inactive buffer is used as a temporary repository to allow you to view logs without having data be overwritten. Logs are copied to the inactive buffer only through manual intervention. Each buffer can store up to 50,000 events. Once these buffers reach their capacity, the oldest information is removed to make room for the newest.

To prevent the loss of log data, the system can be configured to transmit logs to a syslog server over a network interface.

---

**Important**

For releases after 15.0 MR4, TACACS+ accounting (CLI event logging) will not be generated for Lawful Intercept users (priv-level 15 and 13).

### Configuring Event Log Filters

You can filter the contents of event logs at the Exec mode and Global Configuration mode levels. For additional information, see the *Command Line Interface Reference*.

**Exec Mode Filtering**

These commands allow you to limit the amount of data contained in logs without changing global logging parameters.
Follow the example below to filter logs via Exec mode commands.

```
logging active | copy runtime filters | [ event-verbosity event_level ] | [ pdu-data format ] | [ pdu-verbosity pdu_level ]
```

Notes:

- **copy runtime filters** – Copies the runtime filters and uses that copy to filter the current logging session.
- **event-verbosity** *event_level* – Specifies the level of verboness to use in logging of events as one of:
  - **min** – Displays minimal information about the event. Information includes event name, facility, event ID, severity level, date, and time.
  - **concise** – Displays detailed information about the event, but does not provide the event source within the system.
  - **full** – Displays detailed information about event, including source information, identifying where within the system the event was generated.

- **pdu-data format** – Specifies output format for packet data units when logged as one of:
  - **none** – raw format (unformatted).
  - **hex** – hexadecimal format
  - **hex-ascii** – hexadecimal and ASCII similar to a main-frame dump

- **pdu-verbosity** *pdu_level* – Specifies the level of verbosity to use in logging of packet data units as an integer from 1 through 5, where 5 is the most detailed.

Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.

### Global Configuration Mode Filtering

You can filter the contents of event logs at the Exec mode and Global Configuration mode levels. Follow the example below to configure run time event logging parameters for the system:

```
configure
  logging filter runtime facility facility_level report_level
  logging display { event-verbosity | pdu-data | pdu-verbosity }
end
```

Notes:

- **facility facility** and **level severity_level** – Configure the logging filter that determines which system facilities should be logged and at what levels. For detailed information, see Specifying Facilities, on page 135 and Event Severities, on page 158.
- Repeat for every facility that you would like to log.
- **Optional:** Configure event ID restrictions by adding the **logging disable eventid** command. The system provides the ability to restrict the sending of a specific event ID or a range of event IDs to minimize the amount of data logged to that which is most useful. Repeat to disable logging for additional event IDs or event ID ranges.

Save the configuration as described in the *Verifying and Saving Your Configuration* chapter.
Configuring syslog Servers

Information generated by the run time event logging filters can be transmitted to a syslog server for permanent storage.

Important

The data transmitted to the syslog server is meant to be used for informational purposes. Functions such as billing and performance monitoring should not be based on syslogs.

Important

Although the system provides the flexibility to configure syslog servers on a context-by-context basis, it is recommended that all servers be configured in the local context in order to isolate the log traffic from the network traffic.

Use the following example to configure syslog servers:

```
configure
  context local
    logging syslog ip_address
  end
```

Notes:

- `ip_address` specifies the IP address of a system log server on the network in IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation.
- A number of keyword options/variables are available for the `logging syslog` command. Refer to the Context Configuration Mode Commands chapter in the Command Line Interface Reference for more information.
- Repeat as needed to configure additional syslog servers. There is no limit to the number of syslog servers that can be configured.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

Configuring Active Logs

Active logs are event logs that are operator configurable on a CLI instance-by-CLI instance basis. Active logs configured by an administrative user in one CLI instance are not displayed to an administrative user in a different CLI instance. Each active log can be configured with filter and display properties that are independent of those configured globally for the system. Active logs are displayed in real time as they are generated.

Active logs are not written to the active memory buffer by default. To write active logs to the active memory buffer execute the following command in the Global Configuration mode:

```
[local] host_name(config)# logging runtime buffer store all-events
```

When active logs are written to the active memory buffer, they are available to all users in all CLI instances.

Use the following example to configure active logging in Global Configuration mode:

```
[local] host_name(config)# logging filter runtime facility facility level report_level
```

Notes:
Configure the logging filter that determines which system facilities should be logged and at what levels. For detailed information, see Specifying Facilities, on page 135 and Event Severities, on page 158.

Repeat for every facility that you would like to log.

Optional: Configure event ID restrictions by adding the logging disable eventid command. The system provides the ability to restrict the sending of a specific event ID or a range of event IDs to minimize the amount of data logged to that which is most useful. Repeat to disable logging for additional event IDs or event ID ranges.

A number of keyword options/variables are available for the Exec mode logging active command. Refer to the Exec Mode Commands chapter in the Command Line Interface Reference for more information.

Once all of the necessary information has been gathered, the Active log display can be stopped by entering the following command in the Exec mode:

no logging active

## Specifying Facilities

**Important**  
The actual facilities available for logging vary by platform type, StarOS version and installed product licenses.

The following facilities can be configured for logging event data:

- **a10**: A10 interface facility
- **a11**: A11 interface facility
- **a11mgr**: A11 Manager facility
- **aaa-client**: Authentication, Authorization and Accounting (AAA) client facility
- **aaamgr**: AAA manager logging facility
- **aaaproxy**: AAA Proxy facility
- **aal2**: ATM Adaptation Layer 2 (AAL2) protocol logging facility
- **acl-log**: Access Control List (ACL) logging facility
- **acsctrl**: Active Charging Service (ACS) Controller facility
- **acsmgr**: ACS Manager facility
- **afctrl**: Fabric Controller facility [ASR 5500 only]
- **afmgr**: Fabric Manager logging facility [ASR 5500 only]
- **alarmctrl**: Alarm Controller facility
- **alcap**: Access Link Control Application Part (ALCAP) protocol logging facility
- **alcapmgr**: ALCAP manager logging facility
- **all**: All facilities
- **bfd**: Bidirectional Forwarding Detection (BFD) protocol logging facility
• **bgp**: Border Gateway Protocol (BGP) facility

• **bindmux**: IPCF BindMux-Demux Manager logging facility

• **bngmgr**: Broadband Network Gateway (BNG) Demux Manager logging facility

• **bssap+**: Base Station Sub-system Application Part+ protocol facility for the login interface between the SGSN and the MSC/VLR (2.5G and 3G)

• **bssgp**: Base Station Sub-system GPRS Protocol logging facility handles exchange information between the SGSN and the BSS (2.5G only)

• **callhome**: Call Home application logging facility

• **cap**: CAMEL Application Part (CAP) logging facility for protocol used in prepaid applications (2.5G and 3G)

• **cbsmgr**: Cell Broadcasting Service (CBS) logging facility [HNBGW]

• **cdf**: Charging Data Function (CDF) logging facility

• **cgw**: Converged Access Gateway (CGW) logging facility

• **cli**: Command Line Interface (CLI) logging facility

• **cmp**: Certificate Management Protocol (IPSec) logging facility

• **confdmgr**: ConfD Manager proclet (NETCONF) logging facility

• **connectedapps**: SecGW ASR 9000 oneP communication protocol

• **connproxy**: Controller Proxy logging facility

• **credit-control**: Credit Control (CC) facility

• **esp**: Card/Slot/Port controller facility

• **css**: Content Service Selection (CSS) facility

• **css-sig**: CSS RADIUS Signaling facility

• **data-mgr**: Data Manager Framework logging facility

• **dcardctrl**: IPSec Daughter Card Controller logging facility

• **dcardmgr**: IPSec Daughter Card Manager logging facility

• **demuxmgr**: Demux Manager API facility

• **dgmbmgr**: Diameter Gmb Application Manager logging facility

• **dhcp**: Dynamic Host Configuration Protocol (DHCP) logging facility

• **dhcpv6**: DHCPv6

• **dhost**: Distributed Host logging facility

• **diabase**: Diabase messages facility

• **diactrl**: Diameter Controller proclet logging facility

• **diameter**: Diameter endpoint logging facility

• **diameter-acct**: Diameter Accounting
- **diameter-auth**: Diameter Authentication
- **diameter-dns**: Diameter DNS subsystem
- **diameter-ecs**: ACS Diameter signaling facility
- **diameter-engine**: Diameter version2 engine logging facility
- **diameter-hdd**: Diameter Horizontal Directional Drilling (HDD) Interface facility
- **diameter-svc**: Diameter Service
- **diamproxy**: DiamProxy logging facility
- **dpath**: IPSec Data Path logging facility
- **drvctrl**: Driver Controller logging facility
- **eap-diameter**: Extensible Authentication Protocol (EAP) IP Security facility
- **eap-ipsec**: Extensible Authentication Protocol (EAP) IPSec facility
- **eap-sta-s6a-s13-s6b-diameter**: EAP/STA/S6A/S13/S6B Diameter messages facility
- **ecs-css**: ACSMGR <-> Session Manager Signalling Interface facility
- **egtpc**: eGTP-C logging facility
- **egtpmgr**: enhanced GPRS Tunneling Protocol (eGTP) manager logging facility
- **egtpu**: eGTP-U logging facility
- **embms**: evolved Multimedia Broadcast Multicast Services Gateway facility
- **epdg**: eMBMS Gateway Demux facility
- **event-notif**: Event Notification Interface logging facility
- **evlog**: Event log facility
- **famgr**: Foreign Agent manager logging facility
- **firewall**: Firewall logging facility
- **fng**: Femto Network Gateway (FNG) logging facility
- **gbmgr**: SGSN Gb Interface Manager facility
- **gmm**:
  - For 2.5G: Logs the GPRS Mobility Management (GMM) layer (above LLC layer)
  - For 3G: Logs the access application layer (above the RANAP layer)
- **gprs-app**: GPRS Application logging facility
- **gprs-ns**: GPRS Network Service Protocol (layer between SGSN and the BSS) logging facility
- **gq-rx-tx-diameter**: Gq/Rx/Tx Diameter messages facility
- **gss-gcdr**: GTPP Storage Server GCDR facility
- **gtpc**: GTP-C protocol logging facility
• **gtpcmgr**: GTP-C protocol manager logging facility
• **gtpp**: GTP-prime protocol logging facility
• **gtpu**: GTP-U protocol logging facility
• **gtpumgr**: GTP-U Demux manager
• **gx-ty-diameter**: Gx/Ty Diameter messages facility
• **gy-diameter**: Gy Diameter messages facility
• **h248prt**: H.248 port manager facility
• **hamgr**: Home Agent manager logging facility
• **hat**: High Availability Task (HAT) process facility
• **hdctrl**: HD Controller logging facility
• **henbapp**: Home Evolved NodeB (HENB) App facility (Do not use this keyword for HENB-GW in Release 20)
• **henbgw**: HENB-GW facility (Do not use this keyword for HENB-GW in Release 20)
• **henbgw-pws**: HENB-GW Public Warning System logging facility (Do not use this keyword for HENB-GW in Release 20)
• **henbgw-sctp-acs**: HENB-GW access Stream Control Transmission Protocol (SCTP) facility (Do not use this keyword for HENB-GW in Release 20)
• **henbgw-sctp-nw**: HENBGW network SCTP facility (Do not use this keyword for HNB-GW in Release 20)
• **henbgwdemux**: HENB-GW Demux facility (Do not use this keyword for HNB-GW in Release 20)
• **henbgwmgr**: HENB-GW Manager facility (Do not use this keyword for HNB-GW in Release 20)
• **hnb-gw**: HNB-GW (3G Femto GW) logging facility (Do not use this keyword for HNB-GW in Release 20)
• **hnbmgr**: HNB-GW Demux Manager logging facility (Do not use this keyword for HNB-GW in Release 20)
• **hss-peer-service**: Home Subscriber Server (HSS) Peer Service facility
• **igmp**: Internet Group Management Protocol (IGMP)
• **ikev2**: Internet Key Exchange version 2 (IKEv2)
• **ims-authorizatn**: IP Multimedia Subsystem (IMS) Authorization Service facility
• **ims-sh**: HSS Diameter Sh Interface Service facility
• **imsimgr**: SGSN IMSI Manager facility
• **imsue**: IMS User Equipment (IMSUE) facility
• **ip-arp**: IP Address Resolution Protocol facility
• **ip-interface**: IP interface facility
• **ip-route**: IP route facility
- **ipms**: Intelligent Packet Monitoring System (IPMS) logging facility
- **ipne**: IP Network Enabler (IPNE) facility
- **ipsec**: IP Security logging facility
- **ipsecdemux**: IPSec demux logging facility
- **ips**: IP Service Gateway interface logging facility
- **ipsmgd**: IP Services Gateway facility
- **ipsp**: IP Pool Sharing Protocol logging facility
- **kvstore**: Key/Value Store (KVSTORE) Store facility
- **l2tp-control**: Layer 2 Tunneling Precool (L2TP) control logging facility
- **l2tp-data**: L2TP data logging facility
- **l2tpdemux**: L2TP Demux Manager logging facility
- **l2tpmgr**: L2TP Manager logging facility
- **lagmgr**: Link Aggregation Group (LAG) manager logging facility
- **lcs**: Location Services (LCS) logging facility
- **ldap**: Lightweight Directory Access Protocol (LDAP) messages logging facility
- **li**: Refer to the *Lawful Intercept Interface Reference* for a description of this command.
- **linkmgd**: SGSN/BSS SS7 Link Manager logging facility (2.5G only)
- **llc**: Logical Link Control (LLC) Protocol logging facility; for SGSN: logs the LLC layer between the GMM and the BSSGP layers for logical links between the MS and the SGSN
- **local-policy**: Local Policy Service facility
- **location-service**: Location Services facility
- **m3ap**: M3 Application Protocol facility
- **m3ua**: M3UA Protocol logging facility
- **magmgr**: Mobile Access Gateway manager logging facility
- **map**: Mobile Application Part (MAP) protocol logging facility
- **megadiammgr**: MegaDiameter Manager (SLF Service) logging facility
- **mme-app**: Mobility Management Entity (MME) Application logging facility
- **mme-embms**: MME evolved Multimedia Broadcast Multicast Service facility
- **mme-misc**: MME miscellaneous logging facility
- **mmedemux**: MME Demux Manager logging facility
- **mmemgr**: MME Manager facility
- **mmgr**: Master Manager logging facility
- **mobile-ip**: Mobile IP processes
- **mobile-ip-data**: Mobile IP data facility
• **mobile-ipv6**: Mobile IPv6 logging facility
• **mpls**: Multiprotocol Label Switching (MPLS) protocol logging facility
• **mrme**: Multi Radio Mobility Entity (MRME) logging facility
• **mseg-app**: Mobile Services Edge Gateway (MSEG) application logging facility (This option is not supported in this release.)
• **mseg-gtpc**: MSEG GTP-C application logging facility (This option is not supported in this release.)
• **mseg-gtpu**: MSEG GTP-U application logging facility (This option is not supported in this release.)
• **msegmgr**: MSEG Demux Manager logging facility (This option is not supported in this release.)
• **mtp2**: Message Transfer Part 2 (MTP2) Service logging facility
• **mtp3**: Message Transfer Part 3 (MTP3) Protocol logging facility
• **multicast-proxy**: Multicast Proxy logging facility
• **nas**: Non-Access Stratum (NAS) protocol logging facility [MME 4G]
• **netwstrg**: Network Storage facility
• **npucrl**: Network Processor Unit Control facility
• **npudrv**: Network Processor Unit Driver facility [ASR 5500 only]
• **npumgr**: Network Processor Unit Manager facility
• **npumgr-acl**: NPUMGR ACL logging facility
• **npumgr-drw**: NPUMGR DRV logging facility
• **npumgr-flow**: NPUMGR FLOW logging facility
• **npumgr-fwd**: NPUMGR FWD logging facility
• **npumgr-init**: NPUMGR INIT logging facility
• **npumgr-lc**: NPUMGR LC logging facility
• **npumgr-port**: NPUMGR PORT logging facility
• **npumgr-recovery**: NPUMGR RECOVERY logging facility
• **npumgr-rri**: NPUMGR RRI (Reverse Route Injection) logging facility
• **npumgr-vpn**: NPUMGR VPN logging facility
• **npusim**: NPUSIM logging facility [ASR 5500 only]
• **ntfy-intf**: Notification Interface logging facility [Release 12.0 and earlier versions only]
• **ocsp**: Online Certificate Status Protocol logging facility.
• **orbs**: Object Request Broker System logging facility
• **ospf**: OSPF protocol logging facility
• **ospfv3**: OSPFv3 protocol logging facility
• **p2p**: Peer-to-Peer Detection logging facility
• **pagingmgr**: PAGINGMGR logging facility
• **pccmgr**: Intelligent Policy Control Function (IPCF) Policy Charging and Control (PCC) Manager library

• **pdg**: Packet Data Gateway (PDG) logging facility

• **pdgdmgr**: PDG Demux Manager logging facility

• **pdif**: Packet Data Interworking Function (PDIF) logging facility

• **pgw**: Packet Data Network Gateway (PGW) logging facility

• **pmm-app**: Packet Mobility Management (PMM) application logging facility

• **ppp**: Point-To-Point Protocol (PPP) link and packet facilities

• **pppoe**: PPP over Ethernet logging facility

• **procet-map-frwk**: Proclet mapping framework logging facility

• **push**: VPNMGR CDR push logging facility

• **radius-acct**: RADIUS accounting logging facility

• **radius-auth**: RADIUS authentication logging facility

• **radius-coa**: RADIUS change of authorization and radius disconnect

• **ranap**: Radio Access Network Application Part (RANAP) Protocol facility logging info flow between SGSN and RNS (3G)

• **rct**: Recovery Control Task logging facility

• **rdt**: Redirect Task logging facility

• **resmgr**: Resource Manager logging facility

• **rf-diameter**: Diameter Rf interface messages facility

• **rip**: Routing Information Protocol (RIP) logging facility [RIP is not supported at this time.]

• **rlf**: Rate Limiting Function (RLF) logging facility

• **rohc**: Robust Header Compression (RoHC) facility

• **rsvp**: Reservation Protocol logging facility

• **rua**: RANAP User Adaptation (RUA) [3G Femto GW - RUA messages] logging facility

• **s102**: S102 protocol logging facility

• **s102mgr**: S102Mgr logging facility

• **s1ap**: S1 Application Protocol (S1AP) Protocol logging facility

• **sabp**: Service Area Broadcast Protocol (SABP) logging facility

• **saegw**: System Architecture Evolution (SAE) Gateway facility

• **sbc**: SBC protocol logging facility

• **sccp**: Signalling Connection Control Part (SCCP) Protocol logging (connection-oriented messages between RANAP and TCAP layers).

• **sct**: Shared Configuration Task logging facility

• **sctp**: Stream Control Transmission Protocol (SCTP) Protocol logging facility
• **sef ecs**: Severely Errored Frames (SEF) APIs printing facility
• **sess-gr**: SM GR facility
• **sessctrl**: Session Controller logging facility
• **sessmgr**: Session Manager logging facility
• **sesstrc**: session trace logging facility
• **sft**: Switch Fabric Task logging facility
• **sgs**: SGs interface protocol logging facility
• **sgn-app**: SGSN-APP logging various SGSN "glue" interfaces (for example, between PMM, MAP, GPRS-FSM, SMS).
• **sgn-failures**: SGSN call failures (attach/activate rejects) logging facility (2.5G)
• **sgn-gtpc**: SGSN GTP-C Protocol logging control messages between the SGSN and the GGSN
• **sgn-gtpu**: SGSN GTP-U Protocol logging user data messages between the SGSN and GGSN
• **sgn-mbms-bearer**: SGSN Multimedia Broadcast/Multicast Service (MBMS) Bearer app (SMGR) logging facility
• **sgn-misc**: Used by stack manager to log binding and removing between layers
• **sgn-system**: SGSN System Components logging facility (used infrequently)
• **sgn-test**: SGSN Tests logging facility; used infrequently
• **sgtpcmgr**: SGSN GTP-C Manager logging information exchange through SGTPC and the GGSN
• **sgw**: Serving Gateway facility
• **sh-diameter**: Sh Diameter messages facility
• **sitmain**: System Initialization Task main logging facility
• **sls**: Service Level Specification (SLS) protocol logging facility
• **sm-app**: SM Protocol logging facility
• **sms**: Short Message Service (SMS) logging messages between the MS and the SMSC
• **sndcp**: Sub Network Dependent Convergence Protocol (SNDCP) logging facility
• **snmp**: SNMP logging facility
• **sprmgr**: IPCF Subscriber Policy Register (SPR) manager logging facility
• **srdb**: Static Rating Database
• **srp**: Service Redundancy Protocol (SRP) logging facility
• **sscfnni**: Service-Specific Coordination Function for Signaling at the Network Node Interface (SSCF-NNI) logging facility
• **sscop**: Service-Specific Connection-Oriented Protocol (SSCOP) logging facility
• **ssh-ipsecc**: Secure Shell (SSH) IP Security logging facility
• **ssl**: Secure Socket Layer (SSL) message logging facility
• **stat**: Statistics logging facility
• **supserv**: Supplementary Services logging facility [H.323]
• **system**: System logging facility
• **tacacsplus**: TACACS+ Protocol logging facility
• **tcap**: TCAP Protocol logging facility
• **testctrl**: Test Controller logging facility
• **testmgr**: Test Manager logging facility
• **threshold**: threshold logging facility
• **ttg**: Tunnel Termination Gateway (TTG) logging facility
• **tucul**: TCP/UDP Convergence Layer (TUCL) logging facility
• **udr**: User Data Record (UDR) facility (used with the Charging Service)
• **user-data**: User data logging facility
• **user-l3tunnel**: User Layer 3 tunnel logging facility
• **usertcp-stack**: User TCP Stack
• **vim**: Voice Instant Messaging (VIM) logging facility
• **vinfo**: VINFO logging facility
• **vmgctrl**: Virtual Media Gateway (VMG) controller facility
• **vmgctrl**: VMG Content Manager facility
• **vpn**: Virtual Private Network logging facility
• **wimax-data**: WiMAX DATA
• **wimax-r6**: WiMAX R6
• **wsg**: Wireless Security Gateway (ASR 9000 Security Gateway)
• **x2gw-app**: X2GW (X2 proxy Gateway, eNodeB) application logging facility
• **x2gw-demux**: X2GW demux task logging facility

### Configuring Trace Logging

Trace logging is useful for quickly resolving issues for specific sessions that are currently active. They are temporary filters that are generated based on a qualifier that is independent of the global event log filter configured using the `logging filter` command in the Exec mode. Like event logs, however, the information generated by the logs is stored in the active memory buffer.

All debug level events associated with the selected call are stored.

| **Important** | Trace logs impact session processing. They should be implemented for debug purposes only. |
Use the following example to configure trace logs in the Exec mode:

```
[local] host_name# logging trace { callid call_id | ipaddr ip_address | msid ms_id | username username }
```

Once all of the necessary information has been gathered, the trace log can be deleted by entering the following command:

```
[local] host_name# no logging trace { callid call_id | ipaddr ip_address | msid ms_id | username username }
```

## Configuring Monitor Logs

Monitor logging records all activity associated with all of a particular subscriber's sessions. This functionality is available in compliance with law enforcement agency requirements for monitoring capabilities of particular subscribers.

Monitors can be performed based on a subscriber's MSID or username, and are only intended to be used for finite periods of time as dictated by the law enforcement agency. Therefore, they should be terminated immediately after the required monitoring period.

This section provides instructions for enabling and disabling monitor logs.

### Enabling Monitor Logs

Use the following example to configure monitor log targets:

```
configure
  logging monitor { ip_addr | ipv6_addr | msid id | username name }
end
```

Repeat to configure additional monitor log targets.

### Disabling Monitor Logs

Use the following example to disable monitor logs:

```
configure
  no logging monitor { ip_addr | ipv6_addr | msid id | username name }
end
```

### Viewing Logging Configuration and Statistics

Logging configuration and statistics can be verified by entering the following command from the Exec mode:

```
[local] host_name# show logging [ active | verbose ]
```

When no keyword is specified, the global filter configuration is displayed as well as information about any other type of logging that is enabled.

The following table provides information and descriptions of the statistics that are displayed when the `verbose` keyword is used.
### Table 11: Logging Configuration and Statistics Commands

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General Logging Statistics</strong></td>
<td></td>
</tr>
<tr>
<td>Total events received</td>
<td>Displays the total number of events generated by the system.</td>
</tr>
<tr>
<td>Number of applications receiving</td>
<td>Displays the number of applications receiving the events.</td>
</tr>
<tr>
<td>events</td>
<td></td>
</tr>
<tr>
<td><strong>Logging Source Statistics</strong></td>
<td></td>
</tr>
<tr>
<td>Event sequence ids by process</td>
<td>Displays a list of system processes that have generated events and reference identification number of the event that was generated.</td>
</tr>
<tr>
<td>Msg backlog stat with total cnt</td>
<td>Displays the number of event messages that have been back logged in comparison to the total number of events generated.</td>
</tr>
<tr>
<td>LS L2 filter drop rate</td>
<td>Displays the percentage of logging source (LS) layer 2 (L2) event drops.</td>
</tr>
<tr>
<td>Abnormal Log Source Statistics</td>
<td>Displays abnormal logging source (LS) statistics, if any.</td>
</tr>
<tr>
<td><strong>Runtime Logging Buffer Statistics</strong></td>
<td></td>
</tr>
<tr>
<td>Active buffer</td>
<td>Displays the number of events currently logged in the active memory buffer as well as a date/time timestamp for the oldest and most recent entries in the buffer.</td>
</tr>
<tr>
<td>Inactive buffer</td>
<td>Displays the number of events currently logged in the inactive memory buffer.</td>
</tr>
</tbody>
</table>

---

**Viewing Event Logs Using the CLI**

Event logs generated by the system can be viewed in one of the following ways:

- **From the syslog server**: If the system is configured to send logs to a syslog server, the logs can be viewed directly on the syslog server.

- **From the system CLI**: Logs stored in the system memory buffers can be viewed directly from the CLI.

- **From the console port**: By default, the system automatically displays events over the console interface to a terminal provided that there is no CLI session active.

This section provides instructions for viewing event logs using the CLI. These instructions assume that you are at the root prompt for the Exec mode.

**Step 1**

Copy the active log memory buffer to the inactive log memory buffer. When the active log memory buffer is copied to the inactive log memory buffer existing information in the inactive log memory buffer is deleted.
Both active and inactive event log memory buffers can be viewed using the CLI in Exec mode. However, it is preferable to view the inactive log in order to prevent any data from being over-written. The information from the active log buffer can be copied to the inactive log buffer by entering the following command:

```
[local] host_name# logs checkpoint
```

### Step 2
View the logs by entering the following command:

```
[local] host_name# show logs
```

A number of optional keywords/variables are available for the `show logs` command. Refer to the `Exec Mode Show Commands` chapter in the Command Line Interface Reference for more information.

---

**Configuring and Viewing Crash Logs**

In the unlikely even of a software crash, the system stores information that could be useful in determining the reason for the crash. This information can be maintained in system memory or it can be transferred and stored on a network server.

The system supports the generation of the following two types of logs:

- **Crash log**: Crash logs record all possible information pertaining to a software crash (full core dump). Due to their size, they cannot be stored in system memory. Therefore, these logs are only generated if the system is configured with a Universal Resource Locator (URL) pointing to a local device or a network server where the log can be stored.

- **Abridged crash log**: Crash event records are automatically generated when a software crash occurs and are stored in flash memory on management cards. The abridged crash log contains a list crash event records along with associated dump files. This log allows you to view event records and dump files via CLI commands.

---

**Crash Logging Architecture**

The crash log is a persistent repository of crash event information. Each event is numbered and contains text associated with a CPU (minicore), NPU or kernel crash. The logged events are recorded into fixed length records and stored in `/flash/crashlog2`.

Whenever a crash occurs, the following crash information is stored:

1. The event record is stored in `/flash/crashlog2` file (the crash log).
2. The associated minicore, NPU or kernel dump file is stored in the `/flash/crsh2` directory.
3. A full core dump is stored in a user configured directory.

---

**Important**

The crashlog2 file along with associated minicore, NPU and kernel dumps are automatically synchronized across redundant management cards (SMC, MIO/UMIO). Full core dumps are not synchronized across management cards.

The following behaviors apply to the crash logging process.
• When a crash event arrives on an active management card, the event record is stored in its crashlog2 file along with the minicore, NPU, or kernel dump file in /flash/crsh2. The crash event and dump file are also automatically stored in the same locations on the standby management card.

• When a crash log entry is deleted via CLI command, it is deleted on both the active and standby management cards.

• When a management card is added or replaced, active and standby cards will automatically synchronize crash logs and dump files.

• When a crash event is received and the crash log file is full, the oldest entry in the crash log and its related dump file will be replaced with the latest arrived event and dump file on both management cards. Information for a maximum of 120 crash events can be stored on management cards.

• Duplicate crash events bump the count of hits in the existing record and update the new record with the old crash record. Additions to the count use the timestamp for the first time the event happened.

Configuring Software Crash Log Destinations

The system can be configured to store software crash log information to any of the following locations:

• On the ASR 5000:
  - CompactFlash™: Installed on the SMC [abridged crash log and associated dump files only]
  - PCMCIA Flash Card: Installed in the PCMCIA1 slot on the SMC

• On the ASR 5500:
  - Flash memory: Installed on the active MIO/UMIO [abridged crash log and associated dump files only]
  - USB memory stick: Installed in the USB slot on the active MIO/UMIO

• On VPC
  - Flash memory: Accessible by the virtual machine
  - USB memory stick: Installed in the USB slot of the platform (USB slot has been enabled via the hypervisor)

• Network Server: Any workstation or server on the network that the system can access using the Trivial File Transfer Protocol (TFTP), the File Transfer Protocol (FTP), the Secure File Transfer Protocol (SFTP), or the Hyper-Text Transfer Protocol (HTTP); this is recommended for large network deployments in which multiple systems require the same configuration

  Important
  In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

Crash log files (full core dumps) are written with unique names as they occur to the specified location. The name format is crash-card-cpu-time-core. Where card is the card slot, cpu is the number of the CPU on the card, and time is the Portable Operating System Interface (POSIX) timestamp in hexadecimal notation.
Use the following example to configure a software crash log destination in the Global Configuration mode:

```
configure
crash enable [encrypted] url crash_url
end
```

Notes:

- Refer to the Global Configuration Mode Commands chapter in the Command Line Interface Reference for more information on this command.
- Repeat to configure additional software crash log destinations. There is no limit to the number of destinations that can be configured.

Save the configuration as described in the Verifying and Saving Your Configuration chapter.

## Viewing Abridged Crash Log Information Using the CLI

You can view abridged crash information that is stored as a set of event records in flash memory on management cards (/flash/crashlog2). Each crash event record has an associated dump file (minicore, NPU or kernel) that can also be displayed (/flash/crash2).

Follow the instructions in this section to view software crash events that have occurred on the system. These instructions assume that you are at the root prompt for the Exec mode.

### Step 1

View a list of software crash events by entering the following Exec mode command:

```
[local]host_name# show crash { all | list | number crash_num }
```

Notes:

- Run `show crash list` to obtain the number for a specific crash event.
- Run `show crash number crash_num` to display the output for the target crash event.

The resulting output may not be the same for all platforms:

Information about similar crash events is suppressed in the output of this command.

### Step 2

View the dump file associated with a specific crash event.

The information contained in the dump file helps identify and diagnose any internal or external factors causing the software to crash.

- Crash # – unique number assigned by StarOS when logging the crash event
- SW Version – StarOS build release in format: RR.n(bbbbb)
- Similar Crash Count – number of similar crashes
- Time of first crash – timestamp when first crash occurred in format: YYYY-MMM-DD+hh:mm:ss
- Failure message – text of event message
- Function – code identifier
- Process – where the crash occurred (Card, CPU, PID, etc.)
- Crash time – timestamp for when the crash occurred in the format: YYYY-MMM-DD+hh:mm:ss time zone
Recent errno – text of most recent error number.

Stack – memory stack information

Last Bounce – information about the messaging received prior to the crash

Registers – memory register contents

Current inbound message – hexadecimal information for the current inbound message

Address Map

Recent heap activity (oldest first)

Recent events (oldest first)

Profile depth

The informational content of each crash log entry varies based on the type of crash and the StarOS release.

---

**Checkpointing Logs**

Checkpointing identifies logged data as previously viewed or marked. Checkpointing allows you to only display log information since the last checkpoint.

Individual logs may have up to 50,000 events in the active log. Checkpointing the logs results in at most 50,000 events being in the inactive log files. This gives a maximum of 100,000 events in total which are available for each facility logged.

You check point log data via the Exec mode logs checkpoint command to set the log contents to a well-known point prior to special activities taking place. This command may also be a part of periodic regular maintenance to manage log data.

Checkpointing logs moves the current log data to the inactive logs. Only the most recently check pointed data is retained in the inactive logs. A subsequent check pointing of the logs results in the prior check pointed inactive log data being cleared and replaced with the newly check pointed data. Checkpointed log data is not available for viewing.

---

<table>
<thead>
<tr>
<th>Important</th>
<th>Checkpointing logs should be done periodically to prevent the log files becoming full. Logs which have 50,000 events logged will discard the oldest events first as new events are logged.</th>
</tr>
</thead>
</table>

| Important | An Inspector-level administrative user cannot execute this command. |
Saving Log Files

Log files can be saved to a file in a local or remote location specified by a URL. Use the following Exec mode command to save log files:

```bash
save logs { url } [ active ] [ inactive ] [ callid call_id ] [ event-verbosity evt_verboseness ] [ facility facility ] [ level severity_level ] [ pdu-data pdu_format ] [ pdu-verbosity pdu_verboseness ] [ since from_date_time ] [ until to_date_time ] [ grep grep_options ] [ more ]
```

For detailed information on the `save logs` command, see the *Exec Mode Commands* chapter in the *Command Line Interface Reference*.

Event ID Overview

🇮🇳 Important

The use of event IDs depends on the platform type and the licenses running on the platform.

Identification numbers (IDs) are used to reference events as they occur when logging is enabled on the system. As described previously, logs are collected on a per facility basis. Each facility possesses its own range of event IDs as indicated in the following table.

<table>
<thead>
<tr>
<th>Facility</th>
<th>Description</th>
<th>Event ID Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>a10</td>
<td>A10 Protocol Facility</td>
<td>28000-28999</td>
</tr>
<tr>
<td>a11</td>
<td>A11 Protocol Facility</td>
<td>29000-29999</td>
</tr>
<tr>
<td>a11mgr</td>
<td>A11 Manager Facility</td>
<td>9000-9999</td>
</tr>
<tr>
<td>aaa-client</td>
<td>AAA Client Facility</td>
<td>6000-6999</td>
</tr>
<tr>
<td>aaamgr</td>
<td>AAA Manager Facility</td>
<td>36000-36999</td>
</tr>
<tr>
<td>aaaproxy</td>
<td>AAA Proxy Facility</td>
<td>64000-64999</td>
</tr>
<tr>
<td>aal2</td>
<td>AAL2 Protocol Facility</td>
<td>173200-173299</td>
</tr>
<tr>
<td>acl-log</td>
<td>IP Access Control List (ACL) Facility</td>
<td>21000-21999</td>
</tr>
<tr>
<td>acsctrl</td>
<td>Active Charging Service Controller (ACSCtrl) Facility</td>
<td>90000-90999</td>
</tr>
<tr>
<td>acsmgr</td>
<td>Active Charging Service Manager (ACSMgr) Facility</td>
<td>91000-91999</td>
</tr>
<tr>
<td>afctrl</td>
<td>Ares Fabric Controller (ASR 5500 only)</td>
<td>186000-186999</td>
</tr>
<tr>
<td>afmgr</td>
<td>Ares Fabric Manager (ASR 5500 only)</td>
<td>187000-187999</td>
</tr>
<tr>
<td>alarmctrl</td>
<td>Alarm Controller Facility</td>
<td>65000-65999</td>
</tr>
<tr>
<td>alcap</td>
<td>Access Link Control Application Part (ALCAP) Protocol Facility</td>
<td>160900-161399</td>
</tr>
<tr>
<td>alcapmgr</td>
<td>ALCAP Manager Facility</td>
<td>160500-160899</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>asf</td>
<td>ASF Facility</td>
<td>73000-73999</td>
</tr>
<tr>
<td>asfprt</td>
<td>ASFPRT Facility</td>
<td>59000-59999</td>
</tr>
<tr>
<td>asngwmgr</td>
<td>Access Service Network (ASN) Gateway Manager Facility</td>
<td>100000-100499</td>
</tr>
<tr>
<td>asnpcmgr</td>
<td>ASN Paging/Location-Registry Manager Facility</td>
<td>100500-100999</td>
</tr>
<tr>
<td>bcmcs</td>
<td>Broadcast/Multicast Service (BCMCS) Facility</td>
<td>109000-109999</td>
</tr>
<tr>
<td>bfd</td>
<td>Bidirectional Forwarding Detection (BFD) Protocol Facility</td>
<td>170500-170999</td>
</tr>
<tr>
<td>bgp</td>
<td>Border Gateway Protocol (BGP) Facility</td>
<td>85000-85999</td>
</tr>
<tr>
<td>bindmux</td>
<td>BindMux Manager Facility [Intelligent Policy Control Function (IPCF)]</td>
<td>158200-158999</td>
</tr>
<tr>
<td>bngmgr</td>
<td>Broadband Network Gateway (BNG) Manager Facility</td>
<td>182000-182999</td>
</tr>
<tr>
<td>bssap</td>
<td>Base Station System Application Part+ (BSSAP+) Service Facilities</td>
<td>131000-131199</td>
</tr>
<tr>
<td>bssgp</td>
<td>Base Station System GPRS Protocol (BSSGP) Facility</td>
<td>115050-115099</td>
</tr>
<tr>
<td>callhome</td>
<td>Call Home Facility</td>
<td>173600-173999</td>
</tr>
<tr>
<td>cap</td>
<td>CAMEL Application Part (CAP) Facility</td>
<td>87900-88099</td>
</tr>
<tr>
<td>chatconf</td>
<td>CHATCONF Facility</td>
<td>74000-74999</td>
</tr>
<tr>
<td>cli</td>
<td>Command Line Interface (CLI) Facility</td>
<td>30000-30999</td>
</tr>
<tr>
<td>connproxy</td>
<td>Connection Proxy Facility</td>
<td>190000-190999</td>
</tr>
<tr>
<td>crdt-ctl</td>
<td>Credit Control Facility</td>
<td>127000-127999</td>
</tr>
<tr>
<td>csg</td>
<td>Closed Subscriber Groups (CSG) Facility</td>
<td>188000-188999</td>
</tr>
<tr>
<td>csg-acl</td>
<td>CSG Access Control List (ACL) Facility</td>
<td>189000-189999</td>
</tr>
<tr>
<td>csp</td>
<td>Card/Slot/Port (CSP) Facility</td>
<td>7000-7999</td>
</tr>
<tr>
<td>css</td>
<td>Content Steering Service (CSS) Facility [ESC]</td>
<td>77000-77499</td>
</tr>
<tr>
<td>css-sig</td>
<td>Content Service Selection (CSS) RADIUS Signaling Facility</td>
<td>77500-77599</td>
</tr>
<tr>
<td>cx-diameter</td>
<td>Cx Diameter Message Facility</td>
<td>92840-92849</td>
</tr>
<tr>
<td>dcardctrl</td>
<td>Daughter Card Controller Facility</td>
<td>62000-62999</td>
</tr>
<tr>
<td>dcardmgr</td>
<td>Daughter Card Manager Facility</td>
<td>57000-57999</td>
</tr>
<tr>
<td>demuxmgr</td>
<td>Demux Manager Facility</td>
<td>110000-110999</td>
</tr>
<tr>
<td>dgmbmgr</td>
<td>Diameter Gmb (DGMB) Application Manager Facility</td>
<td>126000-126999</td>
</tr>
<tr>
<td>dhcp</td>
<td>DHCP Facility</td>
<td>53000-53999</td>
</tr>
<tr>
<td>dhcpv6</td>
<td>DHCPv6 Protocol Facility</td>
<td>123000-123999</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>dhost</td>
<td>Distributed Host Manager Facility</td>
<td>83000-83999</td>
</tr>
<tr>
<td>diameter</td>
<td>Diameter Endpoint Facility</td>
<td>92000-92599</td>
</tr>
<tr>
<td>diabase</td>
<td>Diabase Message Facility</td>
<td>92800-92809</td>
</tr>
<tr>
<td>diameter-acct</td>
<td>Diameter Accounting Protocol Facility</td>
<td>112000-112999</td>
</tr>
<tr>
<td>diameter-auth</td>
<td>Diameter Authentication Protocol Facility</td>
<td>111000-111999</td>
</tr>
<tr>
<td>diameter-dns</td>
<td>Diameter DNS Subsystem Facility</td>
<td>92600-92699</td>
</tr>
<tr>
<td>diameter-ecs</td>
<td>ECS Diameter Signaling Facility</td>
<td>81990-81999</td>
</tr>
<tr>
<td>diameter-hdd</td>
<td>Diameter Horizontal Directional Drilling (HDD) Interface Facility</td>
<td>92700-92799</td>
</tr>
<tr>
<td>diameter-svc</td>
<td>Diameter Service Facility</td>
<td>121200-121999</td>
</tr>
<tr>
<td>diamproxy</td>
<td>Diameter Proxy Facility</td>
<td>119000-119999</td>
</tr>
<tr>
<td>dpath</td>
<td>Data Path for IPSec Facility</td>
<td>54000-54999</td>
</tr>
<tr>
<td>drvctrl</td>
<td>Driver Controller Facility</td>
<td>39000-39999</td>
</tr>
<tr>
<td>ds3mgr</td>
<td>DS3 and DS3/E Line Card Manager Facility (part of NPU Manager Controller Facility)</td>
<td>40000-40999</td>
</tr>
<tr>
<td>eap-diameter</td>
<td>Extensible Authentication Protocol (EAP) Diameter Facility</td>
<td>92870-92879</td>
</tr>
<tr>
<td>eap-ipsec</td>
<td>EAP IPSec Facility</td>
<td>118000-118999</td>
</tr>
<tr>
<td>ecs-css</td>
<td>ACS Session Manager (ACSMgr) Signalling Interface Facility</td>
<td>97000-97099</td>
</tr>
<tr>
<td>edr</td>
<td>Event Data Record (EDR) Facility</td>
<td>80000-80999</td>
</tr>
<tr>
<td>egtpc</td>
<td>eGTP-C Facility</td>
<td>141000-141999</td>
</tr>
<tr>
<td>egtpmgr</td>
<td>eGTP Manager Facility</td>
<td>143000-143999</td>
</tr>
<tr>
<td>egtpu</td>
<td>eGTP-U Facility</td>
<td>142000-142999</td>
</tr>
<tr>
<td>epdg</td>
<td>Evolved Packet Data Gateway (ePDG) Facility</td>
<td>178000-178999</td>
</tr>
<tr>
<td>evlog</td>
<td>Event Log Facility</td>
<td>2000-2999</td>
</tr>
<tr>
<td>famgr</td>
<td>Foreign Agent (FA) Manager Facility</td>
<td>33000-33999</td>
</tr>
<tr>
<td>firewall</td>
<td>Firewall Facility</td>
<td>96000-96999</td>
</tr>
<tr>
<td>fng</td>
<td>Femto Network Gateway (FNG) Facility</td>
<td>149000-149999</td>
</tr>
<tr>
<td>gbrmgr</td>
<td>Gb-Manager Facility</td>
<td>201900-202699</td>
</tr>
<tr>
<td>gcdr</td>
<td>GGSN-Charging Data Record (G-CDR) Facility</td>
<td>66000-66999</td>
</tr>
<tr>
<td>gmm</td>
<td>GPRS Mobility Management (GMM) Facility</td>
<td>88100-88299</td>
</tr>
<tr>
<td>gprs-app</td>
<td>General Packet Radio Service (GPRS) Application Facility</td>
<td>115100-115399</td>
</tr>
<tr>
<td>gprs-ns</td>
<td>GPRS-NS Protocol Facility</td>
<td>115000-115049</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>gq-rx-tx-diameter</td>
<td>Gq/Rx/Tx Diameter Messages Facility</td>
<td>92830-92839</td>
</tr>
<tr>
<td>gss-gcdr</td>
<td>GTPP Storage Server GCDR Facility</td>
<td>98000-98099</td>
</tr>
<tr>
<td>gtpc</td>
<td>GTPC Protocol Facility</td>
<td>47000-47999</td>
</tr>
<tr>
<td>gtpcmgr</td>
<td>GTPC Signaling Demultiplexer Manager Facility</td>
<td>46000-46999</td>
</tr>
<tr>
<td>gtpp</td>
<td>GTP-PRIME Protocol Facility</td>
<td>52000-52999</td>
</tr>
<tr>
<td>gtpu</td>
<td>GTPU Protocol Facility</td>
<td>45000-45999</td>
</tr>
<tr>
<td>gtpumgr</td>
<td>GTPU Manager Facility</td>
<td>157200-157999</td>
</tr>
<tr>
<td>gx-ty-diameter</td>
<td>Gx/Ty Diameter Messages Facility</td>
<td>92820-92829</td>
</tr>
<tr>
<td>gy-diameter</td>
<td>Gy Diameter Messages Facility</td>
<td>92810-92819</td>
</tr>
<tr>
<td>h248prt</td>
<td>H.248 Protocol Facility</td>
<td>42000-42999</td>
</tr>
<tr>
<td>hamgr</td>
<td>Home Agent (HA) Manager Facility</td>
<td>34000-34999</td>
</tr>
<tr>
<td>hat</td>
<td>High Availability Task (HAT) Facility</td>
<td>3000-3999</td>
</tr>
<tr>
<td>hddshare</td>
<td>HDD Share Facility</td>
<td>184000-184999</td>
</tr>
<tr>
<td>hnbr-gw</td>
<td>Home eNodeB-GW Facility</td>
<td>195000-195999</td>
</tr>
<tr>
<td>hnbapp</td>
<td>Home eNodeB Application Facility</td>
<td>196000-196999</td>
</tr>
<tr>
<td>hnbgwdemux</td>
<td>Home eNodeB-GW Demux Facility</td>
<td>194000-194999</td>
</tr>
<tr>
<td>hnbwmgmgr</td>
<td>Home eNodeB-GW Manager Facility</td>
<td>193000, 193999</td>
</tr>
<tr>
<td>hnb-gw</td>
<td>Home NodeB (HNB) Gateway Facility</td>
<td>151000-151999</td>
</tr>
<tr>
<td>hnbmgr</td>
<td>HNB Manager Facility</td>
<td>158000-158199</td>
</tr>
<tr>
<td>hss-peer-service</td>
<td>Home Subscriber Server (HSS) Facility [MME]</td>
<td>138000-138999</td>
</tr>
<tr>
<td>igmp</td>
<td>Internet Group Management Protocol (IGMP) Facility</td>
<td>113000-113999</td>
</tr>
<tr>
<td>ikev2</td>
<td>IKEv2 Facility</td>
<td>122000-122999</td>
</tr>
<tr>
<td>ims-authoriztn</td>
<td>IMS Authorization Service Library Facility</td>
<td>98100-98999</td>
</tr>
<tr>
<td>ims-sh</td>
<td>IMS SH Library Facility</td>
<td>124000-124999</td>
</tr>
<tr>
<td>imsimgr</td>
<td>International Mobile Subscriber Identity (IMSI) Manager Facility</td>
<td>114000-114999</td>
</tr>
<tr>
<td>imsus</td>
<td>IMS User Equipment (IMSUE) Facility</td>
<td>144000-145999</td>
</tr>
<tr>
<td>ip-arp</td>
<td>IP Address Resolution Protocol (ARP) Facility</td>
<td>19000-19999</td>
</tr>
<tr>
<td>ip-interface</td>
<td>IP Interface Facility</td>
<td>18000-18999</td>
</tr>
<tr>
<td>ip-route</td>
<td>IP Route Facility</td>
<td>20000-20999</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>ipms</td>
<td>Intelligent Packet Monitoring System (IPMS) Facility</td>
<td>134000-134999</td>
</tr>
<tr>
<td>ipne</td>
<td>IP Network Enabler (IPNE) Facility</td>
<td>192000-192999</td>
</tr>
<tr>
<td>ipsec</td>
<td>IPSec Protocol Facility</td>
<td>55000-56998</td>
</tr>
<tr>
<td>ipsg</td>
<td>IP Services Gateway (IPSG) Facility</td>
<td>128000-128999</td>
</tr>
<tr>
<td>ipsmgr</td>
<td>IPSG Manager (IPSGMgr) Facility</td>
<td>99000-99999</td>
</tr>
<tr>
<td>ipsp</td>
<td>IP Pool Sharing Protocol (IPSP) Facility</td>
<td>68000-68999</td>
</tr>
<tr>
<td>kvstore</td>
<td>Key/Value Store (KVSTORE) Facility</td>
<td>125000-125999</td>
</tr>
<tr>
<td>l2tp-control</td>
<td>L2TP Control PDU Protocol Facility</td>
<td>50000-50999</td>
</tr>
<tr>
<td>l2tp-data</td>
<td>L2TP Data PDU Protocol Facility</td>
<td>49000-49999</td>
</tr>
<tr>
<td>l2tpdemux</td>
<td>L2TP Demux Facility</td>
<td>63000-63999</td>
</tr>
<tr>
<td>l2tpmgr</td>
<td>L2TP Manager Facility</td>
<td>48000-48999</td>
</tr>
<tr>
<td>lagmgr</td>
<td>Link Aggregation Group (LAG) Manager Facility</td>
<td>179000-179999</td>
</tr>
<tr>
<td>ldap</td>
<td>Lightweight Directory Access Protocol (LDAP) Request Facility</td>
<td>160000-160499</td>
</tr>
<tr>
<td>li</td>
<td>Lawful Intercept (LI) Log Facility</td>
<td>69000-69999</td>
</tr>
<tr>
<td>linkmgr</td>
<td>Link Manager Facility</td>
<td>89500-89999</td>
</tr>
<tr>
<td>llc</td>
<td>Logical Link-Control (LLC) Layer Facility (GPRS)</td>
<td>115700-115799</td>
</tr>
<tr>
<td>local-policy</td>
<td>Local Policy Configuration Facility</td>
<td>161400-162399</td>
</tr>
<tr>
<td>m3ap</td>
<td>M3 Application Protocol (M3AP) Facility</td>
<td>211500-211999</td>
</tr>
<tr>
<td>m3ua</td>
<td>MTP Level 3 (M3UA) Protocol Facility [SIGTRAN]</td>
<td>87500-87699</td>
</tr>
<tr>
<td>magmgr</td>
<td>Mobile Access Gateway (MAG) Manager Facility</td>
<td>137500-137999</td>
</tr>
<tr>
<td>map</td>
<td>Mobile Application Part (MAP) Protocol Facility [SS7]</td>
<td>87100-87299</td>
</tr>
<tr>
<td>megadiammgr</td>
<td>MegaDiameter Manager Facility</td>
<td>121000-121199</td>
</tr>
<tr>
<td>mme-app</td>
<td>Mobility Management Entity (MME) Application Facility</td>
<td>147000-147999</td>
</tr>
<tr>
<td>mme-embms</td>
<td>MME evolved Multimedia Broadcast Multicast Service (eMBMS) Facility</td>
<td>212000-212499</td>
</tr>
<tr>
<td>mme-misc</td>
<td>MME Miscellaneous Facility</td>
<td>155800-156199</td>
</tr>
<tr>
<td>mmedemux</td>
<td>MME Demux Manager Facility</td>
<td>154000-154999</td>
</tr>
<tr>
<td>mmemgr</td>
<td>MME Manager Facility</td>
<td>137000-137499</td>
</tr>
<tr>
<td>mmgr</td>
<td>Master Manager (MMGR) Facility</td>
<td>86000-86399</td>
</tr>
<tr>
<td>mobile-ip</td>
<td>Mobile IP (MIP) Protocol Facility</td>
<td>26000-26999</td>
</tr>
<tr>
<td>mobile-ip-data</td>
<td>MIP Tunneled Data Facility</td>
<td>27000-27999</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------------------</td>
</tr>
<tr>
<td>mobile-ipv6</td>
<td>Mobile IPv6 Facility</td>
<td>129000-129999</td>
</tr>
<tr>
<td>mpls</td>
<td>Multiprotocol Label Switching (MPLS) Facility</td>
<td>163500-163999</td>
</tr>
<tr>
<td>mseg-app</td>
<td>Mobile Services Edge Gateway (MSEG) Application Facility</td>
<td>172300-172999</td>
</tr>
<tr>
<td>mseg-gtpc</td>
<td>MSEG GTPC Application Facility</td>
<td>172000-172199</td>
</tr>
<tr>
<td>mseg-gtpu</td>
<td>MSEG GTPU Application Facility</td>
<td>172200-172299</td>
</tr>
<tr>
<td>msegmgr</td>
<td>MSEG Manager Facility</td>
<td>171000-171999</td>
</tr>
<tr>
<td>mtp2</td>
<td>Message Transfer Part 2 (MTP2) Service Facility [SS7]</td>
<td>116900-116999</td>
</tr>
<tr>
<td>mtp3</td>
<td>Message Transfer Part 3 (MTP3) Service Facility [SS7]</td>
<td>115600-115699</td>
</tr>
<tr>
<td>multicast-proxy</td>
<td>Multicast Proxy Facility</td>
<td>94000-94999</td>
</tr>
<tr>
<td>nas</td>
<td>Network Access Signaling (NAS) Facility</td>
<td>153000-153999</td>
</tr>
<tr>
<td>netwstrg</td>
<td>Network Storage Facility</td>
<td>78000-78999</td>
</tr>
<tr>
<td>npuctrl</td>
<td>Network Processing Unit (NPU) Control Facility</td>
<td>16000-16999</td>
</tr>
<tr>
<td>npudrv</td>
<td>NPU Driver Facility</td>
<td>191000-191999</td>
</tr>
<tr>
<td>npumgr</td>
<td>NPU Manager (NPUMGR) Facility</td>
<td>17000-17999</td>
</tr>
<tr>
<td>npumgr-acl</td>
<td>NPUMGR ACL Facility</td>
<td>169000-169999</td>
</tr>
<tr>
<td>npumgr-drv</td>
<td>NPUMGR Driver Facility</td>
<td>185000-185999</td>
</tr>
<tr>
<td>npumgr-flow</td>
<td>NPUMGR Flow Facility</td>
<td>167000-167999</td>
</tr>
<tr>
<td>npumgr-fwd</td>
<td>NPUMGR Forwarding Facility</td>
<td>168000-168999</td>
</tr>
<tr>
<td>npumgr-init</td>
<td>NPUMGR Initialization Facility</td>
<td>164000-164999</td>
</tr>
<tr>
<td>npumgr-lc</td>
<td>NPUMGR LC Facility</td>
<td>180000-180999</td>
</tr>
<tr>
<td>npumgr-port</td>
<td>NPUMGR Port Facility</td>
<td>166000-166999</td>
</tr>
<tr>
<td>npumgr-recovery</td>
<td>NPUMGR Recovery Facility</td>
<td>165000-165999</td>
</tr>
<tr>
<td>npumgr-vpn</td>
<td>NPUMGR VPN Facility</td>
<td>181000-181999</td>
</tr>
<tr>
<td>npusim</td>
<td>NPUSIM Facility</td>
<td>176000-176999</td>
</tr>
<tr>
<td>ntfy-intf</td>
<td>Event Notification Interface Facility</td>
<td>170000-170499</td>
</tr>
<tr>
<td>orbs</td>
<td>Object Request Broker (ORB) System Facility</td>
<td>15000-15999</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>ospf</td>
<td>Open Shortest Path First (OSPF) Protocol Facility</td>
<td>38000-38999</td>
</tr>
<tr>
<td>p2p</td>
<td>Peer-to-Peer (P2P) Facility</td>
<td>146000-146999</td>
</tr>
<tr>
<td>pccmgr</td>
<td>Policy Charging and Control (PCC) Manager Facility</td>
<td>159000-159499</td>
</tr>
<tr>
<td>pdg</td>
<td>Packet Data Gateway (PDG) Facility</td>
<td>152010-152999</td>
</tr>
<tr>
<td>pdgdmg</td>
<td>PDG TCP Demux Manager (pdgdmg) Facility (this is a customer-specific facility)</td>
<td>162400-162999</td>
</tr>
<tr>
<td>pdif</td>
<td>Packet Data Interworking Function (PDIF) Facility</td>
<td>120000-120999</td>
</tr>
<tr>
<td>pgw</td>
<td>Packet Data Network Gateway (PGW) Facility</td>
<td>139000-139999</td>
</tr>
<tr>
<td>pmm-app</td>
<td>Packet Mobility Management (PMM) Application Facility [SGSN]</td>
<td>89200-89499</td>
</tr>
<tr>
<td>ppp</td>
<td>Point-To-Point Protocol (PPP) Facility</td>
<td>25000-25999</td>
</tr>
<tr>
<td>pppoe</td>
<td>Point-to-Point Protocol over Ethernet (PPPoE) Facility</td>
<td>183000-183999</td>
</tr>
<tr>
<td>ptt</td>
<td>PTT Facility</td>
<td>76000-76999</td>
</tr>
<tr>
<td>push</td>
<td>PUSH (VPNMgr CDR Push) Facility</td>
<td>133000-133999</td>
</tr>
<tr>
<td>radius-acct</td>
<td>RADIUS Accounting Protocol Facility</td>
<td>24000-24999</td>
</tr>
<tr>
<td>radius-auth</td>
<td>RADIUS Authentication Protocol Facility</td>
<td>23000-23999</td>
</tr>
<tr>
<td>radius-coa</td>
<td>RADIUS Change of Authorization (CoA) and Disconnect Facility</td>
<td>70000-70999</td>
</tr>
<tr>
<td>ranap</td>
<td>Radio Access Network Application Part (RANAP) Facility</td>
<td>87700-87899</td>
</tr>
<tr>
<td>rct</td>
<td>Recovery Control Task (RCT) Facility</td>
<td>13000-13999</td>
</tr>
<tr>
<td>rdt</td>
<td>Redirector Task (RDT) Facility</td>
<td>67000-67999</td>
</tr>
<tr>
<td>resmgr</td>
<td>Resource Manager (RM) Facility</td>
<td>14000-14999</td>
</tr>
<tr>
<td>rf-diameter</td>
<td>Rf Diameter Messages Facility</td>
<td>92860-92869</td>
</tr>
<tr>
<td>rip</td>
<td>Routing Information Protocol (RIP) Facility</td>
<td>35000-35999</td>
</tr>
<tr>
<td>rohc</td>
<td>Robust Header Compression (ROHC) Protocol Facility</td>
<td>103000-103999</td>
</tr>
<tr>
<td>rsvp</td>
<td>RSVP Protocol Facility</td>
<td>93000-93999</td>
</tr>
<tr>
<td>rua</td>
<td>RANAP User Adaptation (RUA) Protocol Facility</td>
<td>152000-152099</td>
</tr>
<tr>
<td>s1ap</td>
<td>S1 Application Protocol (S1AP) Facility</td>
<td>155200-155799</td>
</tr>
<tr>
<td>saegw</td>
<td>System Architecture Evolution Gateway Facility</td>
<td>191000-191999</td>
</tr>
<tr>
<td>Facility</td>
<td>Description</td>
<td>Event ID Range</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------</td>
<td>-------------------</td>
</tr>
<tr>
<td>sct</td>
<td>Shared Configuration Task (SCT) Facility</td>
<td>32000-32099</td>
</tr>
<tr>
<td>sctp</td>
<td>Stream Control Transmission Protocol (SCTP) Protocol Facility</td>
<td>87300-87499</td>
</tr>
<tr>
<td>sess-gr</td>
<td>SESS-GR Facility</td>
<td>77600-77999</td>
</tr>
<tr>
<td>sessctrl</td>
<td>Session Controller Facility</td>
<td>8000-8999</td>
</tr>
<tr>
<td>sessmgr</td>
<td>Session Manager Facility</td>
<td>10000-12999</td>
</tr>
<tr>
<td>sesstrc</td>
<td>Session Trace Facility</td>
<td>155000-155199</td>
</tr>
<tr>
<td>sft</td>
<td>Switch Fabric Task (SFT) Facility</td>
<td>58000-58999</td>
</tr>
<tr>
<td>sgs</td>
<td>SGS Interface Protocol Facility [MME]</td>
<td>173000-173199</td>
</tr>
<tr>
<td>sgsn-app</td>
<td>SGSN Application Interface Facility</td>
<td>115900-115999</td>
</tr>
<tr>
<td>sgsn-failures</td>
<td>SGSN Call Failures Facility</td>
<td>89100-89199</td>
</tr>
<tr>
<td>sgsn-gtpc</td>
<td>SGSN GTP-C Protocol Facility</td>
<td>116000-116599</td>
</tr>
<tr>
<td>sgsn-gtpu</td>
<td>SGSN GTP-U Protocol Facility</td>
<td>86900-87099</td>
</tr>
<tr>
<td>sgsn-mbms-bearer</td>
<td>SGSN MBMS Bearer Application (SMGR) Facility</td>
<td>116600-116799</td>
</tr>
<tr>
<td>sgsn-misc</td>
<td>SGSN Miscellaneous Facility</td>
<td>88800-89099</td>
</tr>
<tr>
<td>sgsn-system</td>
<td>SGSN System Components Facility</td>
<td>86400-86499</td>
</tr>
<tr>
<td>sgsn-test</td>
<td>SGSN Tests Facility</td>
<td>88700-88799</td>
</tr>
<tr>
<td>sgsn2</td>
<td>SGSN2 Facility</td>
<td>114000-117999</td>
</tr>
<tr>
<td>sgtpcmgr</td>
<td>SGSN GTP-C (SGTPC) Manager Facility</td>
<td>117000-117999</td>
</tr>
<tr>
<td>sgw</td>
<td>Serving Gateway (SGW) Facility</td>
<td>140000-140999</td>
</tr>
<tr>
<td>sh-diameter</td>
<td>Sh Diameter Messages Facility</td>
<td>92850-92859</td>
</tr>
<tr>
<td>sipcdprt</td>
<td>SIPCDPRT Facility</td>
<td>95000-95999</td>
</tr>
<tr>
<td>sitmain</td>
<td>System Initiation Task (SIT) Main Facility</td>
<td>4000-4999</td>
</tr>
<tr>
<td>sm-app</td>
<td>Short Message Service (SMS) Facility</td>
<td>88300-88499</td>
</tr>
<tr>
<td>sms</td>
<td>SMS Service Facility</td>
<td>116800-116899</td>
</tr>
<tr>
<td>sndcp</td>
<td>Sub Network Dependent Convergence Protocol (SNDCP) Facility</td>
<td>115800-115899</td>
</tr>
<tr>
<td>snmp</td>
<td>Simple Network Management Protocol (SNMP) Facility</td>
<td>22000-22999</td>
</tr>
<tr>
<td>sprmgr</td>
<td>Subscriber Policy Register (SPR) Manager Facility</td>
<td>159500-159999</td>
</tr>
<tr>
<td>srdb</td>
<td>Static Rating Database Facility</td>
<td>102000-102999</td>
</tr>
<tr>
<td>srp</td>
<td>Service Redundancy Protocol (SRP) Facility</td>
<td>84000-84999</td>
</tr>
<tr>
<td>sscfnni</td>
<td>SSCFNNI Protocol Facility [ATM]</td>
<td>115500-115599</td>
</tr>
</tbody>
</table>
### Event Severities

The system provides the flexibility to configure the level of information that is displayed when logging is enabled. The following levels are supported:

- **critical**: Logs only those events indicating a serious error has occurred that is causing the system or a system component to cease functioning. This is the highest severity level.

<table>
<thead>
<tr>
<th>Facility</th>
<th>Description</th>
<th>Event ID Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>sscop</td>
<td>SSCOP Protocol Facility [ATM]</td>
<td>115400-115499</td>
</tr>
<tr>
<td>ssh-ipsec</td>
<td>SSH IP Security Facility</td>
<td>56999-56999</td>
</tr>
<tr>
<td>ssl</td>
<td>SSL Facility (this is a customer-specific facility)</td>
<td>156200-157199</td>
</tr>
<tr>
<td>stat</td>
<td>Statistics Facility</td>
<td>31000-31999</td>
</tr>
<tr>
<td>system</td>
<td>System Facility</td>
<td>1000-1999</td>
</tr>
<tr>
<td>tacacs+</td>
<td>TACACS+ Protocol Facility</td>
<td>37000-37999</td>
</tr>
<tr>
<td>taclp</td>
<td>TACLCP Facility</td>
<td>44000-44999</td>
</tr>
<tr>
<td>tcap</td>
<td>Transaction Capabilities Application Part (TCAP) Protocol</td>
<td>86500-86699</td>
</tr>
<tr>
<td>testctrl</td>
<td>Test Controller Facility</td>
<td>174000-174999</td>
</tr>
<tr>
<td>testmgr</td>
<td>Test Manager Facility</td>
<td>175000-175999</td>
</tr>
<tr>
<td>threshold</td>
<td>Threshold Facility</td>
<td>61000-61999</td>
</tr>
<tr>
<td>ttg</td>
<td>Tunnel Termination Gateway (TTG) Facility</td>
<td>130000-130999</td>
</tr>
<tr>
<td>tucl</td>
<td>TCP/UDP Convergence Layer (TUCL) Facility [SS7]</td>
<td>88500-88699</td>
</tr>
<tr>
<td>udr</td>
<td>User Data Record (UDR) Facility</td>
<td>79000-79999</td>
</tr>
<tr>
<td>user-data</td>
<td>User-Data Facility</td>
<td>51000-51999</td>
</tr>
<tr>
<td>user-l3tunnel</td>
<td>User L3 Tunnel Facility</td>
<td>75000-75999</td>
</tr>
<tr>
<td>usertcp-stack</td>
<td>User TCP Stack Facility</td>
<td>173300-173499</td>
</tr>
<tr>
<td>vim</td>
<td>Voice Instant Message (VIM) Facility</td>
<td>60000, 60999</td>
</tr>
<tr>
<td>vinfo</td>
<td>VINFO Facility</td>
<td>82000, 82999</td>
</tr>
<tr>
<td>vmgetrl</td>
<td>Virtual Media Gateway (VMG) Controller Facility</td>
<td>41000, 41999</td>
</tr>
<tr>
<td>vmgctxmgmr</td>
<td>VMG Context Manager Facility</td>
<td>43000, 43999</td>
</tr>
<tr>
<td>vpn</td>
<td>Virtual Private Network (VPN) Facility</td>
<td>5000-5999</td>
</tr>
<tr>
<td>wimax-data</td>
<td>WiMAX DATA Facility</td>
<td>104900-104999</td>
</tr>
<tr>
<td>wimax-r6</td>
<td>WiMAX R6 Protocol (Signaling) Facility</td>
<td>104000-104899</td>
</tr>
</tbody>
</table>
- **error**: Logs events that indicate an error has occurred that is causing the system or a system component to operate in a degraded state. This level also logs events with a higher severity level.

- **warning**: Logs events that may indicate a potential problem. This level also logs events with a higher severity level.

- **unusual**: Logs events that are very unusual and may need to be investigated. This level also logs events with a higher severity level.

- **info**: Logs informational events and events with a higher severity level.

- **trace**: Logs events useful for tracing and events with a higher severity level.

- **debug**: Logs all events regardless of the severity.

Each of the above levels correspond to the "severity" level of the event ID. Therefore, only those event IDs with a "severity" level equal to the logging level are displayed.

### Understanding Event ID Information in Logged Output

This section explains the event information that is displayed when logging is enabled.

The following displays a sample output for an event that was logged.

```
2011-Dec-11+5:18:41.993 [cli 30005 info] [8/0/609 cli:8000609 _commands_cli.c:1290] [software internal system] CLI session ended for Security Administrator admin on device /dev/pts/2
```

The following table describes the elements of contained in the sample output.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2011-Dec-11+5:18:41.993</td>
<td>Date/Timestamp indicating when the event was generated</td>
</tr>
<tr>
<td>[cli 30005 info]</td>
<td>Information about the event including:</td>
</tr>
<tr>
<td></td>
<td>• The facility the event belongs to</td>
</tr>
<tr>
<td></td>
<td>• The event ID</td>
</tr>
<tr>
<td></td>
<td>• The event's severity level</td>
</tr>
<tr>
<td>In this example, the event belongs to the CLI facility, has an ID of 3005, and a severity level of &quot;info&quot;.</td>
<td></td>
</tr>
<tr>
<td>[8/0/609 cli:8000609 _commands_cli.c:1290]</td>
<td>Information about the specific CLI instance.</td>
</tr>
<tr>
<td>[software internal system]</td>
<td>Indicates that the event was generated because of system operation.</td>
</tr>
<tr>
<td>CLI session ended for Security Administrator admin on device /dev/pts/2</td>
<td>The event's details. Event details may, or may not include variables that are specific to the occurrence of the event.</td>
</tr>
</tbody>
</table>
Troubleshooting

This chapter provides information and instructions for using the system command line interface (CLI) for troubleshooting any issues that may arise during system operation.

Refer to the ASR 5500 Installation Guide for comprehensive descriptions of the hardware components addressed by these troubleshooting procedures.

- Detecting Faulty Hardware, page 161
- Taking Corrective Action, page 178
- Verifying Network Connectivity, page 181
- Using the System Diagnostic Utilities, page 184
- Generating an SSD, page 187
- Configuring and Using the Support Data Collector, page 188

Detecting Faulty Hardware

When power is applied to the chassis, power is sequentially applied to the Management I/O (MIO/UMIO) cards, Data Processing Cards (DPC/UDPC/DPC2/UDPC2s), Fabric and Storage Cards (FSCs), and System Status Cards (SSCs).

Each PFU and card installed in the system incorporates light emitting diodes (LEDs) that indicate its operating status. This section describes how to use these status LEDs to verify that all of the installed components are functioning properly.

Important

As the system progresses through its boot process, some cards will not exhibit immediate LED activity. Allow several minutes to elapse after a reboot is initiated before checking the LEDs on the various cards to verify that the boot process has successfully completed.
Licensing Issues

The system boot process is governed by StarOS licenses. During the startup process, each card performs a series of Power-On Self Tests (POSTs) to ensure that the hardware is operational. These tests also verify that the card meets all license requirements to operate in this chassis.

Refer to Chassis Universal License Requirements in the ASR 5500 Installation Guide for additional information on the effect licenses and card types have on the boot process.

Using the CLI to View Status LEDs

Status LEDs for all cards can be viewed via the CLI by entering the Exec mode `show leds all` command.

```
[local] host_name# show leds all
```

The following displays a sample of this command's output.

```
Slot 01: Run/Fail: Green | Active: Off | Redundant: Green
Slot 02: Run/Fail: Green | Active: Off | Redundant: Green
Slot 05: Run/Fail: Green | Active: Green | Redundant: Green
Slot 06: Run/Fail: Green | Active: Off | Redundant: Green
Slot 08: Run/Fail: Green | Active: Off | Redundant: Green
Slot 11: Run/Fail: Green | Active: Green | Redundant: Green
Service: Off
Slot 12: Run/Fail: Green | Active: Green | Redundant: Green
Slot 13: Run/Fail: Green | Active: Green | Redundant: Green
Slot 14: Run/Fail: Green | Active: Green | Redundant: Green
Slot 16: Run/Fail: Green | Active: Green | Redundant: Green
Slot 17: Run/Fail: Green | Active: Green | Redundant: Green
```

The status of the two Power Filter Units (PFUs) can be viewed by entering the Exec mode `show power chassis` command.

Checking the LEDs on the PFU

Each PFU has four LEDs along the top edge of its front panel. You must unsnap the top front cover from the chassis to view these LEDs. Each LED is associated with one of the four 48 VDC power feeds connected to the PFU.
Each LED on the PFU should illuminate blue for normal operating conditions.

*Figure 12: PFU LEDs*

The possible states for these LEDs are described in the following table. If the LED is not blue, use the troubleshooting information below to diagnose the problem.

*Table 14: PFU LED States*

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>Power feed is supplying -48VDC to this power plane</td>
<td>None needed.</td>
</tr>
<tr>
<td>None</td>
<td>PFU is not receiving power to one or more of its power planes.</td>
<td>Verify that each circuit breaker is in the ON position. Verify that the RTN and -48VDC lugs are attached properly to the posts on the upper rear of the chassis. Verify that the ground lug is attached properly. Use a voltmeter to verify that the power distribution panel is supplying the correct voltage and sufficient current to the terminals at the rear of the PFU. Check the cables from the power source to the rack for continuity. If a power distribution panel (PDP) is installed between the power distribution frame (PDF) and the chassis, verify that its circuit breakers are set to ON. If a PDP is installed between the PDF and the chassis, check the cables from the PDP to the chassis for continuity. If all of the above suggestions have been verified, then it is likely that the PFU is not functional. Please contact your service representative.</td>
</tr>
</tbody>
</table>
Checking the LEDs on the MIO Card

Each MIO/UMIO is equipped with the following LEDs:

- Run/Fail
- Active
- Redundancy
- Master
- Busy

Figure 13: MIO Card Status LEDs

The possible states for all MIO/UMIO LEDs are described in the sections that follow.

MIO Run/Fail LED States

The MIO/UMIO Run/Fail LED indicates the overall status of the card. This LED should be steady green for normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 15: MIO Run/Fail LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card powered with no errors detected</td>
<td>None needed.</td>
</tr>
</tbody>
</table>
### Troubleshooting Description

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blinking Green</td>
<td>Card is initializing and/or loading software</td>
<td>This is normal operation during boot-up.</td>
</tr>
<tr>
<td>Red</td>
<td>Card powered with error(s) detected</td>
<td>Errors were detected during the Power On Self Tests (POSTs). It is likely that the errors were logged to the system's command line interface during boot.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power</td>
<td>Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking the LEDs on the PFU, on page 162 for troubleshooting information.</td>
</tr>
</tbody>
</table>

Verify that the power source is supplying ample voltage and current to the chassis.
Verify that the card is properly installed per the instructions in the *ASR 5500 Installation Guide*.
If all of the above suggestions have been verified, it is possible that the MIO is not functional. Please contact your service representative.

### MIO Active LED States

The *Active* LED on the MIO/UMIO indicates that the software is loaded on the card and it is ready for operation. For the MIO installed in chassis slot 5, this LED should be green for normal operation. For the MIO installed in slot 6, this LED should be off for normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

**Table 16: Active LED States**

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is active</td>
<td>None needed for the MIO/UMIO in slot 5. If green for the MIO/UMIO in slot 6, verify that the MIO/UMIO in slot 5 is installed and licensed properly according to the instructions in the <em>ASR 5500 Installation Guide</em>.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Tasks or processes being migrated from the active MIO to the standby MIO.</td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of the MIO/UMIO and system software processes.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card has failed.</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to <em>MIO Run/Fail LED States</em>, on page 164 for troubleshooting information.</td>
</tr>
</tbody>
</table>
MIO Redundancy LED States

The *Redundancy* LED on the MIO/UMIO indicates that software is loaded on the card, but it is serving as a redundant component. For the MIO/UMIO installed in slot 6, this LED should be green for normal operation. For the MIO/UMIO installed in slot 8, this LED should be off for normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 17: MIO Redundancy LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is in redundant mode</td>
<td>None needed. If green for the MIO/UMIOs in slot 5 and slot 6, the cards and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ports are fully backed up.</td>
</tr>
<tr>
<td>Amber</td>
<td>Card or port on this card is not backed up by</td>
<td>Check the status of the other MIO/UMIO. If it has failed or one or more of its</td>
</tr>
<tr>
<td></td>
<td>other MIO.</td>
<td>ports are no longer active, the system can continue to function but redundancy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is compromised.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refer to Monitoring the System for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MIO/UMIO and system software processes.</td>
</tr>
<tr>
<td>Blinking Amber</td>
<td>Tasks or processes being migrated from the</td>
<td>Refer to Monitoring the System for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td>active MIO to the standby MIO.</td>
<td>MIO/UMIO and system software processes.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power.</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and</td>
</tr>
<tr>
<td>OR</td>
<td>Card has failed.</td>
<td>POST results are positive. If it is off, refer to MIO Run/Fail LED States, on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>page 164 for troubleshooting information on.</td>
</tr>
</tbody>
</table>

MIO Master LED States

The *Master* LED on the MIO/UMIO indicates whether the card is in Active or Standby mode.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information also provided to diagnose the problem.

Table 18: MIO Master LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>This card is the Active MIO.</td>
<td>None needed.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Tasks or processes being migrated from the active</td>
<td>Refer to Monitoring the System for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td>MIO to the standby MIO.</td>
<td>MIO/UMIO and system software processes.</td>
</tr>
</tbody>
</table>
Verify that the Run/Fail LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to MIO Run/Fail LED States, on page 164 for troubleshooting information. Refer to Monitoring the System for information on determining the status of the MIO/UMIO and system software processes.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>This card is the Standby MIO. OR Card has failed.</td>
<td>None required. Files are being transferred to or accessed from the RAID configuration on the FSCs. Checking the LEDs on the FSC, on page 171</td>
</tr>
</tbody>
</table>

## MIO Busy LED States

The Busy LED on the MIO/UMIO indicates that the card is accessing the RAID solid state drives on the FSCs. This LED is off when no file storage activity is occurring. The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

**Table 19: MIO Busy LED States**

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Files are being transferred to or accessed from the RAID configuration on the FSCs.</td>
<td>None required.</td>
</tr>
<tr>
<td>None</td>
<td>No RAID activity. OR RAID configuration is unavailable.</td>
<td>Checking the LEDs on the FSC, on page 171</td>
</tr>
</tbody>
</table>

## MIO – Interface Link LED States

The Link LED associated with a 1000Base-T (management) or 10 Gigabit Ethernet port on an MIO/UMIO daughter card (subscriber traffic) indicates the status of the network link. This LED should be green for normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

**Table 20: MIO – Interface Link LED States**

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Link is up</td>
<td>None needed. NOTE: This LED will not indicate the presence of a network link until the interface parameters are set during the software configuration process.</td>
</tr>
</tbody>
</table>
Troubleshooting

Description

Verify that the Run/Fail LED is green. If so, the card is receiving power. If it is off, refer to MIO Run/Fail LED States, on page 164 for troubleshooting information.

Verify that the interface is cabled properly.

Verify that the device on which the interface is located is cabled and powered properly.

Nopowertoocard.

OR

Linkisdown.

None

Verifythattheinterfaceiscabledproperly.

Verifythatthedeviceonwhichtheinterfaceislocatediscabledandpoweredproperly.

MIO – Interface Activity LED States

The Activity LED associated with a 1000Base-T (management) or 10 Gigabit Ethernet port on an MIO/UMIO daughter card (subscriber traffic) indicates the presence of traffic on the network link. This LED should be green when data is being transmitted or received over the interface.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 21: MIO – Interface Activity LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>No power to card. OR Link is down.</td>
<td>Verify that the interface is cabled properly.</td>
</tr>
<tr>
<td>Flashing Green</td>
<td>Traffic is present on the link</td>
<td>None needed.</td>
</tr>
<tr>
<td>None</td>
<td>No traffic is present on the link</td>
<td>None needed if there is no activity on the link. Prior to interface configuration, this is normal operation.</td>
</tr>
</tbody>
</table>

Checking the LEDs on the DPC

Each DPC/UDPC or DPC2/UDPC2 is equipped with status LEDs as listed below:

- Run/Fail
- Active
• Redundancy

*Figure 14: DPC Status LEDs*

The possible states for all of the DPC/UDPC or /DPC2/UDPC2 LEDs are described in the sections that follow.

**DPC Run/Fail LED States**

The DPC/UDPC or /DPC2/UDPC2 Run/Fail LED indicates the overall status of the card. This LED should be green for normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

*Table 22: DPC Run/Fail LED States*

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card powered up with no errors detected.</td>
<td>None needed.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Card is initializing and/or loading software.</td>
<td>This is normal operation during boot-up.</td>
</tr>
<tr>
<td>Red</td>
<td>Card powered up with error(s) detected.</td>
<td>Errors were detected during the Power On Self Tests (POSTs). It is likely that the errors were logged to the system's command line interface during boot.</td>
</tr>
</tbody>
</table>
Troubleshooting Description Color

Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking the LEDs on the PFU, on page 162 for troubleshooting information.

Verify that the power source is supplying ample voltage and current to the chassis.

Verify that the card is properly installed and licensed per the instructions in the ASR 5500 Installation Guide.

If all of the above suggestions have been verified, it is possible that the DPC/UDPC or /DPC2/UDPC2 is not functional. Please contact your service representative.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Card is not receiving power.</td>
<td>Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking the LEDs on the PFU, on page 162 for troubleshooting information. Verify that the power source is supplying ample voltage and current to the chassis. Verify that the card is properly installed and licensed per the instructions in the ASR 5500 Installation Guide. If all of the above suggestions have been verified, it is possible that the DPC/UDPC or /DPC2/UDPC2 is not functional. Please contact your service representative.</td>
</tr>
</tbody>
</table>

DPC Active LED States

The *Active* LED on the DDPC/UDPCPC or /DPC2/UDPC2 indicates that the software is loaded on the card and that the card is ready for operation. When the system first boots up, all installed DPC/UDPCs or /DPC2/UDPC2s are booted into standby mode. The system must then be configured as to which DPC/UDPCs or /DPC2/UDPC2s should serve as redundant components (remain in standby mode) and which should function as active components.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 23: DPC Active LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is active.</td>
<td>The first time power is applied to the system, all of the DPC/UDPCs or /DPC2/UDPC2s should be booted into the standby mode. Therefore, this LED should be off.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Tasks or processes are being migrated from an active DPC to a standby DPC.</td>
<td>Verify that the <em>Redundancy</em> LED on a standby DPC/UDPC or /DPC2/UDPC2 is also blinking green. If so, there is an issue with the active DPC/UDPC or /DPC2/UDPC2 and it is transferring its processes. Refer to Monitoring the System for information on determining the status of the DPC/UDPC or /DPC2/UDPC2 and system software processes and functionality.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card is in Standby Mode.</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to DPC Run/Fail LED States, on page 169 for troubleshooting information. Check the state of the <em>Redundancy</em> LED. If it is green, the card is in standby mode. This is normal operation for the initial power-up. If needed, refer to the Configuring DPC Availability section of System Settings for information on making the card active.</td>
</tr>
</tbody>
</table>
DPC Redundancy LED States

The Redundancy LED on the DPC/UDPC or /DPC2/UDPC2 indicates that software is loaded on the card, but it is serving as a standby component. DPC/UDPCs or /DPC2/UDPC2s support n:1 redundancy; the Redundancy LED should be green on only one DPC/UDPC or /DPC2/UDPC2 for normal system operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 24: DPC Redundancy LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is in standby mode.</td>
<td>None needed. There is at least one DPC/UDPC or /DPC2/UDPC2 in Standby mode.</td>
</tr>
<tr>
<td>Amber</td>
<td>Card is not backed up by a standby DPC.</td>
<td>Check the status of the other DPC/UDPCs or /DPC2/UDPC2s. If one DPC/UDPC or /DPC2/UDPC2 has failed or has been removed from the chassis, the system can continue to function but redundancy is compromised. Refer to Monitoring the System for information on determining the status of the DPC/UDPC or /DPC2/UDPC2 and system software processes.</td>
</tr>
<tr>
<td>Blinking Amber</td>
<td>Tasks or processes being migrated from an active DPC to the standby DPC.</td>
<td>Refer to Monitoring the System for information on determining the status of the DPC/UDPC or /DPC2/UDPC2 and system software processes.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card has failed.</td>
<td>Verify that the Run/Fail LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to DPC Run/Fail LED States, on page 169 for troubleshooting information.</td>
</tr>
</tbody>
</table>

Checking the LEDs on the FSC

Each FSC is equipped with the following LEDs as shown in the accompanying figure:

- Run/Fail
- Active
- Redundancy
- Drive 1 Activity
The possible states for all FSC LEDs are described in the sections that follow.

**FSC Run/Fail LED States**

The FSC Run/Fail LED indicates the overall status of the card. This LED should be green for normal operation. The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card powered with no errors detected</td>
<td>None needed.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Card is initializing and/or loading software</td>
<td>This is normal operation during boot-up.</td>
</tr>
<tr>
<td>Red</td>
<td>Card powered with error(s) detected</td>
<td>Errors were detected during the Power On Self Tests (POSTs). It is likely that the errors were logged to the system's command line interface during boot.</td>
</tr>
</tbody>
</table>
Troubleshooting Description

Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking the LEDs on the PFU, on page 162 for troubleshooting information.

Verify that the power source is supplying ample voltage and current to the chassis.

Verify that the card is properly installed per the instructions in the ASR 5500 Installation Guide.

If all of the above suggestions have been verified, it is possible that the FSC is not functional. Please contact your service representative.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Card is not receiving power</td>
<td>Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking the LEDs on the PFU, on page 162 for troubleshooting information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verify that the power source is supplying ample voltage and current to the chassis.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verify that the card is properly installed per the instructions in the ASR 5500 Installation Guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If all of the above suggestions have been verified, it is possible that the FSC is not functional. Please contact your service representative.</td>
</tr>
</tbody>
</table>

FSC Active LED States

The Active LED on the FSC indicates that the software is loaded on the card and that the card is ready for operation. When the system first boots up, all installed FSCs are booted into ready mode.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

Table 26: FSC Active LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is active.</td>
<td>The first time power is applied to the system, all of the FSCs should be booted into the ready mode. Therefore, this LED should be on.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Tasks or processes being migrated from an active FSC to a standby FSC.</td>
<td>Verify that the Redundancy LED on a standby FSC is also blinking green. If so, there is an issue with the active FSC and it is transferring its processes. Refer to Monitoring the System for information on determining the status of the FSC and system software processes and functionality.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card is in Standby Mode.</td>
<td>Verify that the Run/Fail LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to FSC Run/Fail LED States, on page 172 for troubleshooting information. Check the state of the Redundancy LED. If it is green, the card is in standby mode.</td>
</tr>
</tbody>
</table>

FSC Redundancy LED States

The Redundancy LED on the FSC indicates that software is loaded on the card, but it is serving as a redundant component. FSC support n+1 redundancy; the Redundancy LED should be green on only one FSC for normal system operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.
### Table 27: FSC Redundancy LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is in redundant mode</td>
<td>None needed. There is at least one FSC in Standby mode.</td>
</tr>
<tr>
<td>Amber</td>
<td>Card is not backed up by a standby FSC.</td>
<td>Check the status of the other FSCs. If one FSC has failed or has been removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>from the chassis, the system can continue to function but redundancy is</td>
</tr>
<tr>
<td></td>
<td></td>
<td>compromised.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the FSC and system software processes.</td>
</tr>
<tr>
<td>Blinking Amber</td>
<td>Tasks or processes being migrated from an active FSC to the standby FSC.</td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the FSC and system software processes.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card has failed.</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>POST results are positive. If it is off, refer to <em>FSC Run/Fail LED States</em>, on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>page 172 for troubleshooting information.</td>
</tr>
</tbody>
</table>

### FSC Drive n Activity LED States

The *Drive 1 Activity* and *Drive 2 Activity* LEDs on the FSC indicate that the RAID solid state drives (SSDs) are being accessed by the MIO. Drive 1 and Drive 2 on each FSC form a RAID 0 configuration.

**Important**

The FSC-400GB is equipped with a single 400 GB drive. Only the *Drive 1 Activity* LED will be active; the *Drive 2 Activity* LED will always be off (None).

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information also provided to diagnose the problem.

### Table 28: FSC Driven Activity LED States

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Files are being transferred to or accessed from the RAID configuration by the MIO.</td>
<td>None required.</td>
</tr>
<tr>
<td>None</td>
<td>There is no RAID activity. OR RAID configuration is unavailable.</td>
<td>Checking the LEDs on the MIO Card, on page 164 FSC-400GB is not equipped with a second SDD. Only the <em>Drive 1 Activity</em> LED will be active.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and POST results are positive. If it is off, refer to <em>FSC Run/Fail LED States</em>, on page 172 for troubleshooting information.</td>
</tr>
</tbody>
</table>
Checking the LEDs on the SSC

Each SSC is equipped with the following LEDs as shown in the accompanying figure:

- Run/Fail
- Active
- Redundancy
- System Status
- System Service

**Figure 16: SSC Status LEDs**

The possible states for all SSC LEDs are described in the sections that follow.

**SSC Run/Fail LED States**

The SSC Run/Fail LED indicates the overall status of the card. This LED should be green for normal operation. The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.
**Table 29: SSC Run/Fail LED States**

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card powered with no errors detected</td>
<td>None needed.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Card is initializing and/or loading software</td>
<td>This is normal operation during boot-up.</td>
</tr>
<tr>
<td>Red</td>
<td>Card powered with error(s) detected</td>
<td>Errors were detected during the Power On Self Tests (POSTs). It is likely that</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the errors were logged to the system's command line interface during boot.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power</td>
<td>Verify that the LEDs on the PFUs are blue. If they are not, refer to Checking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the LEDs on the PFU, on page 162 for troubleshooting information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verify that the power source is supplying ample voltage and current to the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>chassis.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Verify that the card is properly installed per the instructions in the ASR 5500</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Installation Guide.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If all of the above suggestions have been verified, it is possible that the SSC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>is not functional. Please contact your service representative.</td>
</tr>
</tbody>
</table>

**SSC Active LED States**

The *Active* LED on the SSC indicates that the software is loaded on the card and that the card is ready for operation. When the system first boots up, both SSCs are booted into ready mode.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

**Table 30: SSC Active LED States**

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is active.</td>
<td>The first time power is applied to the system, both SSCs should be booted into</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the ready mode. Therefore, this LED should be on.</td>
</tr>
<tr>
<td>Blinking Green</td>
<td>Tasks or processes being migrated from an active FSC to a standby FSC.</td>
<td>Verify that the Redundancy LED on a Istanbul SSC is also blinking green. If so,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>there is an issue with the active SSC and it is transferring its processes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refer to Monitoring the System for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSC and system software processes and functionality.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card is in Standby Mode.</td>
<td>Verify that the Run/Fail LED is green. If so, the card is receiving power and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>POST results are positive. If it is off, refer to the SSC Run/Fail LED States</td>
</tr>
<tr>
<td></td>
<td></td>
<td>section for troubleshooting information.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Check the state of the Redundancy LED. If it is green, the card is in standby</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mode.</td>
</tr>
</tbody>
</table>
**SSC Redundancy LED States**

The *Redundancy* LED on the SSC indicates that software is loaded on the card, but it is serving as a standby component. SSC support 1:1 redundancy; the *Redundancy* LED should be green on the other SSC for normal system operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

*Table 31: SSC Redundancy LED States*

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>Card is in standby mode</td>
<td>None needed. The other SSC should be in Standby mode.</td>
</tr>
<tr>
<td>Amber</td>
<td>Card is not backed up by the standby SSC.</td>
<td>Check the status of the other SSC. If one it has failed or has been removed from</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the chassis, the system can continue to function but redundancy is compromised.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SSC and system software processes.</td>
</tr>
<tr>
<td>Blinking Amber</td>
<td>Tasks or processes being migrated from the active SSC to the standby SSC.</td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of the</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power. OR Card has failed.</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>POST results are positive. If it is off, refer to the <em>SSC Run/Fail LED States</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>section for troubleshooting information.</td>
</tr>
</tbody>
</table>

**SSC System Status LED States**

The *System Status* LED on the SSC indicates that there is a loss of service somewhere in the system. If this LED is red, the system requires maintenance or service (for example, the system could not locate a valid software image at boot-up, or a high temperature condition exists).

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information also provided to diagnose the problem.

*Table 32: SSC System Status LED States 11*

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Green</td>
<td>System is operating normally</td>
<td>None required.</td>
</tr>
<tr>
<td>Red</td>
<td>The system has experienced a loss of service.</td>
<td>Refer to <em>Monitoring the System</em> for information on determining the status of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>system hardware and software processes.</td>
</tr>
<tr>
<td>None</td>
<td>Card is not receiving power</td>
<td>Verify that the <em>Run/Fail</em> LED is green. If so, the card is receiving power and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>POST results are positive. If it is off, refer to the <em>SSC Run/Fail LED States</em></td>
</tr>
<tr>
<td></td>
<td></td>
<td>section for troubleshooting information.</td>
</tr>
</tbody>
</table>
SSC System Service LED States

The System Service LED on the SSC illuminates amber to indicate that the system has experienced a hardware component failure.

This LED is off during normal operation.

The possible states for this LED are described in the following table. If the LED is not green, use the troubleshooting information in the table to diagnose the problem.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amber</td>
<td>System requires maintenance (fan filter, temperature warning, PFU outage etc.).</td>
<td>Monitoring the System for show commands, the outputs of which will assist in further determining the problem. Refer to System Logs for information on how to view logs.</td>
</tr>
<tr>
<td>None</td>
<td>No component failures have been detected. OR Card is not receiving power.</td>
<td>No maintenance needed.</td>
</tr>
</tbody>
</table>

Testing System Alarm Outputs

The system provides the following two physical alarm mechanisms:

- **System Audible Alarm**: Located on the SSC, the speaker is used to provide an audible indicator that a minor, major, or critical alarm has occurred.

- **CO Alarms Interface**: Located on the SSC, this interface provides a DB-15 connector that enables three dry-contact relays (Form C) for the triggering of external audio and/or visual indicators. These indicators can be used to alert that either a minor, major, or critical alarm has occurred.

The operation of these alarms can be tested by issuing the following command:

```
[local] host_name# test alarm { audible | central-office | critical | major | minor }
```

When this command is executed, the specified alarm is activated for a period of 10 seconds. After this time, the alarm will return to its previous condition.

Taking Corrective Action

In the event that an issue was discovered with an installed application or line card, depending on the severity, it may be necessary to take corrective action.

The system provides several redundancy and fail-over mechanisms to address issues with application and line cards in order to minimize system downtime and data loss. These mechanisms are described in the sections that follow.
Switching MIOs

When the system boots up, the MIO/UMIO installed in chassis slot 5 will boot into the Active mode and begin booting other system components. The MIO/UMIO installed in chassis slot 6 will automatically be booted into Standby mode dictating that it will serve as a redundant component. The active MIO/UMIO automatically synchronizes currently running tasks or processes with the standby MIO/UMIO.

In the event of a critical failure on the MIO/UMIO in slot 5, system control will be automatically switched to the standby MIO/UMIO in slot 6. This is a relatively seamless transition because the two are synchronized. The formerly active MIO will then enter the standby mode allowing it to be safely replaced or restored.

In the event that an issue arises that is not severe enough for the system to perform an automatic switchover, a manual switchover can be invoked by executing the following commands from the Exec mode prompt.

---

**Step 1**

Initiate a manual MIO/UMIO switch over by entering the following command:

```
[local]\host_name# card switch from <5 or 6> to <6 or 5>
```

You will receive the following prompt:

```
Are You Sure? [Yes|No]:
```

**Step 2**

Press **Y** to start the switchover.

**Step 3**

Verify that the switchover was successful by entering the `show card table` command at the Exec mode prompt:

Check the entry in the **Oper State** column next to the MIO/UMIO just switched. Its state should be **Standby**.

---

Busying Out a DPC

This **busy-out** command moves processes from the source DPC/UDPC or DPC2/UDPC2 to the destination DPC/UDPC or DPC2/UDPC2, or disables the DPC/UDPC or DPC2/UDPC2 from accepting any new calls. When busy-out is enabled, the DPC/UDPC or DPC2/UDPC2 stops receiving new calls but continues to process calls until they are completed. The command prompt is returned once the command is initiated. The busy-out procedure is completed in background.

---

**Step 1**

Initiate a busy-out by entering the following command:

```
[local]\host_name# card busy-out slot_number
```

You will receive the following prompt:

```
Are You Sure? [Yes|No]:
```

**Step 2**

Press **Y** to start the switchover.

**Step 3**

Verify that the busy-out was successful by entering the `show card table` command at the Exec mode prompt:

Check the entry in the **Oper State** column next to the DPC/UDPC or DPC2/UDPC2 just busied-out. Its state should be **Standby**.
Migrating a DPC

When the system boots up, all DPC/UDPCs or DPC2/UDPC2s enter the "standby" mode. The standby mode indicates that the card is available for use but is not configured for operation. Installed components can be made active through the software configuration process. Cards that are not configured to enter the "active" mode will remain in standby mode for use as redundant components.

In the event of the critical failure of a DPC/UDPC or DPC2/UDPC2, tasks will be automatically be migrated from the active card to a redundant card in standby mode.

In the event that an issue arises that is not severe enough for the system to perform an automatic migration, a manual migration can be initiated. Follow the instructions below to manually initiate a DPC/UDPC or DPC2/UDPC2 migration. These instructions assume you are at the root prompt for the Exec mode.

Step 1
Initiate a DPC/UDPC or DPC2/UDPC2 migration by entering the following command:

```
[local]host_name> card migration from original_slot# to final_slot#
```

You will receive the following prompt:

```
Are You Sure? [Yes|No]:
```

Step 2
Press Y to start the migration.

Step 3
Verify that the migration was successful by entering the `show card table` command at the Exec mode prompt. Check the entry in the `Oper State` column next to the packet processing card that was just migrated from. Its state should be `Standby`. The state of the packet processing card migrated to should be `Active`.

Halting Cards

Cards other than MIO/UMIOs that are in either the Active or Standby modes can be halted. Halting these cards places them into the "offline" mode. In this mode, the card is unusable for session processing as either an active or redundant component.

If a card in the active mode is halted, its tasks, processes, or network connections will be migrated or switched to a redundant component prior to entering the offline mode.

This section describes how to initiate a card halt and restore halted components.

Initiate a Card Halt

Important

Do not initiate a `card halt` for an active FSC if there are less than two active FSCs in the system. The system returns an error message if there are less than two active FSCs. There are similar restrictions when executing the `card reboot` or `card upgrade` commands on active FSCs. Refer to the `Command Line Interface Reference` for detailed information.
Follow the instructions below to manually initiate a card halt. These instructions assume you are at the root prompt for the Exec mode.

**Step 1**
Initiate a manual card migration by entering the following command:

```
[local] host_name# card halt slot#
```

`slot#` is the chassis slot number in which the card to be halted is installed. It can be any integer from 1 through 4, and 7 through 18. You will receive the following prompt:

Are You Sure? [Yes|No]:

**Step 2**
Press Y to initiate the halt operation.

**Step 3**
Verify that the migration was successful by entering the `show card table` command at the Exec mode prompt.
Check the entry in the `Oper State` column next to the card that was just halted. Its state should be `Offline`. If the card was in active mode prior to the execution of this command, the state of the redundant component associated with it should now be `Active`.

---

**Restore a Previously Halted Card**

Follow the instructions below to restore a card that was previously halted. These instructions assume you are at the root prompt for the Exec mode.

**Step 1**
Reboot the card to be restored by entering the following command.

```
[local] host_name# card reboot slot# -force
```

You will receive the following prompt:

Are You Sure? [Yes|No]:

**Step 2**
Press Y to start the reboot of the card.

**Step 3**
Verify that the migration was successful by entering the `show card table` command at the prompt.
Check the entry in the `Oper State` column next to the card that was just restored. Its state should be the state of that it was in before it was halted.

---

**Verifying Network Connectivity**

There are multiple commands supported by the system to verify and/or troubleshoot network connectivity. Note that network connectivity can only be tested once system interfaces and ports have been configured and bound.

The commands specified in this section should be issued on a context-by-context basis. Contexts act like virtual private networks (VPNs) that operate independently of other contexts. Ports, interfaces, and routes configured in one context cannot be tested from another context without additional configuration.
To switch between contexts enter the following command at the root prompt for the Exec mode:

```
[local] host_name context context_name
```

`context_name` is the name of the context to which you wish to switch. The following prompt appears:

```
[context_name] host_name#
```

### Using the ping or ping6 Command

The `ping` or `ping6` command verifies the system's ability to communicate with a remote node in the network by passing data packets between and measuring the response. This command is useful in verifying network routing and if a remote node is able to respond at the IP layer.

#### Syntax

The `ping` command has the following syntax:

```
ping host_ipv4_address [count num_packets] [flood] [pattern packet_pattern] [size octet_count] [src { src_host_name | src_host_ipv4_address }] [vrf vrf_name]
```

```
ping6 host_ipv6_address [count num_packets] [flood] [pattern packet_pattern] [size octet_count] [src { src_host_name | src_host_ipv6_address }] [vrf vrf_name]
```

For complete information on the above commands, see the Exec Mode Commands chapter of the Command Line Interface Reference.

The following displays a sample of a successful `ping` (IPV4) response.

```
PING 192.168.250.1 (192.168.250.1): 56 data bytes
64 bytes from 192.168.250.1: icmp_seq=0 ttl=255 time=0.4 ms
64 bytes from 192.168.250.1: icmp_seq=1 ttl=255 time=0.2 ms
64 bytes from 192.168.250.1: icmp_seq=2 ttl=255 time=0.2 ms
64 bytes from 192.168.250.1: icmp_seq=3 ttl=255 time=0.2 ms
64 bytes from 192.168.250.1: icmp_seq=4 ttl=255 time=0.2 ms
--- 192.168.250.1 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 0.2/0.2/0.4 ms
```

### Troubleshooting

If no response is received from the target follow these troubleshooting procedures:

- Verify that the correct IP address was entered.
- Attempt to ping a different device on the same network. If the ping was successful then it is likely that your system configuration is correct. Verify that the device you are attempting to ping is powered and functioning properly.
- Verify the port is operational.
- Verify that the configuration of the ports and interfaces within the context are correct.
- If the configuration is correct and you have access to the device that you're attempting to ping, ping the system from that device.
- If there is still no response, it is likely that the packets are getting discarded by a network device. Use the `traceroute` or `traceroute6` and `show ip static-route` commands discussed in this chapter to further troubleshoot the issue.
Using the traceroute or traceroute6 Command

The `traceroute` or `traceroute6` command collects information on the route data that will take to a specified host. This is a useful troubleshooting command that can be used to identify the source of significant packet delays or packet loss on the network. This command can also be used to identify bottlenecks in the routing of data over the network.

**traceroute – IPv4**

The `traceroute` command has the following syntax:

```
traceroute { host_name | host_ipv4_address } [ count packets ] [ df ] [ maxttl max_ttl ] [ minttl min_ttl ] [ port port_number ] [ size octet_count ] [ src { src_host_name | src_host_ipv4_address } ] [ timeout seconds ] [ vrf vrf_nam ]
```

For complete information on the above command, see the `Exec Mode Commands` chapter of the Command Line Interface Reference.

The following displays a sample output.

```
traceroute to 192.168.250.1 (192.168.250.1), 30 hops max, 40 byte packets
 1 192.168.250.1 (192.168.250.1) 0.446 ms 0.235 ms 0.178 ms
```

**traceroute6 – IPv6**

The `traceroute6` command has the following syntax:

```
traceroute6 { host_name | host_ipv6_address } [ count packets ] [ maxttl max_ttl ] [ port port_number ] [ size octet_count ] [ src { src_host_name | src_host_ipv6_address } ] [ timeout seconds ] [ vrf vrf_nam ]
```

For complete information on the above commands, see the `Exec Mode Commands` chapter of the Command Line Interface Reference.

The following displays a sample output.

```
traceroute6 to 2001:4A2B::1f3F (2001:4A2B::1f3F), 30 hops max, 40 byte packets
 1 2001:4A2B::1f3F (2001:4A2B::1f3F) 0.446 ms 0.235 ms 0.178 ms
```

**Viewing IP Routes**

The system provides a mechanism for viewing route information to a specific node or for an entire context. This information can be used to verify network connectivity and to ensure the efficiency of the network connection. The command has the following syntax:

```
show ip route [ route_ip_address ]
show ipv6 route [ route_ipv6_address ]
```

For complete information on the above commands, see the `Exec Mode show Commands` chapter of the Command Line Interface Reference.

If no keywords are specified, all IP routes within the context's routing table are displayed.

The following displays a sample of this command's output showing a context IPv4 routing table.

```
"*" indicates the Best or Used route.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Nexthop</th>
<th>Protocol</th>
<th>Prec</th>
<th>Cost</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>*0.0.0.0/0</td>
<td>10.0.4.1</td>
<td>static</td>
<td>0</td>
<td>0</td>
<td>SPIOL</td>
</tr>
</tbody>
</table>
```
Viewing the Address Resolution Protocol Table

The system provides a mechanism for viewing Address Resolution Protocol (ARP) table information to a specific node or for an entire context. This information can be used to verify that when the system sends an ARP packet, it receives valid responses from other network nodes.

```
[local]host_name# show ip arp [ arp_ip_address ]
```

*arp_ip_address* specifies a specific network node for which to display ARP information. The address can be entered in IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation. If this keyword is not specified, all entries within the context's ARP table are displayed.

---

**Important**

Restarting the VPN Manager removes all interfaces from the kernel which in turn removes all ARP entries. However, the NPU still retains all of the ARP entries so that there is no traffic disruption. From a user point of view, *show ip arp* is broken since this command gathers information from the kernel and not the NPU.

The following displays a sample of this command's output showing a context's ARP table.

<table>
<thead>
<tr>
<th>Address</th>
<th>Link Type</th>
<th>Link Address</th>
<th>Flags</th>
<th>Mask</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.4.240</td>
<td>ether</td>
<td>00:05:47:02:20:20</td>
<td>C</td>
<td></td>
<td>MIO1</td>
</tr>
<tr>
<td>10.0.4.7</td>
<td>ether</td>
<td>00:05:47:02:03:36</td>
<td>C</td>
<td></td>
<td>MIO1</td>
</tr>
<tr>
<td>10.0.4.1</td>
<td>ether</td>
<td>00:01:30:F2:7F:00</td>
<td>C</td>
<td></td>
<td>MIO1</td>
</tr>
</tbody>
</table>

---

Using the System Diagnostic Utilities

The system provides protocol monitor and test utilities that are useful when troubleshooting or verifying configurations. The information generated by these utilities can help identify the root cause of a software or network configuration issue.

This section describes how to use these utilities.

---

**Important**

Only an administrator with Operator or higher privilege can run the diagnostic utilities described in this section.

---

Using the Monitor Utility

For troubleshooting purposes, the system provides a protocol monitoring utility. This tool displays protocol information for a particular subscriber session or for every session being processed.
Caution

The monitor tool may cause session processing delays and/or data loss. Therefore, it should be used only when troubleshooting.

Using the Protocol Monitor

The protocol monitor displays information for every session that is currently being processed. Depending on the number of protocols monitored, and the number of sessions in progress, a significant amount of data is generated. It is highly recommended that logging be enabled on your terminal client in order to capture all of the information that is generated.

Follow the instructions below to invoke and configure the protocol monitoring tool.

Step 1
Invoke the protocol monitor from the Exec mode by entering the `monitor protocol` command.

```
[local] host_name# monitor protocol
```

An output listing all the currently available protocols, each with an assigned number, is displayed.

Step 2
Choose the protocol that you wish to monitor by entering the associated number at the `Select:` prompt. A right arrow (>) appears next to the protocol you selected.

Step 3
Repeat step 2 as needed to choose multiple protocols.

Step 4
Press B to begin the protocol monitor.

```
WARNING!! You have selected options that can DISRUPT USER SERVICE
Existing CALLS MAY BE DROPPED and/or new CALLS MAY FAIL!!
(Under heavy call load, some debugging output may not be displayed)
Proceed? - Select (Y)es or (N)o
```

Step 5
Enter Y to proceed with the monitor or N to go back to the previous menu.

C - Control Events  {ON }
D - Data Events  {ON }
E - EventID Info  {ON }
H - Display ethernet  {ON }
I - Inbound Events  {ON }
O - Outbound Events  {ON }
S - Sender Info  {OFF}
T - Timestamps  {ON }
X - PDU Hexdump  {OFF}
A - PDU Hex/Ascii  {OFF}
+/- Verbosity Level  { 1 }
L - Limit Context  {OFF}
M - Match Newcalls  {ON }
R - RADIUS Dict  {no-override}
G - GTPP Dict  {no-override}
Y - Multi-Call Trace  {OFF}

(Q)uit, <ESC> Prev Menu, <SPACE> Pause, <ENTER> Re-Display Options

Step 6
Configure the amount of information that is displayed by the monitor. To enable or disable options, enter the letter associated with that option (C, D, E, etc.). To increase or decrease the verbosity, use the plus (+) or minus (-) keys. The current state, ON (enabled) or OFF (disabled), is shown to the right of each option.

Step 7
Press the Enter key to refresh the screen and begin monitoring.
The monitor remains active until disabled. To quit the protocol monitor and return to the prompt, press `q`.

**Using the Protocol Monitor for a Specific Subscriber**

The protocol monitor can be used to display information for a specific subscriber session that is currently being processed. Depending on the number of protocols monitored, and the number of sessions in progress, a significant amount of data is generated. It is highly recommended that logging be enabled on your terminal client in order to capture all of the information that is generated.

Follow the instructions in this section to invoke and configure the protocol monitoring tool for a specific subscriber session.

---

### Step 1
To invoke the session-specific protocol monitor from the Exec mode enter the `monitor subscriber` command.

```
[local] host_name# monitorsubscriber { callid | imei | imsi | ipaddr | ipv6addr | msid | msisdn | next-call | pcf | peer-fa | peer-lac | sgsn-address | type | username }
```

### Step 2
Specify the method the monitor should use by entering the appropriate keyword.

### Step 3
Select other options and/or enter the appropriate information for the selected keyword.

If no session matching the specified criteria was being processed when the monitor was invoked, a screen of available monitoring options appears.

### Step 4
Configure the amount of information that is displayed by the monitor. To enable or disable options, enter the letter or 2-digit number associated with that option (C, D, E, 11, 12, etc.). To increase or decrease the verbosity, use the plus (+) or minus (-) keys.

The current state, ON (enabled) or OFF (disabled), is shown to the right of each option.

Option `Y` for performing multi-call traces is only supported for use with the GGSN.

### Step 5
Repeat step 6 as needed to enable or disable multiple protocols.

### Step 6
Press `Enter` to refresh the screen and begin monitoring.

The following displays a portion of a sample of the monitor's output for a subscriber named `user2@aaa`. The default protocols were monitored.

```
---------------------------------------------------------------------------
Incoming Call:  
---------------------------------------------------------------------------
MSID: 0000012345 Callid: 002dc6c2  
Username: user2@aaa SessionType: unknown  
Status: Active Service Name: xxx1  
Src Context: source Dest Context: 
---------------------------------------------------------------------------
<<<<OUTBOUND 10:02:35:415 Eventid:25001(0)  
PPP Tx PDU (9)  
PAP 9: Auth-Ack(1), Msg=
<<<<OUTBOUND 10:02:35:416 Eventid:25001(0)  
PPP Tx PDU (14)  
IPCP 14: Conf-Req(1), IP-Addr=192.168.250.70
```
Generating an SSD

An SSD is an instance of the output when the Exec mode `show support details` command is run. It displays a comprehensive list of system information that is useful for troubleshooting purposes. In most cases, the output of this command is requested by the Technical Assistance Center (TAC).

An SSD output .tar file can redirected to a local or remote location (URL).

The .tar file includes:

- **support_summary** - An ASCII text file that contains the support detail information.
- **information.minicores.tar** - A .tar file that contains any minicore files found on the system. Minicore files contain memory core dumps that are captured during some events. These core dumps provide specific memory locations and other information about the event. This information is useful to the technical support team in identifying where and when an event occurred along with its probably cause.
The `show support details` command includes information that is not otherwise accessible to users but that is helpful in the swift resolution of issues by TAC.

---

**Important** Platforms with large configuration files can take up to 30 minutes to complete an SSD. Executing the `show support details` command consumes system resources and may reduce traffic throughput.

For releases prior to 20.0, an operator could initiate another SSD while an SSD was already running. In a large configuration, the SSD request often timed out while waiting for the first one to complete. The operator was not aware of the failure until the entire timeout expired. An operator could have more than one SSD running simultaneously.

For release 20.0 and higher, if an SSD is in progress when the operator enters the `show support details` command, StarOS responds with a warning message stating that an SSD is already in progress and the user should try again later. The operator is restricted to running only one SSD instance at a time.

For additional information about the `show support details` command, see the Exec Mode show Commands (Q-S) chapter in the Command Line Interface Reference.

---

## Configuring and Using the Support Data Collector

The task of collecting the support data is performed by a background CLI task called the record collector. The administrator configures the Support Data Collector (SDC) via the CLI with the commands to be executed on a periodic basis. The record collector always runs in the background and checks if there are records to be collected.

When it is time to collect support data, the scheduler executes the configured sequence of CLI commands and stores the results in a gunzipped (.gz) file on the hard-disk. This file is called an SDR (Support Data Record), and represents a snapshot of the overall state of the system at that time.

Technical Assistance Center (TAC) personnel and local administrators can review the SDRs on-line or by transferring them off the system. They may also wish to investigate the collector state information.

Refer to the Support Data Collector chapter for a complete description of SDC functionality.
This chapter describes how to recover a system after it has failed to complete a reboot following a power off cycle or interruption of the normal boot sequence following a `reload` command.

**Caution**
This system recovery process interrupts subscriber service by dropping any existing flows and preventing traffic from being processed during the boot interval. It should only be initiated as an emergency measure.

This chapter includes the following sections:

- Prerequisites, page 189
- Accessing the boot CLI, page 190
- Booting from a Selected Image, page 191

**Prerequisites**
Successful recovery from a failed reboot requires that you have access to the system via a console port, and have an uncorrupted copy of the StarOS boot image file stored in flash memory on the management card, or accessible from an external memory device.

**Console Access**
The boot recovery sequence can only be executed via a terminal connected to the serial console port on the active management card. This connection can be through a terminal server that is accessible via a LAN interface.
The boot recovery sequence can only be viewed via the Console port.

**Boot Image**
The boot recovery command line interface allows you to specify from which boot image you would like to boot the system. If the system failed to reload following a software update, you can initiate a boot from a previously stored image.
The system recovery procedure will prompt you to enter the path name for the location of the StarOS boot image from which the system will boot. By default the boot command will timeout and attempt to reload the highest priority image from flash memory using the default configuration file.

The operating system software is delivered as a single binary file (.bin file extension) and is loaded as a single instance for the entire system.

- For StarOS releases prior to 16.1, the image filename is identified by its release version and corresponding build number. Format = production.build_number.platform.bin.
- For StarOS release 16.1 onwards, the image filename is identified by its platform type and release number. Format = platform-release_number.bin.

Refer to the Configuring the Boot Stack section in the Software Management Operations chapter for additional information on boot stack entries and prioritization.

### Accessing the boot CLI

To access the boot CLI you must interrupt an in-progress reload (reboot) sequence.

**Caution**

This system recovery process interrupts subscriber service by dropping any existing flows and preventing traffic from being processed during the boot interval. It should only be initiated as an emergency measure.

### Initiate a Reboot

A reload can be initiated in one of two ways:

- Power cycle the chassis – Turn the circuit breakers on the power filter units (PFUs) Off (0) and then On (I).
- Execute a `reload` command

  `[local]host_name# reload -noconfirm`

The boot sequence displays messages on the terminal as it steps through its processes.

### Interrupt the Boot Sequence

When the "Booting priority" message line appears (and not before), press CTRL+C to break out of the boot process as shown in the example below:

```
Booting priority 8
  image : /flash/image_filename.bin
  config: /flash/system.cfg
Entry at 0x000000000cba45e0
```
Press CTRL+C at this point in the sequence.

A message similar to the following appears after the boot process has been interrupted:

```
*******9/0 Ctrl-C Pressed---------------------------------------------
Failed.
```
Enter CLI Mode

With the boot prompt displayed, enter cli to access the boot recovery CLI. The CLI prompt changes as shown below:

```
8/0:boot> cli
8/0:cli>
```

boot Command Syntax

The boot recovery command has the following syntax:

```
boot [-show | -priority=* | -config=* | -noconfig] {bootfile_URL}
```

The options for this command include:

- `-show`: displays the current boot configuration
- `-priority=*`: selects the desired boot stack priority (*)
- `-config=*`: enters the desired configuration filename (*), if not the default file
- `-noconfig`: boots using no configuration file

`bootfile_URL` is the URL for the location of the StarOS boot image file. It specifies the path and file name of the StarOS .bin file from which the system will be booted.

The URL may refer to a local file (flash) or an external file on a memory device attached to the management card. The URL must be entered in the following format:

```
{/flash | /pcmcia1 | /usb1}/filename
```

Booting from a Selected Image

You will issue a `boot` command via the boot CLI to initiate the system recovery process.

Boot Using No Configuration File

This procedure boots the system using the specified boot image without also loading a configuration file. A sample command string appears below:

```
8/0:cli> boot -noconfig /flash/image_filename.bin
```

The boot sequence ends with a prompt to enter the Quick Setup Wizard for creating a configuration file.

```
Launching StarOS
Starting program at 0x0000000000100000
Starent Networks ASR5x00 Intelligent Mobile Gateway
management_card is starting up..............................
Starting software image_version_number...
```

No configuration found, press enter to continue.

1. Do you wish to continue with the Quick Setup Wizard[yes/no]: 
You can exit the Quick Setup Wizard by entering no in response to the above prompt. Load a desired configuration file using the Exec mode configure command followed by the URL for the configuration file as shown in the example below:

[local]host_name(config)

```
configure /flash/system.cfg
```

**Boot Using A Specified Configuration File**

This procedure boots the system using the specified boot image and configuration file. A sample command string appears below:

```
8/0:cli> boot -config=/flash/system.cfg /flash/image_filename.bin
```

The boot sequence ends with the appearance of the CLI prompt.

[local]host_name#

Confirm that the desired configuration has loaded by running the Exec mode show configuration command.
CHAPTER 14

Access Control Lists

This chapter describes system support for access control lists and explains how they are configured. The product administration guides provide examples and procedures for configuration of basic services on the system. You should select the configuration example that best meets your service model before using the procedures described below.

Important

You do not require a license to configure ACLs. However, the number of ACLs configured may impact performance significantly.

Important

Not all commands and keywords/variables may be available. Availability depends on the platform type.

This chapter contains the following sections:

• Overview, page 193
• Understanding ACLs, page 194
• Configuring ACLs on the System, page 196
• Applying IP ACLs, page 198

Overview

IP access lists, commonly known as access control lists (ACLs), control the flow of packets into and out of the system. They are configured on a per-context basis and consist of "rules" (ACL rules) or filters that control the action taken on packets that match the filter criteria. Once configured, an ACL can be applied to any of the following:

• An individual interface
• All traffic facilitated by a context (known as a policy ACL)
• An individual subscriber
• All subscriber sessions facilitated by a specific context
Separate ACLs may be created for IPv4 and IPv6 access routes.

**Understanding ACLs**

This section discusses the two main aspects to ACLs on the system:

- Rule(s), on page 194
- Rule Order, on page 196

---

Refer to *ACL Configuration Mode Commands* and the *IPv6 ACL Configuration Mode Commands* chapter in the *Command Line Interface Reference* for the full command syntax.

---

**Rule(s)**

A single ACL consists of one or more ACL rules. Each rule is a filter configured to take a specific action when packets matching specific criteria. Up to 128 rules can be configured per ACL.

---

Configured ACLs consisting of no rules imply a "deny any" rule. The **deny** action and **any** criteria are discussed later in this section. This is the default behavior for an empty ACL.

Each rule specifies the action to take when a packet matches the specified criteria. This section discusses the rule actions and criteria supported by the system.

---

**Actions**

ACLs specify that one of the following actions can be taken on a packet that matches the specified criteria:

- **Permit**: The packet is accepted and processed.
- **Deny**: The packet is rejected.
- **Redirect**: The packet is forwarded to the specified next-hop address through a specific system interface or to the specified context for processing.

---

Redirect rules are ignored for ACLs applied to specific subscribers or all subscribers facilitated by a specific context, or APN for UMTS subscribers.

---

**Criteria**

Each ACL consists of one or more rules specifying the criteria that packets will be compared against. The following criteria are supported:
• **Any**: Filters all packets
• **Host**: Filters packets based on the source host IP address
• **ICMP**: Filters Internet Control Message Protocol (ICMP) packets
• **IP**: Filters Internet Protocol (IP) packets
• **Source IP Address**: Filter packets based on one or more source IP addresses
• **TCP**: Filters Transport Control Protocol (TCP) packets
• **UDP**: Filters User Datagram Protocol (UDP) packets

Each of the above criteria are described in detail in the sections that follow.

---

**Important**

The following sections contain basic ACL rule syntax information. Refer to the *ACL Configuration Mode Commands* and *IPv6 ACL Configuration Mode Commands* chapters in the *Command Line Interface Reference* for the full command syntax.

---

• **Any**: The rule applies to all packets.
• **Host**: The rule applies to a specific host as determined by its IP address.
• **ICMP**: The rule applies to specific Internet Control Message Protocol (ICMP) packets, Types, or Codes. ICMP type and code definitions can be found at [www.iana.org](http://www.iana.org) (RFC 3232).
• **IP**: The rule applies to specific Internet Protocol (IP) packets or fragments.
• **IP Packet Size Identification Algorithm**: The rule applies to specific Internet Protocol (IP) packets identification for fragmentation during forwarding.

This configuration is related to the "IP Identification field" assignment algorithm used by the system, when subscriber packets are being encapsulated (such as Mobile IP and other tunneling encapsulation). Within the system, subscriber packet encapsulation is done in a distributed way and a 16-bit IP identification space is divided and distributed to each entity which does the encapsulation, so that unique IP identification value can be assigned for IP headers during encapsulation.

Since this distributed IP Identification space is small, a non-zero unique identification will be assigned only for those packets which may potentially be fragmented during forwarding (since the IP identification field is only used for reassembly of the fragmented packet). The total size of the IP packet is used to determine the possibility of that packet getting fragmented.

• **Source IP Address**: The rule applies to specific packets originating from a specific source address or a group of source addresses.
• **TCP**: The rule applies to any Transport Control Protocol (TCP) traffic and could be filtered on any combination of source/destination IP addresses, a specific port number, or a group of port numbers. TCP port numbers definitions can be found at [www.iana.org](http://www.iana.org)
• **UDP**: The rule applies to any User Datagram Protocol (UDP) traffic and could be filtered on any combination of source/destination IP addresses, a specific port number, or a group of port numbers. UDP port numbers definitions can be found at [www.iana.org](http://www.iana.org).
Rule Order

A single ACL can consist of multiple rules. Each packet is compared against each of the ACL rules, in the order in which they were entered, until a match is found. Once a match is identified, all subsequent rules are ignored.

Additional rules can be added to an existing ACL and properly ordered using either of the following options:

- Before
- After

Using these placement options requires the specification of an existing rule in the ACL and the configuration of the new rule as demonstrated by the following flow:

```
[ before | after ] { existing_rule }
```

## Configuring ACLs on the System

This section describes how to configure ACLs.

### Important

This section provides the minimum instruction set for configuring access control list on the system. For more information on commands that configure additional parameters and options, refer to the ACL Configuration Mode Commands and IPv6 ACL Configuration Mode Commands chapters in the Command Line Interface Reference.

To configure the system to provide an access control list facility to subscribers:

1. **Step 1** Create the access control list by following the example configuration in Creating ACLs, on page 196
2. **Step 2** Specify the rules and criteria for action in the ACL list by following the example configuration in Configuring Action and Criteria for Subscriber Traffic, on page 197
3. **Step 3** **Optional.** The system provides an "undefined" ACL that acts as a default filter for all packets into the context. The default action is to "permit all". Modify the default configuration for "undefined" ACLs for by following the example configuration in Configuring an Undefined ACL, on page 197
4. **Step 4** Verify your ACL configuration by following the steps in Verifying the ACL Configuration, on page 198
5. **Step 5** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the Verifying and Saving Your Configuration chapter.

## Creating ACLs

To create an ACL, enter the following command sequence from the Exec mode of the system CLI:

```
configure
context acl_ctxt_name [-noconfirm]
```
{ ip | ipv6 } access-list acl_list_name
end

Notes:

• The maximum number of ACLs that can be configured per context is limited by the amount of available memory in the VPN Manager software task. Typically, the maximum is less than 200.

Configuring Action and Criteria for Subscriber Traffic

To create rules to deny/permit the subscriber traffic and apply the rules after or before action, enter the following command sequence from the Exec mode of the system CLI:

```bash
configure
  context acl_ctxt_name [-noconfirm]
    { ip | ipv6 } access-list acl_list_name
      deny { ip_address | any | host | icmp | ip | log | tcp | udp }
      permit { ip_address | any | host | icmp | ip | log | tcp | udp }
      after { deny | permit | readdress | redirect }
      before { deny | permit | readdress | redirect }
end
```

Notes:

⚠️ The system does not apply a "deny any" rule, unless it is specified in the ACL. This behavior can be changed by adding a "deny any" rule at the end of the ACL.

• The maximum number of rules that can be configured per ACL varies depending on how the ACL is to be used. For more information, refer to the Engineering Rules chapter.

• Use the information provided in the Actions and Criteria to configure the rules that comprise the ACL. For more information, refer to the ACL Configuration Mode Commands and IPv6 ACL Configuration Mode Commands chapters in the Command Line Interface Reference.

Configuring an Undefined ACL

As discussed previously the system uses an "undefined" ACL mechanism for filtering the packet(s) in the event that an ACL that has been applied is not present. This scenario is likely the result of a mis-configuration such as the ACL name being mis-typed during the configuration process.

For these scenarios, the system provides an "undefined" ACL that acts as a default filter for all packets into the context. The default action is to "permit all".

To modify the default behavior for unidentified ACLs, use the following configuration:

```bash
configure
  context acl_ctxt_name [-noconfirm]
    access-list undefined { deny-all | permit-all }
end
```

Notes:
Verifying the ACL Configuration

To verify the ACL configuration, enter the Exec mode `show { ip | ipv6 } access-list` command. The following is a sample output of this command. In this example, an ACL named `acl_1` was configured.

```
ip access list acl_1
  deny host 10.2.3.4
  deny ip any host 10.2.3.4
  permit any 10.2.4.4
1 ip access-lists are configured.
```

Applying IP ACLs

Once an ACL is configured, it must be applied to take effect.

---

**Important**

All ACLs should be configured and verified according to the instructions in the Configuring ACLs on the System, on page 196 prior to beginning these procedures. The procedures described below also assume that the subscribers have been previously configured.

As discussed earlier, you can apply an ACL to any of the following:

- Applying an ACL to an Individual Interface, on page 200
- Applying an ACL to All Traffic Within a Context, on page 202 (known as a policy ACL)
- Applying an ACL to an Individual Subscriber, on page 204
- Applying a Single ACL to Multiple Subscribers, on page 208
- Applying a Single ACL to Multiple Subscribers, on page 208 (for 3GPP subscribers only)

---

**Important**

ACLs must be configured in the same context in which the subscribers and/or interfaces to which they are to be applied. Similarly, ACLs to be applied to a context must be configured in that context.
If ACLs are applied at multiple levels within a single context (such as an ACL is applied to an interface within the context and another ACL is applied to the entire context), they will be processed as shown in the following figure and table.

**Figure 17: ACL Processing Order**

![ACL Processing Order Diagram]

**Table 34: ACL Processing Order Descriptions**

<table>
<thead>
<tr>
<th>Order</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>An inbound ACL configured for the receiving interface in the Source Context is applied to the tunneled data (such as the outer IP header). The packet is then forwarded to the Destination Context.</td>
</tr>
<tr>
<td>2</td>
<td>An inbound ACL configured for the subscriber (either the specific subscriber or for any subscriber facilitated by the context) is applied.</td>
</tr>
<tr>
<td>3</td>
<td>A context ACL (policy ACL) configured in the Destination Context is applied prior to forwarding.</td>
</tr>
<tr>
<td>4</td>
<td>An outbound ACL configured on the interface in the Destination Context through which the packet is being forwarded, is applied.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Order</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>An inbound ACL configured for the receiving interface configured in the Destination Context is applied.</td>
</tr>
<tr>
<td>2</td>
<td>An outbound ACL configured for the subscriber (either the specific subscriber or for any subscriber facilitated by the context) is applied. The packet is then forwarded to the Source Context.</td>
</tr>
<tr>
<td>3</td>
<td>A context ACL (policy ACL) configured in the Source Context is applied prior to forwarding.</td>
</tr>
<tr>
<td>4</td>
<td>An outbound ACL configured on the interface in the Source Context through which the packet is being forwarded, is applied to the tunneled data (such as the outer IP header).</td>
</tr>
</tbody>
</table>
In the event that an IP ACL is applied that has not been configured (for example, the name of the applied 
ACL was configured incorrectly), the system uses an "undefined" ACL mechanism for filtering the packet(s). 
This section provides information and instructions for applying ACLs and for configuring an "undefined" 
ACL.

### Applying the ACL to an Interface

To apply the ACL to an interface, use the following configuration:

```
configure
  context acl_ctx_name [ -noconfirm ]
    interface interface_name
      { ip | ipv6 } access-group acl_list_name { in | out } [ preference ]
    end
```

**Notes:**

- The context name is the name of the ACL context containing the interface to which the ACL is to be 
  applied.
- The ACL to be applied must be configured in the context specified by this command.
- Up to eight ACLs can be applied to a group provided that the number of rules configured within the 
  ACL(s) does not exceed the 128-rule limit for the interface.

### Applying an ACL to an Individual Interface

This section provides information and instructions for applying one or more ACLs to an individual interface 
configured on the system.

#### Important

This section provides the minimum instruction set for applying the ACL list to an interface on the system. 
For more information on commands that configure additional parameters and options, refer to the Ethernet 
Interface Configuration Mode Commands chapter in the Command Line Interface Reference.

To configure the system to provide ACL facility to subscribers:

---

**Step 1**  
Apply the configured access control list by following the example configuration in Applying the ACL to an Interface, 
on page 200

**Step 2**  
Verify that ACL is applied properly on interface by following the steps in Verifying the ACL Configuration on an 
Interface, on page 201

**Step 3**  
Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode 
**save configuration** command. For additional information refer to the Verifying and Saving Your Configuration chapter.
Verifying the ACL Configuration on an Interface

This section describes how to verify the ACL configuration.

In the Exec Mode, enter the following command:

```
[local]host# show configuration context context_name
```

`context_name` is the name of the context containing the interface to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
  context context_name
    ip access-list acl_name
      deny host ip_address
      deny ip any host ip_address
    exit
  ip access-group access_group_name
    service-redundancy-protocol
    exit
  interface interface_name
    ip address ip_address/mask
    exit
  subscriber default
  exit
  aaa group default
  exit
  gtp group default
end
```

Applying the ACL to a Context

To apply the ACLs to a context, use the following configuration:

```
configure
  context acl_ctxt_name [-noconfirm]
    { ip | ipv6 } access-group acl_list_name | in | out | [ preference ]
end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- The context-level ACL is applied to outgoing packets. This applies to incoming packets also if the flow match criteria fails and forwarded again.

The `in` and `out` keywords are deprecated and are only present for backward compatibility.

Context ACL will be applied in the following cases:
• Outgoing packets to an external source.

• Incoming packets that fail flow match and are forwarded again. In this case, the context ACL applies first and only if it passes are packets forwarded.

  During forwarding, if an ACL rule is added with a destination address as a loopback address, the context ACL is also applied. This is because StarOS handles packets destined to the kernel by going through a forwarding lookup for them. To apply ACL rules to incoming packets, the interface ACL must be used instead of the context ACL.

• The ACL to be applied must be configured in the context specified by this command.

• Up to eight ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128-rule limit for the interface.

### Applying an ACL to All Traffic Within a Context

This section provides information and instructions for applying one or more ACLs to a context configured within a specific context on the system. The applied ACLs, known as policy ACLs, contain rules that apply to all traffic facilitated by the context.

**Important**

This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer to the Context Configuration Mode Commands chapter in the Command Line Interface Reference.

To configure the system to provide access control list facility to subscribers:

**Step 1**  
Apply the configured ACL as described in Applying the ACL to a Context, on page 201

**Step 2**  
Verify that ACL is applied properly on interface as described in Verifying the ACL Configuration in a Context, on page 202

**Step 3**  
Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the Verifying and Saving Your Configuration chapter.

### Verifying the ACL Configuration in a Context

To verify the ACL configuration:

Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

```
[local] host_name# show configuration context context_name
```

`context_name` is the name of the context to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.
Applying an ACL to a RADIUS-based Subscriber

IP ACLs are applied to subscribers via attributes in their profile. The subscriber profile could be configured locally on the system or remotely on a RADIUS server.

To apply an ACL to a RADIUS-based subscriber, use the `Filter-Id` attribute.

For more details on this attribute, if you are using StarOS 12.3 or an earlier release, refer to the *AAA and GTPP Interface Administration and Reference*. If you are using StarOS 14.0 or a later release, refer to the *AAA Interface Administration and Reference*.

This section provides information and instructions for applying an ACL to an individual subscriber whose profile is configured locally on the system.

This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer to the *Subscriber Configuration Mode Commands* chapter in the *Command Line Interface Reference*.

To configure the system to provide access control list facility to subscribers:

**Step 1** Apply the configured access control list by following the example configuration in *Applying an ACL to an Individual Subscriber*, on page 204

**Step 2** Verify that ACL is applied properly on interface by following the steps in *Verifying the ACL Configuration to an Individual Subscriber*, on page 204

**Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the *Verifying and Saving Your Configuration* chapter.
Applying an ACL to an Individual Subscriber

To apply the ACL to an individual subscriber, use the following configuration:

```
configure context acl_ctxt_name [-noconfirm]
    subscriber_name subs_name
        { ip | ipv6 } access-group acl_list_name [ in | out ]
    end
```

Notes:

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- If neither the `in` nor the `out` keyword is specified, the ACL will be applied to all inbound and outbound packets.
- The ACL to be applied must be configured in the context specified by this command.
- Up to eight ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128-rule limit for the interface.

Verifying the ACL Configuration to an Individual Subscriber

These instructions are used to verify the ACL configuration.

Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

```
[local]host_name+ show configuration context context_name
```

`context_name` is the name of the context containing the subscriber `subs1` to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure context context_name
    ip access-list acl_name
        deny host ip_address
        deny ip any host ip_address
    exit
    ip access-group access_group_name
    service-redundancy-protocol
    exit
    interface interface
        ip address ip_address/mask
    exit
    subscriber default
    exit
    subscriber name subscriber_name
```
Applying an ACL to the Subscriber Named default

This section provides information and instructions for applying an ACL to the subscriber named default.

---

**Important**

This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer to `Subscriber Configuration Mode Commands` in the Command Line Interface Reference.

---

To configure the system to provide access control list facility to subscribers:

---

**Step 1**

Apply the configured access control list by following the example configuration in *Applying an ACL to the Subscriber Named default*, on page 205

**Step 2**

Verify that ACL is applied properly on interface by following the steps in *Verifying the ACL Configuration to the Subscriber Named default*, on page 206

**Step 3**

Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the *Verifying and Saving Your Configuration* chapter.

---

**Applying an ACL to the Subscriber Named default**

To apply the ACL to the subscriber named default, use the following configuration:

```plaintext
configure
    context acl_ctxt_name [ -noconfirm ]
    subscriber name subs_name
        [ ip | ipv6 ] access-group acl_list_name [ in | out ]
    end
```

**Notes:**

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
• If neither the **in** nor the **out** keyword is specified, the ACL will be applied to all inbound and outbound packets.

• The ACL to be applied must be configured in the context specified by this command.

• Up to eight ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128-rule limit for the interface.

### Verifying the ACL Configuration to the Subscriber Named default

These instructions are used to verify the ACL configuration.

Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

```bash
[local]host_name+ show configuration context context_name
```

*context_name* is the name of the context containing the subscriber default to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
  context *context_name*
    ip access-list *acl_name*
      deny host *ip_address*
      deny ip any host *ip_address*
    exit
    ip access-group *access_group_name*
    service-redundancy-protocol
    exit
    interface *interface*
      ip address *ip_address/mask*
    exit
    subscriber name default
      ip access-group *access_group_name* in
      ip access-group *access_group_name* out
    exit
    aaa group default
    exit
    gtppp group default
    exit
    content-filtering server-group *cfsg_name*
      response-timeout *response_timeout*
      connection retry-timeout *retry_timeout*
    end
```

### Applying an ACL to Service-specified Default Subscriber

This section provides information and instructions for applying an ACL to the subscriber to be used as the "default" profile by various system services.
This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer to the Subscriber Configuration Mode Commands chapter in the Command Line Interface Reference.

To configure the system to provide access control list facility to subscribers:

**Step 1** Apply the configured access control list by following the example configuration in Applying an ACL to the Subscriber Named default, on page 205.

**Step 2** Verify that the ACL is applied properly on interface by following the steps in Verifying the ACL Configuration to Service-specified Default Subscriber, on page 207.

**Step 3** Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the Verifying and Saving Your Configuration chapter.

### Applying an ACL to Service-specified Default Subscriber

To apply the ACL to a service-specified Default subscriber, use the following configuration:

```
configure
context acl_ctxt_name [-noconfirm ]
    { pdsn-service | fa-service | ha-service } service_name
default subscriber svc_default_subs_name
exit
subscriber name svc_default_subs_name
    { ip | ipv6 } access-group acl_list_name [ in | out ]
end
```

**Notes:**

- The context name is the name of the ACL context containing the interface to which the ACL is to be applied.
- If neither the `in` nor the `out` keyword is specified, the ACL will be applied to all inbound and outbound packets.
- The ACL to be applied must be configured in the context specified by this command.
- Up to eight ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128-rule limit for the interface.

### Verifying the ACL Configuration to Service-specified Default Subscriber

To verify the ACL configuration.

Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

```
[local]host_name# show configuration context context_name
```
context_name is the name of the context containing the service with the default subscriber to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```plaintext
configure
c   context context_name
      ip access-list acl_name
        deny host ip_address
        deny ip any host ip_address
      exit
      ip access-group access_group_name
      interface interface
        ip address ip_address/mask
      exit
    subscriber default
  exit
  subscriber name subscriber_name
    ip access-group access_group_name in
    ip access-group access_group_name out
  exit
pdsn-service service_name
  default subscriber subscriber_name
end
```

---

**Applying a Single ACL to Multiple Subscribers**

As mentioned in the previous section, IP ACLs are applied to subscribers via attributes in their profile. The subscriber profile could be configured locally on the system or remotely on a RADIUS server.

The system provides for the configuration of subscriber functions that serve as default values when specific attributes are not contained in the individual subscriber's profile. The following table describes these functions.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subscriber named default</td>
<td>Within each context, the system creates a subscriber called default. The profile for the subscriber named default provides a configuration template of attribute values for subscribers authenticated in that context. Any subscriber attributes that are not included in a RADIUS-based subscriber profile is configured according to the values for those attributes as defined for the subscriber named default. NOTE: The profile for the subscriber named default is not used to provide missing information for subscribers configured locally.</td>
</tr>
</tbody>
</table>
**default subscriber**

This command in the PDSN, FA, and HA service Configuration modes specifies a profile from a subscriber named something other than *default* to use a configuration template of attribute values for subscribers authenticated in that context.

This command allows multiple services to draw "default" subscriber information from multiple profiles.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>default subscriber</td>
<td>This command in the PDSN, FA, and HA service Configuration modes specifies a profile from a subscriber named something other than <em>default</em> to use a configuration template of attribute values for subscribers authenticated in that context. This command allows multiple services to draw &quot;default&quot; subscriber information from multiple profiles.</td>
</tr>
</tbody>
</table>

When configured properly, the functions described in the table above could be used to apply an ACL to:

- All subscribers facilitated within a specific context by applying the ACL to the profile of the subscriber named *default*.
- All subscribers facilitated by specific services by applying the ACL to a subscriber profile and then using the **default subscriber** command to configure the service to use that subscriber as the "default" profile.

**Applying an ACL to Multiple Subscriber via APNs**

To apply the ACL to multiple subscribers via APN, use the following configuration:

```
configure
context dest_context_name [-noconfirm]
apn apn_name
    { ip | ipv6 } access-group acl_list_name [ in | out ]
end
```

**Notes:**

- The ACL to be applied must be in the destination context of the APN (which can be different from the context where the APN is configured).
- If neither the *in* nor the *out* keyword is specified, the ACL will be applied to all inbound and outbound packets.
- Up to eight ACLs can be applied to a group provided that the number of rules configured within the ACL(s) does not exceed the 128-rule limit for the interface.

**Applying an ACL to Multiple Subscriber via APNs**

If IP ACLs are applied to subscribers via attributes in their profile, the subscriber profile could be configured locally on the system or remotely on a RADIUS server.

To reduce configuration time, ACLs can alternatively be applied to APN templates for GGSN subscribers. When configured, any subscriber packets facilitated by the APN template would then have the associated ACL applied.

This section provides information and instructions for applying an ACL to an APN template.
This section provides the minimum instruction set for applying the ACL list to all traffic within a context. For more information on commands that configure additional parameters and options, refer to the Subscriber Configuration Mode Commands chapter in the Command Line Interface Reference.

To configure the system to provide access control list facility to subscribers:

**Step 1**  
Apply the configured access control list by following the example configuration in Applying an ACL to Multiple Subscriber via APNs, on page 209.

**Step 2**  
Verify that ACL is applied properly on interface by following the steps in Verifying the ACL Configuration to APNs, on page 210.

**Step 3**  
Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode `save configuration` command. For additional information refer to the Verifying and Saving Your Configuration chapter.

---

**Verifying the ACL Configuration to APNs**

To verify the ACL configuration:

Verify that your ACL lists were applied properly by entering the following command in Exec Mode:

```
show configuration context context_name
```

*context_name* is the name of the context containing the APN *apn1* having *default* subscriber to which the ACL(s) was/were applied.

The output of this command displays the configuration of the entire context. Examine the output for the commands pertaining to interface configuration. The commands display the ACL(s) applied using this procedure.

```
configure
context context_name
ip access-list acl_name
  deny host ip_address
  deny ip any host ip_address
  exit
ip access-group access_group_name
interface interface
  ip address ip_address/mask
  exit
subscriber default
  exit
apn apn_name
  ip access-group access_group_name in
  ip access-group access_group_name out
end
```
Overview

Congestion Control monitors the system for conditions that could potentially degrade performance when the system is under heavy load. Typically, these conditions are temporary (for example, high CPU or memory utilization) and are quickly resolved. However, continuous or large numbers of these conditions within a specific time interval may impact the system's ability to service subscriber sessions. Congestion control helps identify such conditions and invokes policies for addressing the situation.

Congestion control operation is based on configuring the following:

• **Congestion Condition Thresholds**: Thresholds dictate the conditions for which congestion control is enabled and establishes limits for defining the state of the system (congested or clear). These thresholds function in a way similar to operation thresholds that are configured for the system as described in the *Thresholding Configuration Guide*. The primary difference is that when congestion thresholds are reached, a service congestion policy and an SNMP trap (starCongestion) are generated.

  A threshold tolerance dictates the percentage under the configured threshold that must be reached in order for the condition to be cleared. An SNMP trap, starCongestionClear, is then triggered.

  • **Port Utilization Thresholds**: If you set a port utilization threshold, when the average utilization of all ports in the system reaches the specified threshold, congestion control is enabled.

  • **Port-specific Thresholds**: If you set port-specific thresholds, when any individual port-specific threshold is reached, congestion control is enabled system-wide.

• **Service Congestion Policies**: Congestion policies are configurable for each service. These policies dictate how services respond when the system detects that a congestion condition threshold has been crossed.
This section provides the minimum instruction set for configuring congestion control. Commands that configure additional interface or port properties are provided in Subscriber Configuration Mode in the Command Line Interface Reference. Always refer to the Administration Guides for all of the licensed products running on this platform for additional configuration information with respect to congestion control. Congestion control functionality varies based on product and StarOS version.

For the MME three levels of congestion control thresholds are supported – critical, major and minor. By default only the critical threshold is supported for other products. SNMP traps also support major and minor congestion control thresholds. A set of congestion-action-profile commands allows an operator to establish additional actions to be taken for specific thresholds and threshold levels.

### Configuring Congestion Control

To configure Congestion Control functionality:

#### Step 1
Configure congestion control thresholds as described in Configuring the Congestion Control Threshold, on page 212

#### Step 2
Configure service congestion policies as described in Configuring Service Congestion Policies, on page 213

#### Step 3
Enable redirect overload policies as described in Enabling Congestion Control Redirect Overload Policy, on page 214

#### Step 4
Configure disconnecting subscribers based on call or inactivity time as described in Disconnecting Subscribers Based on Call or Inactivity Time, on page 214

#### Step 5
Save your configuration as described in the Verifying and Saving Your Configuration chapter.

### Configuring the Congestion Control Threshold

To configure congestion control threshold, apply the following example configuration in the Global Configuration mode of the CLI:

```
configure
  congestion-control threshold max-sessions-per-service-utilization percent
  congestion-control threshold tolerance percent
end
```

Notes:

- There are numerous threshold parameters. See Global Configuration Mode Commands in the Command Line Interface Reference for more information.

- The tolerance is the percentage under a configured threshold that dictates the point at which the condition is cleared.

- Multiple levels of congestion thresholds – critical, major and minor – are supported for various types of congestion control thresholds. If a threshold level is not specified, the default is critical. Currently, major and minor thresholds are only supported for the MME. The congestion-action-profile command under lte-policy defines the action to be taken when thresholds are exceeded. See Global Configuration
Mode Commands, LTE Policy Configuration Mode Commands and Congestion Action Profile Configuration Mode Commands in the Command Line Interface Reference for more information.

• Repeat this configuration as needed for additional thresholds.

**Configuring Service Congestion Policies**

To create a congestion control policy, apply the following example configuration in the Global Configuration mode of the CLI:

```bash
configure
congestion-control policy service action { drop | none | redirect | reject }
end
```

Notes:

• When the redirect action occurs for PDSN services, the PDSN responds to the PCF with a reply code of 136, "unknown PDSN address" along with the IP address of an alternate PDSN.

• **redirect** is not available for PDIF. The default action for PDIF is "none."

• When the redirect action occurs for HA services, the system responds to the FA with a reply code of 136, "unknown home agent address".

• **redirect** cannot be used in conjunction with GGSN services.

• **redirect** is not available for the Local Mobility Anchor (LMA) service.

• When setting the action to reject, the reply code is 130, "insufficient resources".

• For the GGSN, the reply code is 199, "no resources available".

• For the SaMOG, MME, **redirect** is not available.

• For the MME, create action profiles for optional major and minor thresholds using the `congestion-action-profile` command under `lte-policy` in the Global Configuration mode.

• For the MME, you can specify `service` as `critical`, `major` or `minor` to set a policy and associate an action-profile for the respective threshold. See *Global Configuration Mode Commands* in the *Command Line Interface Reference* for more information.

**Configuring Overload Reporting on the MME**

When an overload condition is detected on an MME and the report-overload keyword is enabled in the `congestion-control policy` command, the system reports the condition to a specified percentage of eNodeBs and proceeds to take the configured action on incoming sessions. To create a congestion control policy with overload reporting, apply the following example configuration:

```bash
configure
congestion-control policy mme-service action report-overload reject-new-sessions enodeb-percentage percentage
end
```

Notes:

• Other overload actions include `permit-emergency-sessions` and `reject-non-emergency-sessions`.
Enabling Congestion Control Redirect Overload Policy

To create a congestion control policy and configure a redirect overload policy for the service, apply the following example configuration:

```
configure
congestion-control
  context context_name
    policy overload redirect address
end
```

Notes:

* Optional: If the congestion control policy action was configured to redirect, then a redirect overload policy must be configured for the service(s) that are affected.

* There are several service configuration modes that you can configure. See the Command Line Interface Reference for a complete list of modes.

* You can set various options for redirection. See the Command Line Interface Reference for more information.

* Repeat this configuration example to configure overload policies for additional services configured in the same context.

Verify the Service Overload Policies

To verify that the service overload policies were properly configured enter the following command in the Exec Mode:

```
[local]host_name# show service_type name service_name
```

This command lists the entire service configuration. Verify that the information displayed for the "Overload Policy" is accurate.

Repeat this configuration example to configure additional services in other contexts.

Verify the Congestion Control Configuration

Verify MME Congestion Action Profiles

To verify MME multilevel congestion action profiles, run the following Exec mode command:

```
[local]host_name# show lte-policy congestion-action-profile { name profile_name | summary }
```

Disconnecting Subscribers Based on Call or Inactivity Time

During periods of heavy system load, it may be necessary to disconnect subscribers in order to maintain an acceptable level of system performance. You can establish thresholds to select subscribers to disconnect based on the length of time that a call has been connected or inactive.
To enable overload disconnect for the currently selected subscriber, use the following configuration example:

```
configure
  context context_name
    subscriber name subscriber_name
      default overload-disconnect threshold inactivity-time dur_thresh
      default overload-disconnect threshold connect-time dur_thresh
    end
  end
```

To disable the overload disconnect feature for this subscriber, use the following configuration example:

```
configure
  context context_name
    subscriber subscriber_name
      no overload-disconnect { [threshold inactivity-time] | [threshold connect-time] }
    end
  end
```
Routing

This chapter provides information on configuring an enhanced, or extended, service. The product administration guides provide examples and procedures for configuring basic services on the system. You should select the configuration example that best meets your service model, and configure the required elements for that model before using the procedures described below.

This chapter includes the following sections:

- Routing Policies, page 217
- Static Routing, page 219
- OSPF Routing, page 220
- OSPFv3 Routing, page 223
- Equal Cost Multiple Path (ECMP), page 224
- BGP-4 Routing, page 225
- Bidirectional Forwarding Detection, page 232
- Viewing Routing Information, page 242

Routing Policies

This section describes how to configure the elements needed to define routing policies. Routing policies modify and redirect routes to and from the system to satisfy specific network deployment requirements.

Use the following building blocks to configure routing policies:

- **Route Access Lists** – The basic building block of a routing policy. Route access lists filter routes based on a range of IP addresses.
- **IP Prefix Lists** – A more advanced element of a routing policy. An IP Prefix list filters routes based on IP prefixes.
- **AS Path Access Lists** – A basic building block used for Border Gateway Protocol (BGP) routing. These lists filter Autonomous System (AS) paths.
- **Route Maps** – Route-maps provide detailed control over routes during route selection or route advertisement by a routing protocol, and in route redistribution between routing protocols. For this level
of control you use IP Prefix Lists, Route Access Lists and AS Path Access Lists to specify IP addresses, address ranges, and Autonomous System paths.

Creating IP Prefix Lists

Use the following configuration example to create IP Prefix Lists:

```config
context context_name
  ip prefix-list name list_name { deny | permit } network_address/net_mask
```

Notes:

• Set the IP prefix list to deny, permit or match any prefix.
• IPv4 dotted-decimal and IPv6 colon-separated-hexadecimal addresses are supported.
• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Creating Route Access Lists

Use the following procedure to create a Route Access List:

```config
context context_name
  route-access-list { extended identifier } { deny | permit } [ ip address ip_address ]
  route-access-list named list_name { deny | permit } [ ip_address/mask | any ] | exact-match |
```

Notes:

• A maximum of 64 access lists are supported per context.
• A maximum of 16 entries can defined for each route-access-list.
• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Creating AS Path Access Lists

Use the following procedure to create an AS Path Access List:

```config
context context_name
  ip as-path access-list list_name [ { deny | permit } reg_expr ]
```

Notes:

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.
Creating Route Maps

Use the following configuration example to create a Route Map:

```conf
config
  context context_name
  route-map map_name { deny | permit } seq_number
```

Notes:

- Use the `match` and `set` commands in Route Map Configuration mode to configure the route map. Refer to the Command Line Interface Reference for more information on these commands.
- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Sample Configuration

The example below shows a configuration that creates two route access lists, applies them to a route map, and uses that route map for a BGP router neighbor.

```conf
config
  context isp1
    route-access-list named RACLin1a permit 88.151.1.0/30
    route-access-list named RACLin1a permit 88.151.1.4/30
    route-access-list named RACLAny permit any
    route-map RMnet1 deny 100
      match ip address route-access-list RACLin1a
    #exit
    route-map RMnet1 deny 200
      match ip address route-access-list RACLin1b
    #exit
    route-map RMnet1 permit 1000
      match ip address route-access-list RACLAny
    #exit
    router bgp 1
      neighbor 152.20.1.99 as-path 101
      neighbor 152.20.1.99 route-map RMnet1
```

Static Routing

The system supports static network route configuration on a per context basis. Define network routes by specifying the:

- IP address and mask for the route
- Name of the interface in the current context that the route must use
- Next hop IP address

```Important```
On the ASR 5000, static routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.
Adding Static Routes to a Context

To add static routes to a context configuration, you must know the names of the interfaces that are configured in the current context. Use the `show ip interface` command to list the interfaces in the current context (Exec mode).

Information for all interfaces configured in the current context is displayed as shown in the following example.

```
[local] host_name show ip interface
Intf Name: Egress 1
Description: IP State: Up (Bound to slot/port untagged ifIndex 402718721)
IP Address: 192.168.231.5
Subnet Mask: 255.255.255.0
Bcast Address: 192.168.231.255
MTU: 1500
Resoln Type: ARP ARP timeout: 3600 secs
L3 monitor LC-port switchover: Disabled
Number of Secondary Addresses: 0
Total interface count: 1
```

The first line of information for each interface lists the interface name for the current context as shown in the example output. In this example, there is one interface with the name `Egress 1`.

```
config context context_name
  ip route { ip_address [ ip_mask ] | ip_addr_mask_combo } { next-hop next_hop_address | egress_name [ precedence precedence [ cost cost ] ]
```

Notes:

- You can configure a maximum of 1,200 static routes per context. Save your configuration as described in the `Verifying and Saving Your Configuration` chapter.

Deleting Static Routes From a Context

Use the following configuration example to remove static routes from a context's configuration:

```
config context context_name
  no ip route { ip_address ip_mask | ip_addr_mask_combo } next_hop_address egress_name [ precedence precedence [ cost cost ]
```

Notes

- Save your configuration as described in the `Verifying and Saving Your Configuration` chapter.

OSPF Routing

This section gives an overview of Open Shortest Path First (OSPF) routing and its implementation in the system. It also describes how to enable the base OSPF functionality and lists the commands that are available for more complex configurations.

You must purchase and install a license key before you can use this feature. Contact your Cisco account representative for more information on licenses.
Important During system task recovery, it is possible for a dynamically-learned forwarding entry to incorrectly remain in the system forwarding table if that forwarding entry has been removed from the dynamic routing protocol during the recovery.

Important On the ASR 5000, OPSF routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

**OSPF Version 2 Overview**

OSPF is a link-state routing protocol that employs an interior gateway protocol (IGP) to route IP packets using the shortest path first based solely on the destination IP address in the IP packet header. OSPF routed IP packets are not encapsulated in any additional protocol headers as they transit the network.

An Autonomous System (AS), or Domain, is defined as a group of networks within a common routing infrastructure.

OSPF is a dynamic routing protocol that quickly detects topological changes in the AS (such as router interface failures) and calculates new loop-free routes after a period of convergence. This period of convergence is short and involves a minimum of routing traffic.

In a link-state routing protocol, each router maintains a database, referred to as the link-state database, that describes the Autonomous System's topology. Each participating router has an identical database. Each entry in this database is a particular router's local state (for example, the router's usable interfaces and reachable neighbors). The router distributes its local state throughout the AS by flooding.

All routers run the same algorithm in parallel. From the link-state database, each router constructs a tree of shortest paths with itself as root to each destination in the AS. Externally derived routing information appears on the tree as leaves. The cost of a route is described by a single dimensionless metric.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of this area is hidden from the rest of the AS, which enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection from bad routing data. An area is a generalization of an IP subnetted network.

OSPF enables the flexible configuration of IP subnets so that each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different sizes (that is, different masks). This is commonly referred to as variable-length subnetting. A packet is routed to the best (longest or most specific) match. Host routes are considered to be subnets whose masks are "all ones" (0xffffffff).

OSPF traffic can be authenticated or non-authenticated, or can use no authentication, simple/clear text passwords, or MD5-based passwords. This means that only trusted routers can participate in the AS routing. You can specify a variety of authentication schemes and, in fact, you can configure separate authentication schemes for each IP subnet.

Externally derived routing data (for example, routes learned from an exterior protocol such as BGP) is advertised throughout the AS. This externally derived data is kept separate from the OSPF link state data.

Each external route can also be tagged by the advertising router, enabling the passing of additional information between routers on the boundary of the AS.

OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations.
Basic OSPFv2 Configuration

This section describes how to implement basic OSPF routing.

Enabling OSPF Routing For a Specific Context

Use the following configuration example to enable OSPF Routing for a specific context:

```
config
  context context_name
  router ospf
end
```

Notes:

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Enabling OSPF Over a Specific Interface

After you enable OSPF, specify the networks on which it will run. Use the following command to enable OSPF:

```
network network_ip_address/network_mask area { area_id | area_ip_address }
```

---

Important

The default cost for OSPF on the system is 10. To change the cost, refer to the `ip ospf cost` command in the Ethernet Interface Configuration Mode Commands chapter of the Command Line Interface Reference.

Notes:

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Redistributing Routes Into OSPF (Optional)

Redistributing routes into OSPF means any routes from another protocol that meet specified a specified criterion, such as route type, metric, or rule within a route-map, are redistributed using the OSPFv2 protocol to all OSPF areas. This is an optional configuration.

```
config
  context context_name
  router ospf
    redistribute { connected | static }
end
```

Notes:

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.
Confirming OSPF Configuration Parameters

To confirm the OSPF router configuration, use the following command and look for the section labeled `router ospf` in the screen output:

```
show config context ctx_name [verbose]
```

OSPFv3 Routing

This section gives an overview of Open Shortest Path First Version 3 (OSPFv3) routing and its implementation in the system. It also describes how to enable the base OSPFv3 functionality and lists the commands that are available for more complex configurations.

**Important**
On the ASR 5000, OSPFv3 routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

OSPFv3 Overview

Much of OSPF version 3 is the same as OSPF version 2. OSPFv3 expands on OSPF version 2 to provide support for IPv6 routing prefixes and the larger size of IPv6 addresses. OSPFv3 dynamically learns and advertises (redistributes) IPv6 routes within an OSPFv3 routing domain.

In OSPFv3, a routing process does not need to be explicitly created. Enabling OSPFv3 on an interface will cause a routing process and its associated configuration to be created.

Basic OSPFv3 Configuration

This section describes how to implement basic OSPF routing.

Enabling OSPFv3 Routing For a Specific Context

Use the following configuration example to enable OSPF Routing for a specific context:

```
config
    context context_name
    router ospfv3
end
```

Notes:

- Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.
Enabling OSPFv6 Over a Specific Interface

After you enable OSPFv3 specify the area in which it will run. Use the following command to enable OSPFv3:

```
area { area_id | area_ip_address } [ default-cost dflt-cost ] [ stub stub-area ] [ virtual-link
vl-neighbor-ipv4address ]
```

Important: The default cost for OSPFv3 on the system is 10. To change the cost, refer to the `ipv6 ospf cost` command in the Ethernet Interface Configuration Mode Commands chapter of the Command Line Interface Reference.

Notes:

- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Redistributing Routes Into OSPFv3 (Optional)

Redistributing routes into OSPFv3 means any routes from another protocol that meet specified a specified criterion, such as route type, metric, or rule within a route-map, are redistributed using the OSPFv3 protocol to all OSPF areas. This is an optional configuration.

```
config context context_name
  router ospf3
    redistribute { connected | static }
  end
```

Notes:

- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Confirming OSPFv3 Configuration Parameters

To confirm the OSPF router configuration, use the following command and look for the section labeled `router ipv6 ospf` in the screen output:

```
show config context ctxt_name [ verbose ]
```

Equal Cost Multiple Path (ECMP)

The system supports ECMP for routing protocols. ECMP distributes traffic across multiple routes that have the same cost to lessen the burden on any one route.

ECMP can be used in conjunction with most routing protocols, since it is a per-hop decision that is limited to a single router. It potentially offers substantial increases in bandwidth by load-balancing traffic over multiple paths.
The following command configures the maximum number of equal cost paths that can be submitted by a routing protocol:

```
config
    context context_name
        ip routing maximum-paths [max_num]
```

Notes:

- `max_num` is an integer from 1 through 10 (releases prior to 18.2) or 1 through 32 (release 18.2+).
- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

## BGP-4 Routing

The Border Gateway Protocol 4 (BGP-4) routing protocol is supported through a BGP router process that is implemented at the context level.

Border Gateway Protocol (BGP) is an inter-AS routing protocol. An Autonomous System (AS) is a set of routers under a single technical administration that use an interior gateway protocol and common metrics to route packets within the AS. The set of routers uses an exterior gateway protocol to route packets to other autonomous systems.

BGP runs over TCP. This eliminates the need for the BGP protocol to implement explicit update fragmentation, retransmission, acknowledgement, and sequencing information. Any authentication scheme used by TCP may be used in addition to BGP's own authentication mechanisms.

BGP routers exchange network reachability information with other BGP routers. This information builds a picture of AS connectivity from which routes are filtered and AS-level policy decisions are enforced.

BGP-4 provides classless inter-domain routing. This includes support for advertising an IP prefix and eliminates the concept of network class within BGP. BGP-4 also allows the aggregation of routes, including the aggregation of AS paths.

---

Important: On the ASR 5000, BGP routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

---

### Overview of BGP Support

Mobile devices communicate to the Internet through Home Agents (HAs). HAs assign IP addresses to the mobile node from a configured pool of addresses. These addresses are also advertised to Internet routers through an IP routing protocol to ensure dynamic routing. The BGP-4 protocol is used as a monitoring mechanism between an HA and Internet router with routing to support Interchassis Session Recovery (ICSR). (Refer to Interchassis Session Recovery for more information.)

The objective of BGP-4 protocol support is to satisfy routing requirements and monitor communications with Internet routers. BGP-4 may trigger an active to standby switchover to keep subscriber services from being interrupted.

The following BGP-4 features are supported:

- Exterior Border Gateway Protocol (EBGP) multi-hop
• Route Filtering for inbound and outbound routes
• Route redistribution and route-maps
• Support for BGP communities and extended communities in route maps
• Local preference for IPv4 and IPv6 (IBGP peers)

IP pool routes and loopback routes are advertised in the BGP domain in the following ways:

• Through BGP Configuration Mode redistribution commands, all or some of the connected routes are redistributed into the BGP domain. (IP pool and loopback routes are present in the IP routing table as connected routes.) The network routemap command provides the flexibility to change many BGP attributes.

• Through the BGP Configuration Mode network commands, connected routes are explicitly configured for advertisement into the BGP domain. The network routemap command provides the flexibility to change many BGP attributes. Refer to the BGP Configuration Mode Commands chapter of the Command Line Interface Reference for details on these commands.

Important
If a BGP task restarts because of a processing card failure, a migration, a crash, or the removal of a processing card, all peering session and route information is lost.

Configuring BGP

This section describes how to configure and enable basic BGP routing support in the system.

config
  context context_name
  router bgp AS_number
  neighbor ip_address remote-as AS_num

Notes:
• A maximum of 64 BGP peers are supported per context.
• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Redistributing Routes Into BGP (Optional)

Redistributing routes into BGP simply means that any routes from another protocol that meet a specified criterion, such as a route type, or a rule within a route-map, are redistributed through the BGP protocol to all BGP areas. This is an optional configuration.

config
  context context_name
  router bgp as_number
  redistribute bgp { bgp | connected | static } [ metric metric_value | [ metric-type { 1 | 2 } ] ] [ route-map route_map_name ]

Notes:
• The redistribution options are connected, ospf, rip, or static. Refer to the Border Gateway Protocol Configuration Mode Commands chapter of the Command Line Interface Reference for details on the redistribute command.

• A maximum of 64 route-maps are supported per context.

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

BGP Communities and Extended Communities

Route filtering based on a BGP community or extended community (route target) is configured via CLI Route Map Configuration mode commands.

BGP Communities

Configuring a BGP Community

A BGP community is a group of destinations that share some common attribute. Each destination can belong to multiple communities. Autonomous system administrators define to which communities a destination belongs.

You configure a BGP community via a Context Configuration mode command.

```plaintext
cfg
  context context_name
  ip community-list { named named_list | standard identifier } { deny | permit } { internet | local-AS | no-advertise | no-export | value AS-community_number AS-community_number AS-community_number ...
    { internet | local-AS | no-advertise | no-export | value AS-community_number AS-community_number AS-community_number ...
    { internet | local-AS | no-advertise | no-export | value AS-community_number AS-community_number AS-community_number ...

You can permit or deny the following BGP community destinations.

• internet – Advertise this route to the internet community, and any router that belongs to it.

• local-AS – Use in confederation scenarios to prevent sending packets outside the local autonomous system (AS).

• no-advertise – Do not advertise this route to any BGP peer, internal or external.

• no-export – Do not advertise to external BGP (eBGP) peers. Keep this route within an AS.

• value AS-community_number – Specifies a community string in AS:NN format, where AS = 2-byte AS-community hexadecimal number and NN = 2-byte hexadecimal number (1 to 11 characters).

You can enter multiple destinations and AS community numbers for each community. For additional information, see the Command Line Interface Reference.

Multiple community-list entries can be attached to a community-list by adding multiple permit or deny clauses for various community strings. Up to 64 community-lists can be configured in a context.
Setting the Community Attribute

You set the BGP community attribute via a `set community` command in a route map.

```
config
  context context_name
  route-map map_name { deny | permit } sequence_number
  set community [additive] { internet | local-AS | no-advertise | no-export | none | value AS-community_number AS-community_number AS-community_number ...
    { internet | local-AS | no-advertise | no-export | none | value AS-community_number AS-community_number ...
      { internet | local-AS | no-advertise | no-export | none | value AS-community_number AS-community_number ...
    }
  }
```

The `additive` option allows you to enter multiple destinations and AS community numbers. For additional information, see the `Command Line Interface Reference`.

Filtering via a BGP Community

To filter routes based on a BGP community, you configure a `match` clause in a route map. The command sequence follows below.

```
config
  context context_name
  route-map map_name { deny | permit } sequence_number
  match community { named named_list | standard identifier }
```

BGP Extended Communities

Configuring a BGP Extended Community (Route Target)

A BGP extended community defines a route target. MPLS VPNs use a 64-bit Extended Community attribute called a Route Target (RT). An RT enables distribution of reachability information to the correct information table.

You configure a BGP extended community via a Context Configuration mode command.

```
config
  context context_name
  ip extcommunity-list { named named_list | standard identifier } { deny | permit } rt rt_number rt_number ...
```

`rt_number` specifies a Route Target as a string in AS:NN format, where AS = 2-byte AS-community hexadecimal number and NN = 2-byte hexadecimal number (1 to 11 characters). You can add multiple route numbers to an IP extcommunity list.

Multiple extended community-list entries can be attached to an extended community-list by adding multiple permit or deny clauses for various extended community strings. Up to 64 extended community-lists can be configured in a context.
Setting the Extended Community Attribute

You set the BGP extended community attribute via a `set extcommunity` command in a route map.

```
config
  context context_name
  route-map map_name { deny | permit } sequence_number
  set extcommunity rt rt_number rt_number rt_number ...
```

`rt_number` specifies a Route Target as a string in AS:NN format, where AS = 2-byte AS-community hexadecimal number and NN = 2-byte hexadecimal number (1 to 11 characters). You can add multiple route numbers to an IP extcommunity list.

Filtering via a BGP Extended Community

To filter routes based on a BGP extended community (route target), you configure a `match` clause in a route map. The command sequence follows below.

```
config
  context context_name
  route-map map_name { deny | permit }
  [no] match extcommunity { named named_list | standard identifier }
```

BGP Local Preference

The BGP local preference attribute is sent by a BGP speaker only to IBGP peers. It is set in a route map via the following command sequence:

```
config
  context context_name
  route-map map_name { deny | permit }
  set local-preference pref_number
```

There is no `match` clause corresponding to local preference in the route-map because local-preference is directly used in the route selection algorithm.

ICSR and SRP Groups

BGP is employed with Interchassis Session Recovery (ICSR) configurations linked via Service Redundancy Protocol (SRP). By default an ICSR failover is triggered when all BGP peers within a context are down.

Optionally, you can configure SRP peer groups within a context. ICSR failover would then occur if all peers within a group fail. This option is useful in deployments in which a combination of IPv4 and IPv6 peers are spread across multiple paired VLANs, and IPv4 or IPv6 connectivity is lost by all members of a peer group.

For additional information refer to `Interchassis Session Recovery` in this guide and the description of the `monitor bgp`, `monitor diameter` and `monitor authentication-probe` commands in the `Service Redundancy Protocol Configuration Mode Commands` chapter of the `Command Line Interface Reference`.

Advertising BGP Routes from a Standby ICSR Chassis

An SRP Configuration mode command enables advertising BGP routes from an ICSR chassis in standby state. This command and its keywords allow an operator to take advantage of faster network convergence accrued
from deploying BGP Prefix Independent Convergence (PIC) in the Optical Transport Network Generation Next (OTNGN).

BGP PIC is intended to improve network convergence which will safely allow for setting aggressive ICSR failure detection timers.

```
configure
  context context_name
    service-redundancy-protocol
      advertise-routes-in-standby-state [ hold-off-time hold-off-time ] | [ reset-bfd-nbrs bfd-down-time ]
    end
```

Notes:

- **hold-off-time** hold-off-time delays advertising the BGP routes until the timer expires. Specify hold-off-time in seconds as an integer from 1 to 300.
- After resetting BFD, **reset-bfd-nbrs bfd-down-time** keeps the BFD sessions down for the configured number of milliseconds to improve network convergence. Specify bfd-down-time as an integer from 50 to 120000.

### Configurable BGP Route Advertisement Interval for ICSR

By default, the MinRtAdvInterval is set for each peer with a value of 5 seconds for an iBGP peer and 30 seconds for an eBGP peer. An operator can use the `neighbor identifier advertisement-interval` command to globally change the default interval.

The BGP advertisement-interval can also be separately set for each address family. If configured, this value over-rides the peer's default advertisement-interval for that address-family only. BGP will send route update-message for each AFI/Safi based on the advertisement-interval configured for that AFI/SAFI. If no AFI/SAFI advertisement-interval is configured, the peer-based default advertisement-interval is used.

In ICSR configurations, this feature can be used to speed route advertisements and improve network convergence times.

The `timers bgp icsr-aggr-advertisement-interval` command is available in both the BGP Address-Family (VPNv4/VPNv6) Configuration and BGP Address-Family (VRF) Configuration modes.

```
configure
  context context_name
    router bgp as_number
      address-family { ipv4 | ipv6 | vpnv4 | vpnv6 }
        timers bgp icsr-aggr-advertisement-interval seconds
```

Notes:

- **seconds** – sets the number of seconds as an integer from 0 to 30. Default: 0.

### BGP CLI Configuration Commands

The following table lists the BGP Configuration mode CLI commands that support the configuration of various BGP parameters. For additional information, refer to the BGP Configuration Mode Commands chapter of the Command Line Interface Reference.
configure
context context_name
router bgp as_number

Table 36: BGP Configuration Mode CLI Commands

<table>
<thead>
<tr>
<th>bgp Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept-zero-as-rd</td>
<td>Configures to accept VPN prefixes with Route Distinguisher (RD) value having Administrator Subfield, which is an AS number 0.</td>
</tr>
<tr>
<td>address-family { ipv4</td>
<td>ipv6 }</td>
</tr>
<tr>
<td>address-family { vvpn4</td>
<td>vvpn6 }</td>
</tr>
<tr>
<td>bgp graceful-restart { restart-time rest_time</td>
<td>stalepath-time stale_time</td>
</tr>
<tr>
<td>description text</td>
<td>Allows you to enter descriptive text for this configuration.</td>
</tr>
<tr>
<td>distance { admin distance prefix prefix_addr</td>
<td>route-access-list list_name</td>
</tr>
<tr>
<td>enforce-first-as</td>
<td>Enforces the first AS for Exterior Border Gateway Protocol (eBGP) routes.</td>
</tr>
<tr>
<td>ip vrf vrf_name</td>
<td>Adds a preconfigured IP VRF context instance to the BGP ASN and configures the BGP attributes and related parameters to the VRF. This</td>
</tr>
<tr>
<td>maximum-paths { ebgp max_num</td>
<td>ibgp max_num }</td>
</tr>
</tbody>
</table>
confirming bgp configuration parameters

To confirm the BGP router configuration, use the following command and look for the section labeled `router bgp` in the screen output:

```
show config context  ctxt_name  [ verbose ]
```

bidirectional forwarding detection

Bidirectional Forwarding Detection (BFD) is a network protocol used to detect faults between two forwarding engines connected by a link. BFD establishes a session between two endpoints over a particular link. If more than one link exists between two systems, multiple BFD sessions may be established to monitor each one of

---

**bgp Command**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| neighbor ip_address { activate | Configures BGP routers that interconnect to non-broadcast networks. Note that a remote AS number must be specified for a neighbor before other parameters can be configured.  

**Note:** The advertisement-interval must be explicitly configured for an address-family so that it can take effect for that address-family. By default it will be applicable only for the IPv4 address-family. Specify the address family via the `address-family` command. You can then set the neighbor advertisement-interval in the address family configuration mode.

| network ip_address/mask [ route-map map_name ] | Specifies a network to announce via BGP. |
| redistribute { connected | Redistributes routes via BGP from another protocol to BGP neighbors. |
| route-map map_name ] | router-id ip_address | Overrides the configured router identifier and causes BGP peers to reset. |
| scan-time time | Configures the BGP background scanner interval in seconds. BGP monitors the next hop of the installed routes to verify next-hop reachability and to select, install, and validate the BGP best path. loops. |
| timers bgp keepalive-interval interval holdtime-interval time | Configures BGP routing timers. |
| min-peer-holdtime-interval time ] | |

---

**confirming bgp configuration parameters**

### bidirectional forwarding detection

Bidirectional Forwarding Detection (BFD) is a network protocol used to detect faults between two forwarding engines connected by a link. BFD establishes a session between two endpoints over a particular link. If more than one link exists between two systems, multiple BFD sessions may be established to monitor each one of
them. The session is established with a three-way handshake, and is torn down the same way. Authentication may be enabled on the session. A choice of simple password, MD5 or SHA1 authentication is available.

**Overview of BFD Support**

BFD does not have a discovery mechanism; sessions must be explicitly configured between endpoints. BFD may be used on many different underlying transport mechanisms and layers, and operates independently of all of these. Therefore, it needs to be encapsulated by whatever transport it uses.

Protocols that support some form of adjacency setup, such as OSPF or IS-IS, may also be used to bootstrap a BFD session. These protocols may then use BFD to receive faster notification of failing links than would normally be possible using the protocol's own keepalive mechanism.

In asynchronous mode, both endpoints periodically send Hello packets to each other. If a number of those packets are not received, the session is considered down.

When Echo is active, a stream of Echo packets is sent to the other endpoint which then forwards these back to the sender. Echo can be globally enabled via the `bfd-protocol` command, and/or individually enabled/disabled per interface. This function is used to test the forwarding path on the remote system.

The system supports BFD in asynchronous mode with optional Echo capability via static or BGP routing.

---

**Important**

On an ASR 5000 one of the packet processing cards must be configured as a demux card in order for BFD to function. See the *Configuring a Demux Card* section in the *System Settings* chapter for additional information.

---

**Configuring BFD**

This section describes how to configure and enable basic BFD routing protocol support in the system. There are several factors affecting the configuration of BFD protocol:

- Configuring a BFD Context, on page 234
- Configuring IPv4 BFD for Static Routes, on page 234
- Configuring IPv6 BFD for Static Routes, on page 234
- Configuring BFD for Single Hop, on page 235
- Configuring Multihop BFD, on page 235
- Scaling of BFD, on page 236
- Associating BGP Neighbors with the Context, on page 236
- Associating OSPF Neighbors with the Context, on page 236
- Associating BFD Neighbor Groups with the BFD Protocol, on page 236
- Enabling BFD on OSPF Interfaces, on page 237
- Monitoring BFD Connection for ICSR, on page 237
Configuring a BFD Context

```
config
    context context_name
        bfd-protocol [ bfd echo ]
    exit
```

Notes:

- Echo function can be optionally enabled for all interfaces in this context.
- 16 BFD sessions per context and 64 per chassis.

Configuring IPv4 BFD for Static Routes

Enable BFD on an interface.

```
config
    context bfd_context_name
        interface if_name
            ip address ipv4_address ipv4_mask
            bfd interval interval_value min_rx rx_value multiplier multiplier_value [ bfd echo ]
        exit
```

Configure BFD static route.

```
ip route static bfd if_name ipv4_gw_address
```

Add static routes.

```
    ip route ipv4_address ipv4_mask
    ip route ipv4_address ipv4_mask
```

Configuring IPv6 BFD for Static Routes

Enable BFD on an Interface

```
config
    context bfd_context_name
        interface if_name
            ipv6 address ipv6_address ipv6_mask
            bfd interval interval_value min_rx rx_value multiplier multiplier_value [ bfd echo ]
        exit
```

Configure BFD static route.

```
ipv6 route static bfd if_name ipv6_gw_address
```

Add static routes.

```
    ipv6 route ipv6_address ipv6_mask
    ipv6 route ipv6_address ipv6_mask
```
On the ASR 5000, static routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

---

**Important**

On the ASR 5000, static routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

### Configuring BFD for Single Hop

Enable BFD on an interface.

```config
context bfd_context_name
    interface if_name
        ip address ipv4_address ipv4_mask
        ipv6 address ipv6_address ipv6_mask
        bfd interval interval_value min_rx rx_value multiplier multiplier_value
        [ bfd echo ]
    exit
```

Enable BFD on a BGP Neighbor. For additional information, see [Associating BGP Neighbors with the Context, on page 236.](#)

Enable BFD on an OSPF Neighbor. For additional information, see [Associating OSPF Neighbors with the Context, on page 236.](#)

---

**Important**

On the ASR 5000, routes with IPv6 prefix lengths less than /12 and between the range of /64 and /128 are not supported.

### Configuring Multihop BFD

Enable BFD on an interface.

```config
context bfd_context_name
    interface if_name
        ip address ipv4_address ipv4_mask
        ipv6 address ipv6_address ipv6_mask
        bfd interval interval_value min_rx rx_value multiplier multiplier_value
        [ bfd echo ]
    exit
```

Configure a Multihop BFD session.

```config
    bfd-protocol
        bfd multihop peer destination-address interval interval-value multiplier multiplier-value
```

Enable BFD on a BGP Neighbor. For additional information, see [Associating BGP Neighbors with the Context, on page 236.](#)
Scaling of BFD

Configure an active BFD session using one of the above methods and use same BFD neighbor while configuring the active interface. For additional information, see Associating BFD Neighbor Groups with the BFD Protocol, on page 236.

```
  bfd-protocol
    bfd nbr-group-name grp_name active-if-name if_name nexthop_address
```

Apply the same BFD results to one or more passive interfaces.

```
  bfd nbr-group-name grp_name passive-if-name if_name nexthop_address
```

Associating BGP Neighbors with the Context

```
config
  context context_name
    router bgp AS_number
      neighbor neighbor_ip-address remote-as rem_AS_number
      neighbor neighbor_ip-address ebgp-multihop max-hop max_hops
      neighbor neighbor_ip-address update-source update-src_ip-address
      neighbor neighbor_ip-address failover bfd [multihop]
```

Notes:

- Repeat the sequence to add neighbors.

Associating OSPF Neighbors with the Context

```
config
  context context_name
    router ospf
      neighbor neighbor_ip-address
```

Notes:

- Repeat the sequence to add neighbors.

Associating BFD Neighbor Groups with the BFD Protocol

```
config
  context context_name
    bfd-protocol
      bfd nbr-group-name grp_name active-if-name if_name nexthop_address
      bfd nbr-group-name grp_name passive-if-name if_name nexthop_address
```
Enabling BFD on OSPF Interfaces

All OSPF Interfaces

```plaintext
config
context context_name
router ospf
bfd-all-interfaces
```

Specific OSPF Interface

```plaintext
config
context context_name
interface interface_name
broadcast
ip ospf bfd
```

Monitoring BFD Connection for ICSR

For ICSR configurations, the following command sequence initiates monitoring of the connection between the primary chassis and the BFD neighbor in the specified context. If the connection drops, the standby chassis becomes active.

```plaintext
config
context context_name
service-redundancy-protocol
monitor bfd context context_name { ipv4_address | ipv6_address } { chassis-to-chassis | chassis-to-router }
```

Notes:

- `ipv4_address | ipv6_address` defines the IP address of the BFD neighbor to be monitored, entered using IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation
- `chassis-to-chassis` enables BFD to run between primary and backup chassis on non-SRP links.
- `chassis-to-router` enables BFD to run between chassis and router.

Saving the Configuration

Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Chassis-to-Chassis BFD Monitoring for ICSR

An operator can configure BFD to more quickly advertise routes during an ICSR switchover. This solution complements the feature that allows the advertising of BGP routes from a Standby ICSR chassis. The overall goal is to support more aggressive failure detection and recovery in an ICSR configuration when implementing of VoLTE.

You must configure the following features for chassis-to-chassis BFD monitoring in ICSR configurations:
Enable Primary Chassis BFD Monitoring

You must enable monitoring of the connection between the primary chassis and specified BFD neighbors. If the connection drops, the standby chassis becomes active. For more information, see Monitoring BFD Connection for ICSR, on page 237.

Set BFD to Ignore ICSR Dead Interval

The SRP Configuration mode `bfd-mon-ignore-dead-interval` command causes the standby ICSR chassis to ignore the dead interval and remain in the standby state until all the BFD chassis-to-chassis monitors fail.

Enable this feature in association with BFD chassis-to-chassis monitoring to support more aggressive ICSR failure detection times.

```
configure
  context context_name
    service-redundancy-protocol variable
      bfd-mon-ignore-dead-interval
    end
```

Configure ICSR Switchover Guard Timer

The SRP Configuration mode `guard timer` command configures the redundancy-guard-period and monitor-damping-period for SRP service monitoring.

Use these guard timers to ensure that local failures, such as card reboots and task restarts, do not result in ICSR events which can be disruptive.

```
configure
  context context_name
    service-redundancy-protocol variable
      guard-timer { aaa-switchover-timers { damping-period seconds | guard-period seconds } |
        diameter-switchover-timers { damping-period seconds | guard-period seconds } | srp-redundancy-timers
        { aaa { damping-period seconds | guard-period seconds } |
          bgp { damping-period seconds } |
          guard-period seconds } |
        diam { damping-period seconds | guard-period seconds } }
    end
```

Notes:

- `aaa-switchover-timers` – sets timers that prevent back-to-back ICSR switchovers due to an AAA failure (post ICSR switchover) while the network is still converging.

  - `damping-period` – configures a delay time to trigger an ICSR switchover due to a monitoring failure within the guard-period.

  - `guard-period` – configures the local-failure-recovery network-convergence timer.
• diameter-switchover-timers – sets timers that prevent a back-to-back ICSR switchover due to a Diameter failure (post ICSR switchover) while the network is still converging.
  - damping-period – configures a delay time to trigger an ICSR switchover due to a monitoring failure within the guard-period.
  - guard-period – configures the local-failure-recovery network-convergence timer.

• srp-redundancy-timers – sets timers that prevent an ICSR switchover while the system is recovering from a local card-reboot/critical-task-restart failure.
  - aaa – local failure followed by AAA monitoring failure
  - bgp – local failure followed by BGP monitoring failure
  - diam – local failure followed by Diameter monitoring failure

Enable BFD Multihop Fall-over

A fall-over bfd multihop mhsess_name keyword in the Context Configuration mode ip route and ipv6 route commands enables fall-over BFD functionality for the specified multihop session. The fall-over bfd option uses BFD to monitor neighbor reachability and liveness. When enabled it will tear down the session if BFD signals a failure.

```configure
context context_name
  ip route { ip_address/ip_mask | ip_address ip_mask } { gateway_ip_address | next-hop
  next_hop_ip_address | point-to-point | tunnel } egress_intrfc_name [ cost cost ] fall-over bfd multihop
  mhsess_name [ [ precedence precedence ] [ vrf vrf_name ] [ cost value ] [ fall-over bfd multihop
  mhsess_name [ [ precedence precedence ] ] +
end```

The ip route command now also allows you to add a static multihop BFD route.

```ip route static multihop bfd mhbfd_sess_name local_endpt_ipaddr remote_endpt_ipaddr```

**Important**

SNMP traps are generated when BFD sessions go up and down (BFDSSessUp and BFDSSessDown).

**ip route Command**

```configure
context context_name
  ip route { ip_address/ip_mask | ip_address ip_mask } { gateway_ip_address | next-hop
  next_hop_ip_address | point-to-point | tunnel } egress_intrfc_name [ cost cost ] fall-over bfd multihop
  mhsess_name [ [ precedence precedence ] [ vrf vrf_name ] [ cost value ] [ fall-over bfd multihop
  mhsess_name [ [ precedence precedence ] ] +
end```

The ip route command now also allows you to add a static multihop BFD route.

```ip route static multihop bfd mhbfd_sess_name local_endpt_ipaddr remote_endpt_ipaddr```
ip routev6 Command

```
configure
context context_name
  ipv6 route ipv6_address/prefix_length { interface name | next-hop ipv6_address interface name } [ cost cost ] [ fall-over bfd multihop mhsess_name ] [ precedence precedence ] [ vrf vrf_name ] [ cost value ] [ fall-over bfd multihop mhsess_name ] [ precedence precedence ] end
```

The `ipv6 route` command now also allows you to add a static multihop BFD route.

```
ipv6 route static multihop bfd mhbfd_sess_name local_endpt_ipv6addr remote_endpt_ipv6addr
```

Adjust BFD Interval

Set the transmit interval (in milliseconds) between BFD packets to meet the convergence requirements of your network deployment.

```
configure
context context_name
  interface interface_name broadcast
    bfd interval interval_num min_rx milliseconds multiplier value
end
```

Notes:

- `milliseconds` is an integer from 50 through 10000. (Default 50)

Enable Advertising BGP Routes from Standby ICSR Chassis

For information on configuring the feature, see Advertising BGP Routes from a Standby ICSR Chassis, on page 229.

Saving the Configuration

Save your configuration as described in the Verifying and Saving Your Configuration chapter.

BFD Support for Link Aggregation Member Links

Member-link based BFD detects individual link failures faster than LACP and reduces the overall session/traffic down period as a result of single member link failure.
Overview

A BFD Configuration mode CLI command configures BFD interactions with the linkagg task. Once a session is configured, BFD creates per member link BFD sessions and starts sending packets on each of the linkagg member links. If a member link BFD session fails, StarOS notifies failures to the linkagg task.

Figure 18: BFD Interactions

If you define a linkagg-peer using a slot number, you may configure a linkagg-peer for a redundant LC (Line Card) slot which must also specify a slot in its member-link configuration. Likewise, if you configure a linkagg-peer without a slot, you must delete it before configuring a peer with a slot specified.

Important

Only one IPv4 or IPv6 BFD session-based configuration is allowed per linkagg interface for compliance with RFC 7130.

Configuring Support for BFD Linkagg Member-links

The `bfd linkagg-peer` command enables member-link BFD and configures the BFD link aggregation (linkagg) session values [RFC 7130].

```
configure
  context context_name
  bfd-protocol
```
**Saving the Configuration**

Save your configuration as described in the *Verifying and Saving Your Configuration* chapter.

**Viewing Routing Information**

To view routing information for the current context, run one of the following Exec mode commands:

- `show ip route`: Displays information for IPv4 routes in the current context.
- `show ipv6 route`: Displays information for ipv6 routes in the current context.
- `show ip static-route`: Displays information only for IPv4 static routes in the current context.
- `show ip ospf`: Displays IPv4 OSPF process summary information in the current context.
- `show ipv6 ospf`: Displays IPv6 OSPFv3 process summary information in the current context.
- `show ip bgp`: Displays IPv4 BGP information.

This example shows sample output of the command, `show ip route`.

```
[local]
host_name# show ip route
** indicates the Best or Used route.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Nexthop</th>
<th>Protocol</th>
<th>Prec</th>
<th>Cost</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>*44.44.44.0/24</td>
<td>208.230.231.50</td>
<td>static</td>
<td>1</td>
<td>0</td>
<td>local1</td>
</tr>
<tr>
<td>*192.168.82.0/24</td>
<td>0.0.0.0</td>
<td>connected</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>*192.168.83.0/24</td>
<td>0.0.0.0</td>
<td>connected</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>208.230.231.0/24</td>
<td>0.0.0.0</td>
<td>ospf</td>
<td>110</td>
<td>10</td>
<td>local1</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>*208.230.231.0/24</th>
<th>0.0.0.0</th>
<th>connected 0 0 local1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total route count: 5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This chapter provides information on configuring virtual local area networks (VLANs) in support of enhanced or extended services. The product administration guides provide examples and procedures for configuration of services on the system that may utilize VLANs. You should select the configuration example that best meets your service model before using the procedures described below.

**VLANs**

Virtual LANs (VLANs) provide greater flexibility in the configuration and use of contexts and services. They are configured as "tags" on a per-port basis and allow more complex configurations to be implemented. The VLAN tag allows a single physical port to be bound to multiple logical interfaces that can be configured in different contexts. Therefore, each Ethernet port can be viewed as containing many logical ports when VLAN tags are employed.

**Overview**

VLAN – Layer 2 Traffic Management is a Cisco feature that requires a separate license. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the Managing License Keys section of Software Management Operations.

- Overview, page 245
- Creating VLAN Tags, page 247
- Verifying the Port Configuration, page 247
- Configuring Subscriber VLAN Associations, page 248
- VLAN-Related CLI Commands, page 249

Important

VLANS are supported in conjunction with subscriber traffic ports on Management I/O (MIO/UMIO) cards. The system supports the configuration limits for VLANs as described in Engineering Rules.
Overlapping IP Address Pool Support – GGSN

Overlapping IP Address pools provides allow operators to more flexibly support multiple corporate VPN customers with the same private IP address space without expensive investments in physically separate routers or virtual routers.

The system supports two types of overlapping pools – resource and overlap. Resource pools are designed for dynamic assignment only, and use a VPN tunnel (such as a GRE tunnel) to forward and receive the private IP addresses to and from the VPN. Overlapping type pools can be used for both dynamic and static addressing, and use VLANs and a next hop forwarding address to connect to the VPN customer.

To forward downstream traffic to the correct PDP context, the GGSN uses either the GRE tunnel ID or the VLAN ID to match the packet. When forwarding traffic upstream, the GGSN uses the tunnel and forwarding information in the IP pool configuration; overlapping pools must be configured in the APN in such instances.

When a PDP context is created, the IP address is assigned from the IP pool. In this case the forwarding rules are also configured into the GGSN. If the address is assigned statically, when the GGSN confirms the IP address from the pool configured in the APN, the forwarding rules are also applied.

The GGSN can scale to as many actual overlapping pools as there are VLAN interfaces per context, and there can be multiple contexts per GGSN. The limit is the number of IP pools. This scalability allows operators who wish to provide VPN services to customers using the customer’s private IP address space, not to be concerned about escalating hardware costs or complex configurations.

RADIUS VLAN Support – Enhanced Charging Services

VPN customers often use private address space which can easily overlap with other customers. The subscriber addresses are supported with overlapping pools which can be configured in the same virtual routing context.

RADIUS Server and NAS IP addresses do not need to be in separate contexts, thereby simplifying APN and RADIUS configuration and network design. This feature allows the following scenarios to be defined in the same context:

- Overlapping RADIUS NAS-IP addresses for various RADIUS server groups representing different APNs.
- Overlapping RADIUS server IP addresses for various RADIUS servers groups.

Every overlapping NAS-IP address is given a unique next-hop address which is then bound to an interface that is bound to a unique VLAN, thereby allowing the configuration to exist within the same context.

The system forwards RADIUS access requests and accounting messages to the next hop defined for that NAS-IP; the connected routers forward the messages to the RADIUS server. The next hop address determines the interface and VLAN to use. Traffic from the server is identified as belonging to a certain NAS-IP by the port/VLAN combination.

The number of RADIUS NAS-IP addresses that can be configured is limited by the number of loopback addresses that can be configured.

APN Support – PDN Gateway (P-GW)

P-GW Access Point Name (APN) supports extensive parameter configuration flexibility for the APN. VLAN tagging may be selected by the APN, but are configured in the P-GW independently from the APN.
Creating VLAN Tags

Use the following example to create VLANs on a port and bind them to pre-existing interfaces. For information on creating interfaces, refer to System Interfaces and Ports.

```
config
   port ethernet slot/port
      no shutdown
      vlan vlan_tag_ID
      no shutdown
      bind interface interface_name context_name
end
```

Notes:

- **Optional**: Configure VLAN-subscriber associations. Refer to Configuring Subscriber VLAN Associations, on page 248 for more information.
- Repeat this procedure as needed to configure additional VLANs for the port.
- Refer to VLAN-Related CLI Commands, on page 249 and the Command Line Interface Reference for additional information.
- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Verifying the Port Configuration

Run the following command to verify the port configuration:

```
[local]/host_name# show port info slot/port
```

An example of this command's output when at least one VLAN has been configured for the port is shown below:

```text
Port: 5/11
   Port Type : 10G Ethernet
   Role : Service Port
   Description : (None Set)
   Redundancy Mode : Port Mode
   Redundant With : 6/11
   Preferred Port : Non-Revertive
   Physical ifIndex : 85262336
   Administrative State : Enabled
   Configured Duplex : Auto
   Configured Speed : Auto
   Fault Unidirection Mode : 802.3ae clause 46
   Configured Flow Control : Enabled
   Interface MAC Address : 64-9E-F3-69-5B-EA
   SRP Virtual MAC Address : None
   Fixed MAC Address : 64-9E-F3-69-5B-CA
   Link State : Up
   Link Duplex : Full
   Link Speed : 10 Gb
   Flow Control : Enabled
   Link Aggregation Group : None
   Untagged:
      Logical ifIndex : 85262337
      Operational State : Up, Active
   Tagged VLAN: VID 10
      Logical ifIndex : 285270210
      VLAN Type : Standard
      VLAN Priority : 0
```
Administrative State : Enabled
Operational State : Up, Active
Number of VLANs : 1
SFP Module : Present (10G Base-SR)

Notes:

• Repeat this sequence as needed to verify additional ports.

• Optional: Configure VLAN-subscriber associations. Refer to Configuring Subscriber VLAN Associations, on page 248 for more information.

• Refer to VLAN-Related CLI Commands, on page 249 for additional information.

• Save your configuration as described in the Verifying and Saving Your Configuration chapter.

Configuring Subscriber VLAN Associations

Subscriber traffic can be routed to specific VLANs based on the configuration of their user profile. This functionality provides a mechanism for routing all traffic from a subscriber over the specified VLAN. All packets destined for the subscriber must also be sent using only IP addresses valid on the VLAN or they will be dropped.

RADIUS Attributes Used

The following RADIUS attributes can be configured within subscriber profiles on the RADIUS server to allow the association of a specific VLAN to the subscriber:

• SN-Assigned-VLAN-ID: In the Starent VSA dictionary
• SN1-Assigned-VLAN-ID: In the Starent VSA1 dictionary

Important

Since the instructions for configuring subscriber profiles differ between RADIUS server applications, this section only describes the individual attributes that can be added to the subscriber profile. Please refer to the documentation that shipped with your RADIUS server for instructions on configuring subscribers.

Configuring Local Subscriber Profiles

Use the configuration example below to configure VLAN associations within local subscriber profiles on the system.

Important

These instructions assume that you have already configured subscriber-type VLAN tags according to the instructions provided in Creating VLAN Tags, on page 247.

```bash
config
context context_name
  subscriber name user_name
```
Verify the Subscriber Profile Configuration

Use the following command to view the configuration for a subscriber profile:

```
[local]host_name# show subscriber configuration username user_name
```

Notes:

- Repeat this command for each subscriber.
- Save your configuration as described in the Verifying and Saving Your Configuration chapter.

VLAN-Related CLI Commands

VLAN-related features and functions are supported across several CLI command modes. The following tables identify commands associated with configuration and monitoring of VLAN-related functions.

For detailed information regarding the use of the commands listed below, see the Command Line Interface Reference.

Table 37: VLAN-Related Configuration Commands

<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
</table>
| AAA Server Group Configuration Mode | `radius attribute nas-ip-address address
ip_address nexthop-forwarding-address
ip_address vlan vlan_id` | Sets the RADIUS client to provide the VLAN ID with the nexthop forwarding address to a system when running in single nexthop gateway mode.  
**Note:** To access the `vlan` keyword, `aaa-large configuration` must be enabled via the Global Configuration mode. |
| ACS Charging Action Configuration Mode | `ip vlan vlan_id` | Configures the VLAN identifier to be associated with the subscriber traffic in the destination context. |
| Context Configuration Mode | `ip pool pool_name nexthop forwarding address ip_address overlap vlanid
vlan_id` | When a nexthop forwarding address is configured, the `overlap vlanid` keyword enables support for overlapping IP address pools and associates the pool with the specified VLAN ID. |
| Context Configuration Mode | `ip routing overlap-pool` | Advertises overlap-pool addresses in dynamic routing protocols when overlap pools are configured using VLAN IDs. When enabled, the overlap addresses are added as interface addresses and advertised. |
### VLAN-Related CLI Commands

<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Configuration Mode</td>
<td><code>radius attribute nas-ip-address address ip_address nexthop-forwarding-address ip_address vlan vlan_id</code></td>
<td>Specifies the VLAN ID to be associated with the next-hop IP address.</td>
</tr>
<tr>
<td>Ethernet Interface Configuration Mode</td>
<td><code>[no] logical-port-statistics</code></td>
<td>Enables or disables the collection of logical port (VLAN and NPU) bulk statistics for the first 32 configured Ethernet or PVC interface types.</td>
</tr>
<tr>
<td>Ethernet Interface Configuration Mode</td>
<td><code>vlan-map next-hop ipv4_address</code></td>
<td>Sets a single next-hop IP address so that multiple VLANs can use a single next-hop gateway. The vlan-map is associated with a specific interface.</td>
</tr>
<tr>
<td>Ethernet Port Configuration Mode</td>
<td><code>vlan vlan_id</code></td>
<td>Enters VLAN Configuration mode.</td>
</tr>
<tr>
<td>PVC Configuration Mode</td>
<td><code>[no] shutdown</code></td>
<td>Enables or disables traffic over a specified VLAN. See below.</td>
</tr>
<tr>
<td>Subscriber Configuration Mode</td>
<td><code>ip vlan vlan_id</code></td>
<td>Configures the subscriber VLAN ID that is used with the assigned address for the subscriber session to receive packets. If the IP pool from which the address is assigned is configured with a VLAN ID, this subscriber configured VLAN ID overrides it.</td>
</tr>
<tr>
<td>VLAN Configuration Mode</td>
<td><code>bind interface interface_name context_name</code></td>
<td>Binds a virtual interface and context to support VLAN service.</td>
</tr>
<tr>
<td>VLAN Configuration Mode</td>
<td><code>[no] ingress-mode</code></td>
<td>Enables or disables port ingress (incoming) mode.</td>
</tr>
<tr>
<td>VLAN Configuration Mode</td>
<td><code>priority value</code></td>
<td>Configures an 802.1p VLAN priority bit for ASN-GW service only.</td>
</tr>
<tr>
<td>VLAN Configuration Mode</td>
<td><code>[no] shutdown</code></td>
<td>Enables or disables traffic over the current VLAN.</td>
</tr>
<tr>
<td>VLAN Configuration Mode</td>
<td><code>vlan-map interface if_name context_name</code></td>
<td>Associates an IP interface having a VLAN ID with a context.</td>
</tr>
</tbody>
</table>

### Table 38: VLAN-Related Monitoring Commands

<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exec Mode show commands</td>
<td><code>clear port slot/port vlan vlan_id</code></td>
<td>Clears NPU statistics for the port that has a previously configured VLAN ID.</td>
</tr>
<tr>
<td>CLI Mode</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Exec Mode show commands</td>
<td>`show logical-port utilization table vlan {5-minute</td>
<td>hourly }`</td>
</tr>
<tr>
<td>Exec Mode show commands</td>
<td><code>show port info slot/port vlan vlan_id</code></td>
<td>Displays NPU counters for a previously configured VLAN ID.</td>
</tr>
</tbody>
</table>
This chapter describes services that are supported by the ASR 5x00 for Border Gateway Protocol (BGP) Multi-Protocol Label Switching (MPLS) Virtual Private Networks (VPNs).

Important
MPLS is a licensed Cisco feature that requires a separate license. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the Managing License Keys section of Software Management Operations.

It includes the following topics:

- Introduction, page 253
- MPLS-CE Connected to PE, page 254
- ASR 5x00 as a PE, page 255
- IPv6 Support for BGP MPLS VPNs, page 257
- VPN-Related CLI Commands, page 260

Introduction

Service providers require the ability to support a large number of corporate Access Point Names (APNs) which have a number of different addressing models and requirements. The ASR 5x00 uses BGP MPLS Layer 3 VPNs to segregate corporate customer APNs in a highly scalable manner. This solution conforms to RFC 4364 – BGP/MPLS IP Virtual Private Networks (VPNs).

The BGP/MPLS solution supports the following scenarios:

- MPLS-CE Connected to PE, on page 254
- ASR 5x00 as a PE, on page 255

The ASR 5x00 also supports VPNv6 as described in RFC 4659 – BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN. See IPv6 Support for BGP MPLS VPNs, on page 257 for details.
MPLS-CE Connected to PE

In this scenario the ASR 5x00 functions as an MPLS-CE (Customer Edge) network element connected to a Provider Edge (PE) Label Edge Router (LER), which in turn connects to the MPLS core (RFC 4364). See the figure below.

Figure 19: ASR 5x00 MPLS-CE to PE

The MPLS-CE functions like a PE router within its own Autonomous System (AS). It maintains Virtual Routing and Forwarding (VRF) routes and exchanges VPN route information with the PE via an MP-eBGP (Multi-Protocol-external BGP) session.

The PE is also configured with VRFs and exchanges VPN routes with other PEs in its AS via MP-iBGP (Multi-Protocol-internal BGP) connections and the MPLS-CE via an MP-eBGP connection.

The EBGP connection allows the PE to change next-hop IP addresses and labels in the routes learned from IBGP peers before advertising them to the MPLS-CE. The MPLS-CE in this case uses only MP-eBGP to advertise and learn routes. Label Distribution Protocol (LDP) and Resource Reservation Protocol (RSVP) are not required because of direct-connect EBGP peering. The MPLS-CE in this scenario pushes/pops a single label (learned over the MP-eBGP connection) to/from the PE.
ASR 5x00 as a PE

Overview

In this scenario, the ASR 5x00 functions as a PE router sitting at the edge of the MPLS core. See the figure below.

Figure 20: ASR 5x00 as a PE

The ASR 5x00 eliminates the need for an ASBR or PE as shown in the first two scenarios. In this scenario, two main requirements are introduced: IBGP functionality and MPLS label distribution protocols.

The ASR 5x00 can be configured to add two labels:

- an outer label learned from LDP or RSVP-TE (RSVP-Traffic Engineering)
- an inner label learned from MP-iBGP

This solution supports traffic engineering and QoS initiated via the ASR 5x00.

Sample Configuration

In this example, VRFs are configured on the ASR 5x00 PE and pools are associated with VRFs. The ASR 5x00 exchanges VPN routes with its IBGP peers (PE routers) and learns the MPLS paths to reach PEs via
LDP. The ASR 5x00 forwards the packets to the next-hop with two labels – an inner label learned from PE and an outer label learned from the next hop IBGP neighbor.

Figure 21: Sample Configuration

```
mpls ip
  protocol ldp
  enable
  exit
  exit

ip vrf vrf1
  mpls traffic-class copy
  exit
ip vrf vrf2
  mpls traffic-class value 5
  exit

router bgp 300
  ip vrf vrf1
    route-target export 300 1
    route-target import 300 1
    route-distinguisher 300 1
  exit
  ip vrf vrf2
    route-target export 300 2
    route-target import 300 2
    route-distinguisher 300 2
  exit

router-id 2.2.2.2
neighbor 192.168.107.20 remote-as 300
neighbor 192.168.107.20 update-source node1_loopback

address-family vpnv4
  neighbor 192.168.107.20 activate
  neighbor 192.168.107.20 send-community both
  neighbor 192.168.107.20 next-hop-self
  exit

address-family ipv4 vrf vrf1
  redistribute connected
  exit

address-family ipv4 vrf vrf2
  redistribute connected
  exit

interface interface_to_internet
  ip address 192.168.109.65/24
  mpls ip
  exit
router ospf
```
IPv6 Support for BGP MPLS VPNs

Overview

The ASR 5x00 supports VPNv6 as described in RFC 4659 – *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*.

An IPv6 VPN is connected over an IPv6 interface or sub-interface to the Service Provider (SP) backbone via a PE router. The site can be both IPv4 and IPv6 capable. Each VPNv6 has its own address space which means a given address denotes different systems in different VPNs. This is achieved via a VPNv6 address-family which prepends a Route Distinguisher (RD) to the IP address.

A VPNv6 address is a 24-byte quantity beginning with an 8-byte RD and ending with a 16-byte IPv6 address. When a site is IPv4 and IPv6 capable, the same RD can be used for the advertisement of both IPv4 and IPv6 addresses.

The system appends RD to IPv6 routes and exchanges the labeled IPv6-RD using the VPNv6 address-family. The Address Family Identifier (AFI) and Subsequent Address Family Identifier (SAFI) fields for VPNv6 routes will be set to 2 and 128 respectively.

The IPv6 VPN traffic will be transported to the BGP speaker via IPv4 tunneling. The BGP speaker advertises to its peer a Next Hop Network Address field containing a VPN-IPv6 address whose 8-octet RD is set to zero and whose 16-octet IPv6 address is encoded as an IPv4-mapped IPv6 address (RFC 4291) containing the IPv4 address of the advertising router. It is assumed that only EBGP peering will be used to exchange VPNv6 routes.

Support for VPN-IPv6 assumes the following:

- Dual Stack (IPv4/IPv6) routing
- IPv6 pools in VRFs
- BGP peering over a directly connected IPv4 interface

See the figure below.

*Figure 22: IPv6-RD Support for VPNv6*
Sample Configuration

This example assumes three VRFs. VRF 1 has only IPv4 routes, VRF 2 has both IPv4 and IPv6 routes, and VRF 3 has only IPv6 routes.

Configure VRFs.

ip vrf vrf1
exit
ip vrf vrf2
exit
ip vrf vrf3
exit

Enable MPLS BGP forwarding.

mpls bgp forwarding

Configure pools.

ip pool vrf1-pool 51.52.53.0 255.255.255.0 private 0 vrf vrf1
exit
ip pool vrf2-pool 51.52.53.0 255.255.255.0 private 0 vrf vrf2
exit
ipv6 pool vrf2-v6pool prefix 2005:0101::/32 private 0 vrf vrf2
exit
ipv6 pool vrf3-v6pool prefix 2005:0101::/32 private 0 vrf vrf3
exit

Configure interfaces.

interface ce_interface_to_rtr
  ip address 192.168.110.90 255.255.255.0
exit
interface ce_v6_interface
  ip address 2009:0101:0101:0101::1/96
exit
interface ce_loopback loopback
  ip address 52.1.2.3 255.255.255.255
exit
interface vrf1-loop loopback
  ip vrf forwarding vrf1
  ip address 1.52.53.54 255.255.255.255
exit
interface vrf2-loop loopback
  ip vrf forwarding vrf2
  ip address 2.52.53.54 255.255.255.255
exit
interface vrf2-v6loop loopback
  ip vrf forwarding vrf2
ip address 2005:0202:0101::1/128
exit
interface vrf3-v6loop loopback
ip vrf forwarding vrf3
ip address 2005:0303:0101::1/128
exit

Configure BGP along with address families and redistribution rules.

router bgp 800
  router-id 1.1.1.1
  neighbor 192.168.110.20 remote-as 1003
  neighbor 192.168.110.20 activate
  address-family vpnv4
    neighbor 192.168.110.20 activate
    neighbor 192.168.110.20 send-community both
  exit
  address-family vpnv6
    neighbor 192.168.110.20 activate
    neighbor 192.168.110.20 send-community both
  exit
  ip vrf vrf1
    route-distinguisher 800 1
    route-target export 800 1
    route-target import 800 1
  exit
  address-family ipv4 vrf vrf1
    redistribute connected
    redistribute static
  exit
  ip vrf vrf2
    route-distinguisher 800 2
    route-target export 800 2
    route-target import 800 2
  exit
  address-family ipv4 vrf vrf2
    redistribute connected
    redistribute static
  exit
  address-family ipv6 vrf vrf2
    redistribute connected
    redistribute static
  exit
  ip vrf vrf3
    route-distinguisher 800 3
    route-target export 800 3
    route-target import 800 3
  exit
  address-family ipv6 vrf vrf3
    redistribute connected
    redistribute static
  exit

Configure APNs.

apn walmart51.com
  selection-mode sent-by-ms
  accounting-mode none
  aaa group walmart-group
  authentication pap 1 chap 2 allow-noauth
  ip context-name Gi_ce
  ip address pool name vrf1-pool
  exit
apn amazon51.com
  selection-mode sent-by-ms
  accounting-mode none
  aaa group amazon-group
  authentication pap 1 chap 2 allow-noauth
  ip context-name Gi_ce
  ip address pool name vrf2-pool
  ipv6 address prefix-pool vrf2-v6pool
  exit
apn apple51.com
  selection-mode sent-by-ms
VPN-Related CLI Commands

VPN-related features and functions are supported across several CLI command modes. The following tables identify commands associated with configuration and monitoring of VPN-related functions.

For detailed information regarding the use of the commands listed below, see the Command Line Interface Reference.

Table 39: VPN-Related Configuration Commands

<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP Address-Family (IPv4/IPv6) Configuration Mode</td>
<td>neighbor ip_address activate</td>
<td>Enables the exchange of routing information with a peer router.</td>
</tr>
<tr>
<td>BGP Address-Family (IPv4/IPv6) Configuration Mode</td>
<td>neighbor ip_address send community { both</td>
<td>extended</td>
</tr>
<tr>
<td>BGP Address-Family (IPv4/IPv6) Configuration Mode</td>
<td>redistribute connected</td>
<td>Redistributes routes into BGP from another protocol as BGP neighbors.</td>
</tr>
<tr>
<td>BGP Address-Family (VPNv4) Configuration Mode</td>
<td>neighbor ip_address activate</td>
<td>Enables the exchange of routing information with a peer router.</td>
</tr>
<tr>
<td>BGP Address-Family (VPNv4) Configuration Mode</td>
<td>neighbor ip_address send community { both</td>
<td>extended</td>
</tr>
<tr>
<td>BGP Address-Family (VRF) Configuration Mode</td>
<td>neighbor ip_address activate</td>
<td>Enables the exchange of routing information with a peer router.</td>
</tr>
<tr>
<td>CLI Mode</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>----------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>BGP Address-Family (VRF)</td>
<td>neighbor <em>ip_address</em> send community {both</td>
<td>extended</td>
</tr>
<tr>
<td>BGP Address-Family (VRF)</td>
<td>redistribute connected</td>
<td>Redistributes routes into BGP from another protocol as BGP neighbors.</td>
</tr>
<tr>
<td>BGP Configuration Mode</td>
<td>address-family { ipv4 vrf vrf_name</td>
<td>vpnv4 }</td>
</tr>
<tr>
<td>BGP Configuration Mode</td>
<td>address-family { ipv6 vrf vrf_name</td>
<td>vpnv6 }</td>
</tr>
<tr>
<td>BGP Configuration Mode</td>
<td>ip vrf vrf_name</td>
<td>Adds a VRF to BGP and switches to the VRF Configuration mode to allow configuration of BGP attributes for the VRF.</td>
</tr>
<tr>
<td>BGP IP VRF Configuration Mode</td>
<td>route-distinguisher { as_value</td>
<td><em>ip_address</em></td>
</tr>
<tr>
<td>BGP IP VRF Configuration Mode</td>
<td>route-target { both</td>
<td>import</td>
</tr>
<tr>
<td>Context Configuration Mode</td>
<td>ip pool pool_name addr_range vrf vrf_name [ mpls-label input inlabel1 output outlabel1 outlabel2 ]</td>
<td>Configures a pool into the specified VRF. This parameter must be specified with the Next-Hop parameter. <em>inlabel1</em> is the MPLS label that identifies inbound traffic destined for this pool. <em>outlabel1</em> and <em>outlabel2</em> specify the MPLS labels to be added to packets sent for subscribers from this pool.</td>
</tr>
<tr>
<td>Context Configuration Mode</td>
<td>ip vrf vrf_name</td>
<td>Creates a VRF and assigns a VRF-ID. A VRF is created in the router.</td>
</tr>
<tr>
<td>CLI Mode</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Context Configuration Mode</td>
<td><code>ipv6 pool pool_name vrf vrf_name</code></td>
<td>Associates the pool with that VRF.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> By default the configured ipv6 pool will be associated with the global routing domain.</td>
</tr>
<tr>
<td>Context Configuration Mode</td>
<td><code>mpls bgp forwarding</code></td>
<td>Globally enables MPLS Border Gateway Protocol (BGP) forwarding.</td>
</tr>
</tbody>
</table>
| Context Configuration Mode           | `mpls exp value`                             | Sets the default behavior as Best Effort using a zero value in the 3-bit MPLS EXP header. This value applies to all the VRFs in the context. The default behavior is to copy the DSCP value of mobile subscriber traffic to the EXP header, if there is no explicit configuration for DSCP to EXP (via the `mpls map-dscp-to-exp dscp n exp m` command).  
`mpls exp` disables the default behavior and sets the EXP value to the configured value. |
| Context Configuration Mode           | `mpls ip`                                    | Globally enables the MPLS forwarding of IPv4 packets along normally routed paths.                                                            |
| Context Configuration Mode           | `radius change-authorize-nas-ip  
  ip_address ip_address {  
  encrypted | key } value port  
  port_num  
  mpls input inlabel  
  output outlabel1 outlabel2` | Configures COA traffic to use the specified MPLS labels. `inlabel` identifies inbound COA traffic. `outlabel1` and `outlabel2` specify the MPLS labels to be added to the COA response. `outlabel1` is the inner output label; `outlabel2` is the outer output label. |
<p>| Ethernet Interface Configuration Mode| <code>mpls ip</code>                                    | Enables dynamic MPLS forwarding of IP packets on this interface.                                                                            |
| Exec Mode                            | <code>clear ip bgp peer</code>                          | Clears BGP sessions.                                                                                                                         |
| Exec Mode                            | <code>lsp-ping ip_prefix_FEC</code>                    | Checks MPLS Label-Switched Path (LSP) connectivity for the specified forwarding equivalence class (FEC). It must be followed by an IPv4 or IPv6 FEC prefix. |</p>
<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exec Mode</td>
<td><code>lsp-traceroute ip_prefix_FEC</code></td>
<td>Discovers MPLS LSP routes that packets actually take when traveling to their destinations. It must be followed by an IPv4 or IPv6 FEC prefix.</td>
</tr>
<tr>
<td>IP VRF Context Configuration Mode</td>
<td><code>mpls map-dscp-to-exp dscp dscp_bit_value exp exp_bit_value</code></td>
<td>Maps the final differentiated services code point (DSCP) bit value in the IP packet header to the final Experimental (EXP) bit value in the MPLS header for incoming traffic.</td>
</tr>
<tr>
<td>IP VRF Context Configuration Mode</td>
<td><code>mpls map-exp-to-dscp exp exp_bit_value dscp dscp_bit_value</code></td>
<td>Maps the incoming EXP bit value in the MPLS header to the internal DSCP bit value in IP packet headers for outgoing traffic.</td>
</tr>
<tr>
<td>MPLS-IP Configuration Mode</td>
<td><code>protocol ldp</code></td>
<td>Creates the MPLS protocol family configuration modes, or configures an existing protocol and enters the MPLS-LDP Configuration Mode in the current context. This command configures the protocol parameters for the MPLS protocol family.</td>
</tr>
<tr>
<td>MPLS-LDP Configuration Mode</td>
<td>`advertise-labels { explicit-null</td>
<td>implicit-null }`</td>
</tr>
<tr>
<td>MPLS-LDP Configuration Mode</td>
<td>`discovery { hello { hello-interval seconds</td>
<td>hold-interval seconds }</td>
</tr>
<tr>
<td>MPLS-LDP Configuration Mode</td>
<td><code>enable</code></td>
<td>Enables Label Distribution Protocol (LDP).</td>
</tr>
<tr>
<td>MPLS-LDP Configuration Mode</td>
<td><code>router-id ip_address</code></td>
<td>Configures the LDP Router ID.</td>
</tr>
<tr>
<td>MPLS-LDP Configuration Mode</td>
<td>`session timers { hold-interval seconds</td>
<td>keepalive-interval seconds }`</td>
</tr>
</tbody>
</table>
### Table 40: VPN-Related Monitoring Commands

<table>
<thead>
<tr>
<th>CLI Mode</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show ip bgp neighbors</code></td>
<td>Displays information regarding BGP neighbors.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td>`show ip bgp vpnv4 { all</td>
<td>route-distinguisher</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show ip bgp vpnv6</code></td>
<td>Displays contents of VPNv6 routing table.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td>`show ip bgp vpnv6 { all</td>
<td>route-distinguisher</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show ip pool</code></td>
<td>Displays pool details including the configured VRF.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls cross-connect</code></td>
<td>Displays MPLS cross-connect information. MPLS tunnel cross-connects between interfaces and Label-Switched Paths (LSPs) connect two distant interface circuits of the same type via MPLS tunnels that use LSPs as the conduit.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls ft { vrf vrf_name }</code></td>
<td>Displays MPLS FEC-to-NHLFE (FTN) table information.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls ft { vrf vrf_name }</code></td>
<td>Displays contents of the MPLS FTN table for a specified VRF.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls ilm</code></td>
<td>Displays MPLS Incoming Label Map (ILM) table information.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls ldp</code></td>
<td>Displays the MPLS LDP information.</td>
</tr>
<tr>
<td>Exec Mode show Commands</td>
<td><code>show mpls nexthop-label-forwarding-entry</code></td>
<td>Displays MPLS Next-Hop Label Forwarding Entry (NHLFE) table information.</td>
</tr>
</tbody>
</table>
This chapter provides information on configuring Content Service Steering (CSS). The product administration
guides provide examples and procedures for configuration of basic services on the system. You should select
the configuration example that best meets your service model, and configure the required elements for that
model as described in the respective product administration guide, before using the procedures described
below.

Important

Internal CSS is a generic feature, if an ECSv2 license is installed on your system, internal CSS can be
enabled. A separate license is not required to enable internal CSS. Contact your local Cisco account
representative for information on how to obtain a license.

This chapter contains the following topics:

- **Overview**, page 265
- **Configuring Internal Content Service Steering**, page 266

Overview

Content Server Selection (CSS) is a StarOS function that defines how traffic will be handled based on the
"content" of the data presented by a mobile subscriber (or to a mobile subscriber). CSS is a broad term that
includes features such as load balancing, NAT, HTTP redirection, and DNS redirection.

The content server (services) can be either external to the platform or integrated inside the platform.

CSS uses Access Control Lists (ACLs) to redirect subscriber traffic flows. ACLs control the flow of packets
into and out of the system. ACLs consist of "rules" (ACL rules) or filters that control the action taken on
packets matching the filter criteria.

ACLs are configurable on a per-context basis and applies to a subscriber through either a subscriber profile
(or an APN profile in the destination context. For additional information, refer to the *Access Control Lists*
chapter.
Configuring Internal Content Service Steering

To configure and activate a single CSS service for redirecting all of a subscriber's IP traffic to an internal in-line service:

Step 1
Define an IP ACL as described in Defining IP Access Lists for Internal CSS, on page 266.

Step 2
Optional: Apply an ACL to an individual subscriber as described in Applying an ACL to an Individual Subscriber (Optional), on page 267.

Step 3
Optional: Apply a single ACL to multiple subscribers as described in Applying an ACL to Multiple Subscribers (Optional), on page 267.

Step 4
Optional: Apply an ACL to multiple subscribers via APNs as described in Applying an ACL to Multiple Subscriber via APNs, on page 209.

Step 5
Save your configuration to flash memory, an external memory device, and/or a network location using the Exec mode command save configuration. For additional information on how to verify and save configuration files, refer to the System Administration Guide and the Command Line Interface Reference.

Commands used in the configuration examples in this section provide base functionality to the extent that the most common or likely commands and/or keyword options are presented. In many cases, other optional commands and/or keyword options are available. Refer to the Command Line Interface Reference for complete information regarding all commands. Not all commands or keywords/variables may be supported or available. Availability varies on the platform type and installed license(s).

Defining IP Access Lists for Internal CSS

IP ACLs specify what type of subscriber traffic and which direction (uplink, downlink, or both) traffic is redirected. The IP ACL must be specified in the context in which subscriber authentication is performed.

Caution
To minimize the risk of data loss, do not make configuration changes to ACLs while the system is facilitating subscriber sessions.

Use the following configuration example to define an IP ACL for internal CSS; start in the Exec mode of the CLI:

```
configure
  context context_name
    ip access-list acl_name
      redirect css service service_name keywords options
    end
end
```

Notes:

- `service_name` must be an ACL service name.
- For information on the keywords and options available with the `redirect css service` command, see the ACL Configuration Mode Commands chapter in the Command Line Interface Reference.
• For IPv6 ACLs, the same configurations must be done in the IPv6 ACL Configuration Mode. See the IPv6 ACL Configuration Mode Commands chapter in the Command Line Interface Reference.

Applying an ACL to an Individual Subscriber (Optional)

For information on how to apply an ACL to an individual subscriber, refer to the Applying an ACL to an Individual Subscriber section of the Access Control Lists chapter.

Applying an ACL to Multiple Subscribers (Optional)

IP ACLs are applied to subscribers via attributes in their profiles. The subscriber profile can be configured locally on the system or remotely on a RADIUS server.

The system provides for the configuration of subscriber functions that serve as default values when specific attributes are not contained in the individual subscriber's profile. When configured properly, the functions can be used to apply an ACL to:

• All subscribers facilitated within a specific context by applying the ACL to the profile of the subscriber named default.
• All subscribers facilitated by specific services by applying the ACL to a subscriber profile and then using the default subscriber command to configure the service to use that subscriber as the "default" profile.

Applying an ACL to the Subscriber Named default (Optional)

For information on how to apply an ACL to the default subscriber, refer to the Applying an ACL to the Subscriber Named default section in the Access Control Lists chapter.

Applying an ACL to Service-specified Default Subscribers (Optional)

For information on how to apply an ACL to the subscriber to be used as the "default" profile by various system services, refer to the Applying an ACL to Service-specified Default Subscribers section in the Access Control Lists chapter.

Applying an ACL to Multiple Subscribers via APNs (Optional)

IP ACLs are applied to subscribers via attributes in their profiles. The subscriber profile can be configured locally on the system or remotely on a RADIUS server.

To reduce configuration time, ACLs can alternatively be applied to APN templates. When configured, any subscriber packets facilitated by the APN template would then have the associated ACL applied.

For information on how to apply an ACL to multiple subscribers via APNs, refer to the Applying a Single ACL to Multiple Subscribers via APNs section in the Access Control Lists chapter.
Session Recovery

With robust hardware failover and redundancy protection, any card-level hardware failures on the system can quickly be corrected. However, software failures can occur for numerous reasons, often without prior indication.

This chapter describes the Session Recovery feature that provides seamless failover and reconstruction of subscriber session information in the event of a hardware or software fault.

---

Important

Session Recovery is a licensed Cisco feature. A separate feature license may be required. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the Managing License Keys section of Software Management Operations.

This chapter includes the following sections:

- How Session Recovery Works, page 269
- Additional ASR 5x00 Hardware Requirements, page 272
- Configuring the System to Support Session Recovery, page 272
- Recovery Control Task Statistics, page 276

---

How Session Recovery Works

This section provides an overview of how this feature is implemented and the recovery process.

The Session Recovery feature provides seamless failover and reconstruction of subscriber session information in the event of a hardware or software fault within the system preventing a fully connected user session from being disconnected.

Session recovery is performed by mirroring key software processes (for example, session manager and AAA manager) within the system. These mirrored processes remain in an idle state (standby-mode) wherein they perform no processing, until they may be needed in the event of a software failure (for example, a session manager task aborts).

The system spawns new instances of "standby mode" session and AAA managers for each active control processor (CP) being used. These mirrored processes require both memory and processing resources, which
means that additional hardware may be required to enable this feature (see Additional ASR 5x00 Hardware Requirements, on page 272).

Other key system-level software tasks, such as VPN manager, are performed on a physically separate packet processing card to ensure that a double software fault (for example, session manager and VPN manager fails at same time on same card) cannot occur. The packet processing card that hosts the VPN manager process is in active mode and reserved by the operating system for this sole use when session recovery is enabled.

There are two modes of session recovery.

- **Task recovery mode:** Wherein one or more session manager failures occur and are recovered without the need to use resources on a standby packet processing card. In this mode, recovery is performed by using the mirrored "standby-mode" session manager task(s) running on active packet processing cards. The "standby-mode" task is renamed, made active, and is then populated using information from other tasks such as AAA manager. In case of Task failure, limited subscribers will be affected and will suffer outage only until the task starts back up.

- **Full packet processing card recovery mode:** Used when a packet processing card hardware failure occurs, or when a planned packet processing card migration fails. In this mode, the standby packet processing card is made active and the "standby-mode" session manager and AAA manager tasks on the newly activated packet processing card perform session recovery.

Session/Call state information is saved in the peer AAA manager task because each AAA manager and session manager task is paired together. These pairs are started on physically different packet processing cards to ensure task recovery.

There are some situations wherein session recovery may not operate properly. These include:

- Additional software or hardware failures occur during the session recovery operation. For example, an AAA manager fails while the state information it contained was being used to populate the newly activated session manager task.

- A lack of hardware resources (packet processing card memory and control processors) to support session recovery.

---

**Important**

After a session recovery operation, some statistics, such as those collected and maintained on a per manager basis (AAA Manager, Session Manager, etc.) are in general not recovered, only accounting and billing related information is checkpointed and recovered.

Session Recovery is available for the following functions:

- Any session needing L2TP LAC support (excluding regenerated PPP on top of an HA or GGSN session)
- ASR 5000 only – Closed RP PDSN services supporting simple IP, Mobile IP, and Proxy Mobile IP
- ASR 5000 only – eHRPD service (evolved High Rate Packet Data)
- ASR 5000 only – ePDG service (evolved Packet Data Gateway)
- GGSN services for IPv4 and PPP PDP contexts
- HA services supporting Mobile IP and/or Proxy Mobile IP session types with or without per-user Layer 3 tunnels
- ASR 5000 only – HNB-GW: HNB Session over IuH
Session Recovery

How Session Recovery Works

- ASR 5000 only – HNB-GW: HNB-CN Session over IuPS and IuCS
- ASR 5000 only – HNB-GW: SeGW Session IPSec Tunnel
- ASR 5000 only – HSGW services for IPv4
- IPCF (Intelligent Policy Control Function)
- ASR 5000 only – IPSG-only systems (IP Services Gateway)
- LNS session types (L2TP Network Server)
- MME (Mobility Management Entity)
- ASR 5000 only – NEMO (Network Mobility)
- P-GW services for IPv4
- ASR 5000 only – PDIF (Packet Data Interworking Function)
- PDSN services supporting simple IP, Mobile IP, and Proxy Mobile IP
- S-GW (Serving Gateway)
- SaMOG (S2a Mobility over GTP) Gateway (CGW and MRME)
- ASR 5000 only – SAE-GW (System Architecture Evolution Gateway)
- ASR 5000 only – SGSN services (3G and 2.5G services) for IPv4 and PPP PDP contexts

Session recovery is **not supported** for the following functions:

- Destination-based accounting recovery
- GGSN network initiated connections
- GGSN session using more than 1 service instance
- MIP/L2TP with IPSec integration
- MIP session with multiple concurrent bindings
- Mobile IP sessions with L2TP
- Multiple MIP sessions

**Important**

Always refer to the Administration Guides for individual products for other possible session recovery and Interchassis Session Recovery (ICSR) support limitations.

When session recovery occurs, the system reconstructs the following subscriber information:

- Data and control state information required to maintain correct call behavior.
- A minimal set of subscriber data statistics; required to ensure that accounting information is maintained.
- A best-effort attempt to recover various timer values such as call duration, absolute time, and others.
- The idle time timer is reset to zero and the re-registration timer is reset to its maximum value for HA sessions to provide a more conservative approach to session recovery.
Any partially connected calls (for example, a session where HA authentication was pending but has not yet been acknowledged by the AAA server) are not recovered when a failure occurs.

Additional ASR 5x00 Hardware Requirements

Because session recovery requires numerous hardware resources, such as memory, control processors, NPU processing capacity, some additional hardware may be required to ensure that enough resources are available to fully support this feature.

A minimum of four packet processing cards (three active and one standby) per individual chassis is required to use this feature.

To allow for complete session recovery in the event of a hardware failure during a packet processing card migration, a minimum of three active packet processing cards and two standby packet processing cards should be deployed.

To assist you in your network design and capacity planning, consider the following factors:

- Subscriber capacity is decreased depending on the hardware configuration. A fully configured chassis would experience a smaller decrease in subscriber capacity versus a minimally configured chassis.
- The amount by which control transaction processing capacity is reduced.
- The reduction in subscriber data throughput.
- The recovery time for a failed software task.
- The recovery time for a failed packet processing card.

A packet processing card migration may temporarily impact session recovery as hardware resources (memory, processors, etc.) that may be needed are not available during the migration. To avoid this condition, a minimum of two standby packet processing cards should be configured.

Configuring the System to Support Session Recovery

The following procedures allow you to configure the session recovery feature for either an operational system that is currently in-service (able to accept incoming calls) or a system that is out-of-service (not part of your production network and, therefore, not processing any live subscriber/customer data).

The session recovery feature, even when the feature use key is present, is disabled by default on the system.
Enabling Session Recovery

As noted earlier, session recovery can be enabled on a system that is out-of-service (OOS) and does not yet have any contexts configured, or on an in-service system that is currently capable of processing calls. However, if the system is in-service, it must be restarted before the session recovery feature takes effect.

Enabling Session Recovery on an Out-of-Service System

The following procedure is for a system that does not have any contexts configured.

To enable the session recovery feature on an out-of-service system, follow the procedure below. This procedure assumes that you begin at the Exec mode prompt.

Step 1: At the Exec mode prompt, verify that the session recovery feature is enabled via the session and feature use licenses on the system by running the `show license info` command. If the current status of the Session Recovery feature is Disabled, you cannot enable this feature until a license key is installed in the system.

Step 2: Use the following configuration example to enable session recovery.

```
configure
   require session recovery
end
```

Step 3: Save your configuration as described in Verifying and Saving Your Configuration. The system, when started, enables session recovery, creates all mirrored "standby-mode" tasks, and performs packet processing card reservations and other operations automatically.

Step 4: After the system has been configured and placed in-service, you should verify the preparedness of the system to support this feature as described in Viewing Session Recovery Status, on page 274

Enabling Session Recovery on an In-Service System

When enabling session recovery on a system that already has a saved configuration, the session recovery commands are automatically placed before any service configuration commands in the configuration file.

To enable the session recovery feature on an in-service system, follow the procedure below. This procedure assumes that you begin at the Exec mode prompt.

Step 1: At the Exec mode prompt, verify that the session recovery feature is enabled via the session and feature use licenses on the system by running the `show license info` command. If the current status of the Session Recovery feature is Disabled, you cannot enable this feature until a license key is installed in the system.
Step 2  
Use the following configuration example to enable session recovery.
```
configure
  require session recovery
end
```
This feature does not take effect until after the system has been restarted.

Step 3  
Save your configuration as described in Verifying and Saving Your Configuration.

Step 4  
Perform a system restart by entering the `reload` command:
The following prompt appears:
```
Are you sure? [Yes|No]:
```
Confirm your desire to perform a system restart by entering `yes`.
The system, when restarted, enables session recovery and creates all mirrored "standby-mode" tasks, performs packet processing card reservations, and other operations automatically.

Step 5  
After the system has been restarted, you should verify the preparedness of the system to support this feature as described in Viewing Session Recovery Status, on page 274

More advanced users may opt to simply insert the `require session recovery` command syntax into an existing configuration file using a text editor or other means, and then applying the configuration file manually. Exercise caution when doing this to ensure that this command is placed among the first few lines of any existing configuration file; it must appear before the creation of any non-local context.

---

## Disabling the Session Recovery Feature

To disable the session recovery feature on a system, enter the `no require session recovery` command from the Global Configuration mode prompt.

**Important** If this command is issued on an in-service system, then the system must be restarted by issuing the `reload` command.

## Viewing Session Recovery Status

To determine if the system is capable of performing session recovery, when enabled, enter the `show session recovery status verbose` command from the Exec mode prompt.

The output of this command should be similar to the examples shown below.

```
[local] host_name> show session recovery status
Session Recovery Status:
  Overall Status          : SESSMGR Not Ready For Recovery
  Last Status Update      : 1 second ago

[local] host_name> show session recovery status
Session Recovery Status:
  Overall Status          : Ready For Recovery
  Last Status Update      : 8 seconds ago
```
Session Recovery Status:

Overall Status : Ready For Recovery
Last Status Update : 2 seconds ago

<table>
<thead>
<tr>
<th></th>
<th>sessmgr-</th>
<th>aaamgr-</th>
<th>demux</th>
</tr>
</thead>
<tbody>
<tr>
<td>cpu</td>
<td>state</td>
<td>active</td>
<td>standby</td>
</tr>
<tr>
<td>1/1</td>
<td>Active</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>1/2</td>
<td>Active</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1/3</td>
<td>Active</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2/1</td>
<td>Active</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2/2</td>
<td>Active</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2/3</td>
<td>Active</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3/0</td>
<td>Active</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3/2</td>
<td>Active</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4/1</td>
<td>Standby</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>4/2</td>
<td>Standby</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4/3</td>
<td>Standby</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Viewing Recovered Session Information

To view session state information and any session recovery status, enter the following command:

```bash
[local]host_name# show subscriber debug-info { callid | msid id | username name }
```

The following example shows the output of this command both before and after a session recovery operation has been performed. The "Redundancy Status" fields in this example have been bold-faced for clarity.

```
username: user1  callid: 01ca11b1  msid: 0000100003
Card/Cpu: 4/2
Sessmgr Instance: 7
Primary callline:

Redundancy Status: Original Session
```

```
Card/Cpu: 4/2
Sessmgr Instance: 7
Primary callline:

Redundancy Status: Recovered Session
```

```
username: user1  callid: 01ca11b1  msid: 0000100003
Card/Cpu: 4/2
Sessmgr Instance: 7
Primary callline:

Redundancy Status: Original Session
```

```
Card/Cpu: 4/2
Sessmgr Instance: 7
Primary callline:

Redundancy Status: Recovered Session
```
Session Recovery

Recovery Control Task Statistics

Recovery Control Task (RCT) statistics show the following:

- Recovery action taken – Migration, Shutdown, Switchover
- Type of event – Planned or Unplanned
- From card to card – slot numbers
- Start time – YYYY-MMM-DD+hh:mm:ss.sss
- Duration – seconds
- Card failure device (such as CPU\textsuperscript{n})
- Card failure reason
- Card is in usable state or not failed
- Recovery action status – Success or failure reason
- If recovery action failed, failure time stamp
- If recovery action failed, failure task facility name
- If recovery action failed, failure instance number
show rct stats Command

The Exec mode show rct stats command employs the following syntax:

[local]host_name# show rct stats [verbose]

Without the verbose keyword, a summary output is displayed as show in the example below:

RCT stats summary
-----------------
Migrations = 0
  Management Card: 0 Average time: 0.000 sec
Packet Card : 1 Average time: 0.006 sec
Switchovers = 1, Average time - 25.855 sec

With the verbose keyword the detailed statistics show in Sample Output for show rct stats verbose, on page 277 are provided.

Sample Output for show rct stats verbose

[local]host_name# show rct stats verbose

RCT stats Details (Last 5 Actions)
Stats 1:
  Action : Shutdown
  From : 4
  To : 5
  Start Time : 2013-Aug-30+03:02:00.132
  Duration : 002.804 sec
  Is Card Usable : Yes
  Failure Reason : CPU_CRITICAL_TASK_FAILURE
  Failure Device : CPU_0
  Recovery Status : Success
  Facility : N.A
  Instance : N.A

RCT stats Details (Last 5 Actions)
Stats 2:
  Action : Shutdown
  From : 12
  To : 13
  Start Time : 2013-Aug-30+03:02:10.100
  Duration : 002.901 sec
  Is Card Usable : Yes
  Failure Reason : NPU_LC_CONNECT_TOP_FAIL
  Failure Device : PAC_LC_CONNECT_HARDWARE
  Recovery Status : Success
  Facility : N.A
  Instance : N.A

Stats 3:
  Action : Migration
  From : 7
  To : 11
  Start Time : 2013-Aug-30+03:03:40.120
  Duration : 003.423 sec
  Is Card Usable : Yes
  Failure Reason : N.A.
  Failure Device : N.A
  Recovery Status : Success
  Facility : N.A
  Instance : N.A

Stats 4:
  Action : Migration
Sample Output for show rct stats verbose

From : 7
To : 11
Start Time : 2013-Aug-30+03:03:41.256
Duration : 005.222 sec
Is Card Usable : Yes
Failure Reason : N.A.
Failure Device : N.A.
Recovery Status : TASK_MIGRATION_FAIL_PREMIGRATE
Facility : vpnmgr
Instance : 13

Stats 5:
Action : Migration
From : 6
To : 7
Start Time : 2013-Aug-30+04:18:30.106
Duration : 004.134 sec
Is Card Usable : Yes
Failure Reason : N.A.
Failure Device : N.A.
Recovery Status : TASK_MIGRATION_FAIL_RENAME
Facility : sessmgr
Instance : 63

RCT stats Summary
-----------------
Migrations = 3, Average time = 4.260 sec
Switchovers = 0
Interchassis Session Recovery

This chapter describes how to configure Interchassis Session Recovery (ICSR). The product Administration Guides provide examples and procedures for configuration of basic services on the system. You should select the configuration example that best meets your service model, and configure the required elements for that model as described in the respective product Administration Guide, before using the procedures described below.

Important

ICSR is a licensed Cisco feature that requires a separate license. Contact your Cisco account representative for detailed information on specific licensing requirements. For information on installing and verifying licenses, refer to the Managing License Keys section of Software Management Operations.

This chapter discusses the following:

- Overview, page 279
- ICSR Operation, page 285
- Configuring Interchassis Session Recovery (ICSR), page 289
- Troubleshooting ICSR Operation, page 304
- Updating the Operating System, page 305

Overview

The ICSR feature provides the highest possible availability for continuous call processing without interrupting subscriber services. ICSR allows the operator to configure geographically distant gateways for redundancy purposes. In the event of a node or gateway failure, ICSR allows sessions to be transparently routed around the failure, thus maintaining the user experience. ICSR also preserves session information and state.

ICSR is implemented through the use of redundant chassis. The chassis are configured as primary and backup, with one being active and one standby. Both chassis are connected to the same AAA server. A checkpoint duration timer controls when subscriber data is sent from the active chassis to the standby chassis. If the active chassis handling the call traffic goes out of service, the standby chassis transitions to the active state and continues processing the call traffic without interrupting the subscriber session.
The chassis determine which is active through a proprietary TCP-based connection known as the Service Redundancy Protocol (SRP) link. The SRP link is used to exchange Hello messages between the primary and backup chassis and must be maintained for proper system operation.

ICSR licenses are currently supported for the following services:

- eHRPD – Evolved High Rate Packet Data
- ePDG – Evolved Packet Data Gateway
- GGSN – Gateway GPRS Support Node
- HA – Home Agent
- MME – Mobility Management Entity
- P-GW – Packet Data Network Gateway
- PDSN – Packet Data Serving Node
- S-GW – Serving Gateway
- SAEGW – System Architecture Evolution Gateway
- SaMOG – S2a Mobility over GTP

L2TP Access Concentrator (LAC) functionality for ICSR is supported by the following protocol and services:

- eGTP – enhanced GPRS Tunneling Protocol
- GGSN – Gateway GPRS Support Node
- P-GW – Packet Data Network Gateway
- SAEGW – System Architecture Evolution Gateway

L2TP Access Concentrator (LAC) functionality for ICSR is not supported by the following services:

- HA – Home Agent
- PMIP – Proxy Mobile IP

L2TP Network Server (LNS) functionality for ICSR is not supported by any services.

---

**Important**

For releases prior to 17.0, ICSR should not be configured on chassis supporting L2TP calls.

---

**Important**

ICSR support for LAC requires a separate LAC license, as well as an Inter-Chassis Session Recovery license.

---

**Important**

Contact your Cisco account representative to verify whether a specific service supports ICSR as an option.
Interchassis Communication

Chassis configured to support ICSR communicate using periodic Hello messages. These messages are sent by each chassis to notify the peer of its current state. The Hello message contains information about the chassis such as its configuration and priority. A dead interval is used to set a time limit for a Hello message to be received from the chassis' peer. If the standby chassis does not receive an Hello message from the active chassis within the dead interval, the standby chassis transitions to the active state. In situations where the SRP link goes out of service, a priority scheme is used to determine which chassis processes the session. The following priority scheme is used:

- route modifier
- chassis priority
- MIO/UMIO MAC address

Checkpoint Messages

Checkpoint messages are sent from the active chassis to the standby chassis. These messages are sent at specific intervals and contain all the information needed to recreate the sessions on the standby chassis, if that chassis were to become active. Once a session exceeds the checkpoint duration, checkpoint data is collected on the session.

SRP CLI Commands

Exec Mode CLI Commands

Exec mode srp CLI configuration commands can be used to enable, disable and initiate SRP functions. The table below lists and briefly describes these commands. For complete information see the Exec Mode Commands (D-S) chapter of the Command Line Interface Reference.

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>srp disable nack micro-chkpt-cmd</td>
<td>Enables the sending of NACK messages from the standby chassis that may trigger a full checkpoint from the active chassis. Sending full checkpoints increases SRP bandwidth. This command disables the NACK feature for a specific micro-checkpoint which is failing continuously.</td>
</tr>
<tr>
<td>srp initiate-audit manual-with-sync</td>
<td>Initiates a forced audit between ICSR chassis. This audit ensures that two ICSR peers are synchronized and identifies any discrepancies prior to scheduled or unscheduled switchover events.</td>
</tr>
</tbody>
</table>
### Command Table

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>srp initiate-switchover</strong></td>
<td>Executes a forced switchover from active to inactive. When executed on the active chassis, this command switches the active chassis to the inactive state and the inactive chassis to an active state. See Note below.</td>
</tr>
<tr>
<td><strong>srp reset-auth-probe-fail</strong></td>
<td>Resets the auth probe monitor failure information to 0.</td>
</tr>
<tr>
<td><strong>srp reset-diameter-fail</strong></td>
<td>Resets the Diameter monitor failure information to 0.</td>
</tr>
<tr>
<td><strong>srp terminate-post-process</strong></td>
<td>Forcibly terminates post-switchover processing.</td>
</tr>
<tr>
<td><strong>srp validate-configuration</strong></td>
<td>Validates the configuration for an active chassis.</td>
</tr>
<tr>
<td><strong>srp validate-switchover</strong></td>
<td>Validates that both active and standby chassis are ready for a planned SRP switchover.</td>
</tr>
</tbody>
</table>

### Important

For release 20.0 and higher, ICSR will verify session manager connectivity on both chassis prior to allowing a manual switchover. If one or more of the session managers in the active chassis is not connected on the standby chassis, the switchover will not be initiated. An error message will appear on the screen noting the number of session managers that are mismatched. The **force** keyword can be used to initiate the switchover despite the mismatch(es). The output of the **show checkpoint statistics verbose** command will **not** indicate "Ready" for a session manager instance ("smgr inst") in the "peer conn" column for any instance that is not connected to the peer chassis.

### show Commands

Exec mode **show srp** commands display a variety of information related to SRP functions. The table below lists and briefly describes these commands. For complete information on these commands, see the **Exec Mode show Commands (Q-S)** chapter of the **Command Line Interface Reference**.

#### Table 42: show srp Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>show srp audit-statistics</strong></td>
<td>Displays statistics of an external audit.</td>
</tr>
<tr>
<td><strong>show srp call-loss statistics</strong></td>
<td>Displays the history of calls lost during switchover.</td>
</tr>
<tr>
<td><strong>show srp checkpoint statistics</strong></td>
<td>Displays check pointing statistics on session redundancy data (session managers, current call recovery records, etc.).</td>
</tr>
<tr>
<td><strong>show srp info</strong></td>
<td>Displays Service Redundancy Protocol information (context, chassis state, peer, connection state, etc.).</td>
</tr>
<tr>
<td><strong>show srp monitor</strong></td>
<td>Displays SRP monitor information.</td>
</tr>
<tr>
<td><strong>show srp statistics</strong></td>
<td>Displays SRP statistics (hello messages sent, configuration validation, resource messages, switchovers, etc.).</td>
</tr>
</tbody>
</table>
For additional information about the output of `show srp` commands, see the *Statistics and Counters Reference*.

**AAA Monitor**

AAA servers are monitored using the authentication probe mechanism. AAA servers are considered Up if the authentication-probe receives a valid response. AAA servers are considered Down when the `max-retries count` specified in the configuration of the AAA server has been reached. SRP initiates a switchover when none of the configured AAA servers responds to an authentication probe. AAA probing is only performed on the active chassis.

---

**Important**

A switchover event caused by an AAA monitoring failure is non-revertible.

If the newly active chassis fails to monitor the configured AAA servers, it remains as the active chassis until one of the following occurs:

- a manual switchover
- another non-AAA failure event causes the system to switchover
- a CLI command is used to clear the AAA failure flag and allow the chassis to switch to standby

**BGP Interaction**

The Service Redundancy Protocol implements revertible switchover behavior via a mechanism that adjusts the route modifier value for the advertised loopback/IP Pool routes. The initial value of the route modifier value is determined by the chassis' configured role and is initialized to a value that is higher than a normal operational value. This ensures that in the event of an SRP link failure and an SRP task failure, the correct chassis is still preferred in the routing domain.

---

**Important**

For ICSR you must configure `busyout ip pool` commands *in the same order* on Active and Standby chassis to avoid SRP validation failures.

The Active and Standby chassis share current route modifier values. When BGP advertises the loopback and IP pool routes, it converts the route modifier into an autonomous systems (AS) path prepend count. The Active chassis always has a lower route modifier, and thus prepends less to the AS-path attribute. This causes the route to be preferred in the routing domain.

If communication on the SRP link is lost, and both chassis in the redundant pair are claiming to be Active, the previously Active chassis is still preferred since it is advertising a smaller AS-path into the BGP routing domain. The route modifier is incremented as switchover events occur. A threshold determines when the route modifier should be reset to its initial value to avoid rollover.

**Requirements**

ICSR configurations require the following:
• Two chassis configured for the same service types. The services must be bound on an SRP-activated loopback interface.
• Both chassis must have identical hardware.
• Three contexts:
  • **Redundancy** – to configure the primary and backup chassis redundancy.
  • **Source** – AAA configuration of the specified nas-ip-address must be the IP address of an interface bound to an HA, or any core network service configured within the same context.
  • **Destination** – to configure monitoring and routing to the PDN.

• AAA RADIUS server
• Border Gateway Protocol (BGP) – ICSR uses the route modifier to determine the chassis priority.

---

**Important**

ICSR is a licensed Cisco feature. Verify that each chassis has the appropriate license before using these procedures. To do this, log in to both chassis and execute a `show license information` command. Look for "Inter-Chassis Session Recovery". If the chassis is not licensed, please contact your Cisco account representative.
The following figure shows an ICSR network.

*Figure 24: ASR 5500 ICSR Network*

**ICSR Operation**

This section shows operational flows for ICSR.
The following figure shows an ICSR process flow due to a primary failure.

*Figure 25: ICSR Process Flow (Primary Failure)*

- **Primary**
  - State = Active

- **Backup**
  - State = Standby

**Pass Hello messages.**

Send session checkpoint message after call duration of 60 seconds

**Fails and stops sending hello messages.**

**State = Active**

**Hello Message**

**State = Standby**

**Hello Message**

**State = Standby**

**Hello Message**

**State = Standby**

Detects that the primary has not sent a hello within the dead-interval and initiates switchover from the standby to the active state.

**State = Active**

**Dead Interval**

(Default = 60 secs)

**Recovers and begins sending hello messages.**

**State = Standby**

**Hello Message**

**State = Standby**

**Hello Message**

**State = Active**

**Receives hello message with backup status of active and transitions to the standby state.**

**State = Standby**

**Begins advertising the srp-activated loopback and pool routes into the routing domain, and AAA services are enabled. Begins taking calls.**

**State = Active**
The following figure shows an ICSR process flow due to a manual switchover.

**Figure 26: ICSR Process Flow (Manual Switchover)**

- **Primary**
  - State = Active

- **Backup**
  - State = Standby

---

**Pass Hello messages every hello interval.**

**User initiates ICSR switchover command.**
- State = Active-Pending-Standby

**Send outstanding checkpoints.**

---

**Switchover Request Message initiating switchover**

**Waiting for Peer system to respond to switchover request.**
- State = Active-Pending-Standby

---

**Switchover Response Message acknowledging successful switchover to Active state.**
- State = Active

- **Transitions to Standby and updates Route Modifier.**
  - State = Standby

**Session Subsystem retains all checkpoint data.**

---

**Session Audit Request**

**Session Audit Response**
- Send Checkpoint data (from newly Active).

**Session subsystem invalidates old checkpoint data and refreshes with new checkpoints from Active chassis.**

---

**Switchover complete.**

**Exchange Hello Messages every hello interval.**
Chassis Initialization

When the chassis are simultaneously initialized, they send Hello messages to their configured peer. The peer sends a response, establishes communication between the chassis, and messages are sent that contain configuration information.

During initialization, if both chassis are misconfigured in the same mode - both active (primary) or both standby (backup), the chassis with the highest priority (lowest number set with the ICSR priority command) becomes active and the other chassis becomes the standby.

If the chassis priorities are the same, the system compares the two MAC addresses and the chassis with the higher MIO/UMIO MAC address becomes active. For example, if the chassis have MAC addresses of 00-02-43-03-1C-2B and 00-02-43-03-01-3B, the last 3 sets of octets (the first 3 sets are the vendor code) are compared. In this example, the 03-1C-2B and 03-01-3B are compared from left to right. The first pair of octets in both MAC addresses are the same, so the next pairs are compared. Since the 01 is lower than the 1C, the chassis with the MIO/UMIO MAC address of 00-02-43-03-1C-2B becomes active and the other chassis the standby.

Chassis Operation

This section describes how the chassis communicate, maintain subscriber sessions, and perform chassis switchover.

Chassis Communication

If one chassis in the active state and one in the standby state, they both send Hello messages at each hello interval. Subscriber sessions that exceed the checkpoint session duration are included in checkpoint messages that are sent to the standby chassis. The checkpoint message contains subscriber session information so if the active chassis goes out of service, the backup chassis becomes active and is able to continue processing the subscriber sessions. Additional checkpoint messages occur at various intervals whenever subscriber session information is updated on the standby chassis.

The SRP Configuration mode checkpoint session command includes a number of keywords that allows you to:

- Set the type of compression algorithm to be used for SRP payload messages.
- Set the amount of time the chassis waits before checkpointing an existing call session. Checkpoints can be separately set for IMS and/or non-IMS sessions.
- Configure the interval between the sending of macro-checkpoints (full checkpoints) between the active and standby chassis.

For additional information see the Service Redundancy Protocol Configuration Mode Commands chapter in the Command Line Interface Reference.

Chassis Switchover

If the active chassis goes out of service, the standby chassis continues to send Hello messages. If the standby chassis does not receive a response to the Hello messages within the dead interval, the standby chassis initiates a switchover. During the switchover, the standby chassis begins advertising its srp-activated loopback and
pool routes into the routing domain. Once the chassis becomes active, it continues to process existing AAA services and subscriber sessions that had checkpoint information, and is also able to establish new subscriber sessions.

When the primary chassis is back in service, it sends Hello messages to the configured peer. The peer sends a response, establishes communication between the chassis, and sends Hello messages that contain configuration information. The primary chassis receives an Hello message that shows the backup chassis state as active and then transitions to standby. The Hello messages continue to be sent to each peer, and checkpoint information is now sent from the active chassis to the standby chassis at regular intervals.

When chassis switchover occurs, the session timers are recovered. The access gateway session recovery is recreated with the full lifetime to avoid potential loss of the session and the possibility that a renewal update was lost in the transitional checkpoint update process.

**Configuring Interchassis Session Recovery (ICSR)**

---

**Important**

The ICSR configuration must be the same on the primary and backup chassis. If each chassis has a different Service Redundancy Protocol (SRP) configuration, the session recovery feature does not function and sessions cannot be recovered when the active chassis goes out of service.

---

This section describes how to configure basic ICSR on each chassis. For information on commands that configure additional parameters and options, refer to the *Command Line Interface Reference*.

---

**Caution**

ICSR should **not** be configured for chassis supporting L2TP calls.

---

The procedures described below assume the following:

- The chassis have been installed and configured with core network services. For more configuration information and instructions on configuring services, refer to the respective product Administration Guide.
- In addition, the IP address pools must be **srp activated**.
- AAA server is installed, configured and accessible by both chassis. For more information on configuring the AAA server, refer to the *AAA Interface Administration and Reference*.
- BGP router installed and configured. See *Routing* for more information on configuring BGP services.
To configure the ICSR on a primary and/or backup chassis:

**Step 1** Configure the SRP context by applying the example configuration in Configuring the Service Redundancy Protocol (SRP) Context, on page 290.

**Step 2** Modify the source context of the core network service by applying the example configuration in Modifying the Source Context for ICSR, on page 300.

**Step 3** Modify the destination context of core network service by applying the example configuration in Modifying the Destination Context for ICSR, on page 302.

**Step 4** Optional: Disable bulk statistics collection on the standby system by applying the example configuration in Disabling Bulk Statistics Collection on a Standby System, on page 303.

**Step 5** Verify your primary and backup chassis configuration as described in Verifying the Primary and Backup Chassis Configuration, on page 303.

**Step 6** Save your configuration as described in Verifying and Saving Your Configuration.

---

**Configuring the Service Redundancy Protocol (SRP) Context**

To configure the system to work with ICSR:

**Step 1** Create the chassis redundancy context and bind it to the IP address of the primary chassis by applying the example configuration in Creating and Binding the SRP Context, on page 290.

**Step 2** Configure the chassis redundancy context with priority, chassis mode, hello interval, dead-interval and peer IP address by applying the example configuration in Configuring SRP Context Parameters, on page 291.

**Step 3** Configure the SRP context with interface parameters (including interface name, IP address and port number) for interchassis communication by applying the example configuration in Configuring the SRP Context Interface Parameters, on page 297.

**Step 4** Verify your SRP context configuration as described in Verifying SRP Configuration, on page 300.

**Step 5** Save your configuration as described in Verifying and Saving Your Configuration.

---

**Creating and Binding the SRP Context**

Use the example below to create the SRP context and bind it to primary chassis IP address:
ICSR is configured on two systems. Be sure to create the redundancy context on both systems. CLI commands must be executed on both systems. Log onto both chassis before continuing. Always make configuration changes on the primary chassis first. Before starting this configuration, identify which chassis to configure as the primary and use that login session.

```
configure
crntext srp_ctxt_name [-noconfirm]
  service-redundancy-protocol
    bind address ip_address
end
```

Notes:

- ICSR should be configured and maintained in a separate context.
- Be sure to bind the local IP address to the primary chassis. When configuring the backup chassis, be sure to bind the local IP address to the backup chassis.

### Configuring SRP Context Parameters

#### Important

CLI commands must be executed on both chassis. Log onto both chassis before continuing. Always make configuration changes on the primary chassis first.

### Basic Parameters

This configuration assigns a chassis mode and priority, and also configures the redundancy link between the primary and backup chassis:

```
configure
crntext srp_ctxt_name
  service-redundancy-protocol
  chassismode { primary | backup }
priority priority
  peer-ip-address ip_address
  hello-interval dur_sec
dead-interval dead_dur_sec
end
```

Notes:

- ICSR should be configured and maintained in a separate context.
- When assigning the chassis mode on the backup chassis be sure to enter the backup keyword.
- The checkpoint command sets the amount of time the chassis waits before checkpointing an existing call session. Checkpoints can be set for IMS (VoLTE) and/or non-IMS sessions. The checkpoint is a snapshot of the current application state that can be used to restart its execution in case of failure. The default setting is 60 seconds.
The priority determines which chassis becomes active in the event that both chassis are misconfigured with the same chassis mode; see Chassis Initialization, on page 288. The higher priority chassis has the lower number. Be sure to assign different priorities to each chassis.

Enter the IP chassis of the backup chassis as the peer-ip-address to the primary chassis. Assign the IP address of the primary chassis as the peer-ip-address to the backup chassis.

The dead-interval must be at least three times greater than the hello-interval. For example, if the hello interval is 10, the dead interval should be at least 30. System performance is severely impacted if the hello interval and dead interval are not set properly. An optional delay-interval command allows you to delay the start dead-interval for an interval following the loading of configuration files.

SRP Redundancy, AAA and Diameter Guard Timers

Guard timers ensure that local failures, such as card reboots and task restarts, do not result in ICSR events which can be disruptive.

The guard timer command configures the redundancy-guard-period and monitor-damping-period for SRP service monitoring.

```
configure
custom context_name
    service-redundancy-protocol variable
        guard-timer { aaa-switchover-timers { damping-period seconds | guard-period seconds }
            diameter-switchover-timers { damping-period seconds | guard-period seconds }
            srp-redundancy-timers
                aaa { damping-period seconds | guard-period seconds }
                bgp { damping-period seconds }
                diam { damping-period seconds | guard-period seconds }
            }
end
```

Notes:

- **aaa-switchover-timers** – sets timers that prevent back-to-back ICSR switchovers due to an AAA failure (post ICSR switchover) while the network is still converging.
  
  * **damping-period** – configures a delay time to trigger an ICSR switchover due to a monitoring failure within the guard-period.
  
  * **guard-period** – configures the local-failure-recovery network-convergence timer.

- **diameter-switchover-timers** – sets timers that prevent a back-to-back ICSR switchover due to a Diameter failure (post ICSR switchover) while the network is still converging.
  
  * **damping-period** – configures a delay time to trigger an ICSR switchover due to a monitoring failure within the guard-period.
  
  * **guard-period** – configures the local-failure-recovery network-convergence timer.

- **srp-redundancy-timers** – sets timers that prevent an ICSR switchover while the system is recovering from a local card-reboot/critical-task-restart failure.
  
  * **aaa** – local failure followed by AAA monitoring failure
  
  * **bgp** – local failure followed by BGP monitoring failure
  
  * **diam** – local failure followed by Diameter monitoring failure
DSCP Marking of SRP Messages

You can enable separate DSCP marking of SRP control and checkpoint messages. The `dscp-marking` command sets DSCP marking values for SRP control and checkpoint (session maintenance) messages.

```
classcontextcontext_name
service-redundancy-protocol
dscp-marking{control|session}dscp_value
```

Notes:
- `dscp_value` can be:
  - `af11` – Assured Forwarding Class 1 low drop PHB (Per Hop Behavior)
  - `af12` – Assured Forwarding Class 1 medium drop PHB
  - `af13` – Assured Forwarding Class 1 high drop PHB
  - `af21` – Assured Forwarding Class 2 low drop PHB
  - `af22` – Assured Forwarding Class 2 medium drop PHB
  - `af23` – Assured Forwarding Class 2 high drop PHB
  - `af31` – Assured Forwarding Class 3 low drop PHB
  - `af32` – Assured Forwarding Class 3 medium drop PHB
  - `af33` – Assured Forwarding Class 3 high drop PHB
  - `af41` – Assured Forwarding Class 4 low drop PHB
  - `af42` – Assured Forwarding Class 4 medium drop PHB
  - `af43` – Assured Forwarding Class 4 high drop PHB
  - `be` – Best effort Per-Hop-Behaviour (default)
  - `cs1` – Class selector 1 PHB
  - `cs2` – Class selector 2 PHB
  - `cs3` – Class selector 3 PHB
  - `cs4` – Class selector 4 PHB
  - `cs5` – Class selector 5 PHB
  - `cs6` – Class selector 6 PHB
  - `cs7` – Class selector 7 PHB
  - `ef` – Expedited Forwarding PHB, for low latency traffic

Optimizing Switchover Transitions

There are several SRP configuration options that reduce the transition time from the active to standby gateways (primarily P-GW) in support of VoLTE traffic.
These features require an updated ICSR license to support the enhancements. Contact your Cisco account representative for additional information.

### Allow Non-VoLTE Traffic During ICSR Switchover

The ICSR framework reduces switchover disruption for VoLTE traffic by enabling VoLTE traffic on the newly active gateway prior to reconciling the billing information and enabling communication with the newly active gateway when accounting is not deemed critical.

This functionality extends to all other traffic, including data sessions and default bearer traffic for IMS/e911. The following ICSR functionality is provided for all non-VoLTE data traffic:

- When a switchover occurs, the newly active gateway forwards all traffic the moment the gateway becomes active.
- External communication with billing servers is deferred. See the traffic flow diagram below.
• When the newly active gateway receives all billing-related checkpointing information from the previously active gateway, it reconciles the billing data before communicating with external billing servers OCS (Online Charging System) or OFCS (Offline Charging System).

**Figure 27: Call Flow: Reduce Non-VoLTE Data Outage**

The **switchover allow-all-data-traffic** SRP Configuration mode CLI command allows all data traffic (VoLTE and non-VoLTE) during switchover transition. This command overwrites the **switchover allow-volte-data-traffic** command if enabled on a P-GW.

```
configure
c   context context_name
   service-redundancy-protocol
   switchover allow-all-data-traffic
```

**Important** The **switchover allow-all-data-traffic** command must be run on both chassis to enable this feature.

The **switchover allow-volte-data-traffic** SRP Configuration mode CLI command allows VoLTE data traffic during ICSR switchover transition.

```
configure
c   context context_name
```

service-redundancy-protocol
switchover allow-volte-data-traffic | maintain-accounting |

Notes:

- When maintain-accounting is enabled, accounting accuracy is maintained for VoLTE calls. VoLTE data is allowed on the active gateway after VoLTE accounting statistics are flushed.

**Allow All Data Traffic**

The SRP Configuration mode switchover allow-all-data-traffic command allows all data traffic (VoLTE and non-VoLTE) during switchover transition. This command overwrites the switchover allow-volte-data-traffic command if enabled on a P-GW. This feature reduces data traffic outage during the switchover.

Important
This CLI command must be run on both the active and standby chassis to enable this feature.

All data traffic is allowed on the active chassis during flushing and internal auditing. The billing information is reconciled in the background once the flush is complete.

**Graceful Cleanup of ICSR After Audit of Failed Calls**

During an Audit on the gateways (P-GW/S-GW/GGSN/SAE-GW) after Session Recovery or an ICSR event, if any critical information, internally or externally related to a subscriber session seems inconsistent, ICSR will locally purge the associated session information.

Since external gateways (peer nodes) are unaware of the purging of this session, the UE session may be maintained at other nodes. This leads to hogging of resources external to the gateway and an unreachable UE for VoLTE calls.

When this feature is enabled, graceful cleanup for an ICSR audit of failed calls occurs. External signaling notifies peers of session termination before purging the session. The gateway will attempt to notify external peers of the removal of the session. External nodes to the local gateway include S-GW, P-GW, SGSN, MME, AAA, PCRF and IMSA.

Audit failure can occur because of missing or incomplete session information. Therefore, only the peers for which the information is available will be notified.

The require graceful-cleanup-during-audit-failure Global Configuration mode CLI command enables or disables the graceful cleanup feature.

configure
require graceful-cleanup-during-audit-failure | del-cause non-ims-apn { system-failure | none } }

**Optimization of Switchover Control Outage Time**

The ICSR framework minimizes control outage time associated with the flushing of critical full checkpoint statistics, network convergence and internal auditing.

The amount of time consumed by the following activities affects control outage time during switchover:

- **Critical Flush** – During the Active to Pending-Standby transition, all sessmgrs flush any pending critical FCs (Full Checkpoints). During this time, the active chassis drops all control packets. If control signaling
is allowed during this stage, a call may get disconnected based on the control message type and accounting information will be lost.

• **Network Convergence** – This encompasses the amount of time taken to update routes and send control/data to the newly active chassis. Control messages are dropped during the transition.

• **Accounting Flush** – During this flush stage data counts are synchronized between chassis. If control signaling is allowed during this flush, the call may get disconnected based on the control message type, and accounting information will be lost for calls that existed before switchover.

• **Audit** – During audit new calls are not allowed because synchronization of call resources may result in clearing of the calls.

The switchover control-outage-optimization CLI command allows new calls during the Accounting Flush, as soon as the Audit is completed. This SRP Configuration mode command enables the quicker restoration of control traffic (call-setup, modification, deletion) following an ICSR switchover.

```plaintext
configure
  context context_name
    service-redundancy-protocol
      switchover control-outage-optimization
  end
```

### Configuring the SRP Context Interface Parameters

This procedure configures the communication interface with the IP address and port number within the SRP context. This interface supports interchassis communication.

| Important | CLI commands must be executed on both chassis. Log onto both chassis before continuing. Always make configuration changes on the primary chassis first. |

```plaintext
configure
  context vpn_ctxt_name [-noconfirm]
    interface srp_if_name
      ip-address { ip_address | ip_address/mask }
    exit
  exit
  port ethernet slot_num/port_num
    description des_string
    medium { auto | speed { 10 | 100 | 1000 } duplex { full | half } }
    no shutdown
    bind interface srp_if_name vpn_ctxt_name
  end
```

### Configuring NACK Generation for SRP Checkpoint Messaging Failures

#### Enabling NACK Messaging from the Standby Chassis

Transport (TCP) level re-transmission is supported on the SRP link between ICSR chassis. SRP configuration also supports optional application level checks to ensure checkpoints are received at the Standby chassis. Failed attempts to receive and apply checkpoints send NACK messages to the Active chassis.
When this feature is enabled and the standby chassis sends a NACK in response to the receipt of a micro-checkpoint (MC) that fails to be successfully applied, the standby chassis sends another NACK. The standby chassis will send more NACKs (configurable, default = 3) within a 10-minute window if a macro-checkpoint (FC) is not received. NACKs will continue to be sent and the 10-minute reset until an FC is received and applied, or the configured number of max-responses is reached.

You can also specify the number of times a NACK is sent within the 10-minute window in response to a failed MC or FC (Default = 3).

A **nack** keyword in the SRP Configuration mode **checkpoint session** command allows you to enable generation of NACK messages in response to checkpoint message failures on a Standby ICSR chassis.

### Important

The **nack** keyword will only appear if a special ICSR optimization feature license has been purchased and installed. Contact your Cisco account representative for assistance.

```bash
configure
context context_name
  service-redundancy-protocol variable
    checkpoint session nack { macro | micro } [ max-response number ]
  no checkpoint session nack { macro | micro }
end
```

**Notes:**

- **max-response** is the number of times a NACK is sent within the 10-minute window in response to a failed MC or FC expressed as an integer from 0 through 65535 (Default = 3).

A **periodic-interval** keyword in the SRP Configuration mode **checkpoint session** command allows you to configure the interval between the sending of macro-checkpoints (FCs) between the active and standby chassis.

### Important

The **periodic-interval** keyword will only appear if a special ICSR optimization feature license has been purchased and installed. Contact your Cisco account representative for assistance.

```bash
configure
context context_name
  service-redundancy-protocol variable
    checkpoint session periodic-interval minutes
  default checkpoint session periodic-interval
  no checkpoint session periodic-interval
end
```

### Selective Disabling of NACK Messaging

The NACK mechanism sends a NACK message for any ICSR checkpoint failure on the standby chassis. Every NACK sent from the standby chassis triggers a full checkpoint from the active chassis.

If the micro-checkpoint is failing continuously and sending NACKs, the active chassis keeps sending full-checkpoints. This increases SRP bandwidth.

CLI commands allow an operator to selectively disable and re-enable NACK messages for specific micro-checkpoints.
The Exec mode `srp disable nack micro-chkpt-cmd` disables the sending of a NACK from the standby chassis.

`srp disable nack micro-chkpt-cmd chkpt_number`

*chkpt_number* specifies the checkpoint number to be disabled as an integer from 1 through 255. You can obtain checkpoint numbers (CMD IDs) from the output of the `show srp checkpoint info` command.

You can re-enable the micro-checkpoint using the `srp enable nack micro-chkpt-cmd` command.

`srp enable nack micro-chkpt-cmd chkpt_number`

---

**Configuring LZ4 Compression Algorithm**

You can optionally enable LZ4 compression algorithm for SRP messaging payload. The zlib algorithm remains as the default.

LZ4 is a very fast lossless compression algorithm with near-linear scalability for multi-threaded applications. The `compression` keyword in the SRP Configuration mode `checkpoint session` command allows you to enable the use of the LZ4 compression algorithm.

---

**Important**

The `compression` keyword will only appear if a special ICSR optimization feature license has been purchased and installed. Contact your Cisco account representative for assistance.

The following command sequence enables the use of LZ4 compression:

```
configure
  context context_name
  service-redundancy-protocol
    checkpoint session compression lz4
  end
```

LZ4 compression is effective only if both chassis are configured with LZ4. If any one chassis has zlib (default) configured, the compression algorithm reverts to zlib. The algorithm is negotiated only during initial socket establishment. Once agreed no more negotiation takes place until the TCP socket connection is reset.

---

**Reducing Sync-Up Time with Standby ICSR Chassis**

The default method for synchronizing the SRP database requires tens of seconds of delay whenever the TCP connection between the Active and Standby session managers is established. Once the TCP connection is established, heart beat messages are exchanged between both ICSR chassis every 3 seconds. The standby chassis waits for seven heartbeat messages from the active chassis before it is ready to accept data. This may cause significant delay in session manager database synchronization on the standby chassis.

You can enable an aggressive method for synchronizing the session manager database reduces recovery time in the following scenarios:

- Standby Session Manager crash
- Packet processing card failure on Standby chassis
- Standby chassis reboot
- Temporary loss and recovery of SRP connection
The aggressive method reduces the number of heartbeat messages and amount of housekeeping information exchanged between ICSR chassis.

The SRP Configuration mode `standby database-recovery aggressive` command allows you to select normal or aggressive restoration of the SRP database.

The following command sequence enables the aggressive recovery mode:

```bash
configure
text context_name
  service-redundancy-protocol
  standby database-recovery aggressive
end
```

The default form of this command restores the normal mode of SRP database recovery.

**Verifying SRP Configuration**

Verify that your SRP contexts were created and configured properly by running the `show srp info` command (Exec Mode) on each chassis.

Notes:

- The interval is specified as an integer divisible by 15 in the range from 30 through 1440 (Default = 45 minutes). The interval range for sending full checkpoints is 30 minutes to 24 hours (1140 minutes).

**Modifying the Source Context for ICSR**

To modify the source context of core service:

---

**Step 1** Add the Border Gateway Protocol (BGP) router AS-path and configure the gateway IP address, neighbor IP address, remote IP address in the source context where the core network service is configured, by applying the example configuration in Configuring BGP Router and Gateway Address, on page 300.

**Step 2** Configure the service redundancy context with the BGP neighbor context and IP address to monitor the BGP link activity by applying the example configuration in Configuring the SRP Context for BGP, on page 301.

**Step 3** Verify your BGP context configuration by following the steps in Verifying BGP Configuration, on page 301.

**Step 4** Save your configuration as described in Verifying and Saving Your Configuration.

---

**Configuring BGP Router and Gateway Address**

Use the following example to create the BGP context and network addresses.

```bash
configure
  context source_ctxt_name
  router bgp AS_num
    network gw_ip_address
    neighbor neighbor_ip_address remote-as AS_num
end
```
Notes:

• *source_ctxt_name* is the context where the core network service is configured.

### Configuring the SRP Context for BGP

Use the following example to configure the BGP context and IP addresses in the SRP context.

```
configure
c   context srp_ctxt_name
   service-redundancy-protocol
   monitor bgp context source_ctxt_name neighbor_ip_address
end
```

*neighbor_ip_address* can be entered in IPv4 dotted-decimal or IPv6 colon-separated-hexadecimal notation. Multiple IP addresses can be added per context as IPv4 and/or IPv6 IP addresses.

An ICSR failover is triggered when all BGP peers within a context are down.

Optionally, you can configure SRP peer groups within a context. ICSR failover would then occur if all peers within a group fail. This option is useful in deployments in which a combination of IPv4 and IPv6 peers are spread across multiple paired VLANs, and IPv4 or IPv6 connectivity is lost by all members of a peer group.

A sample configuration for SRP peer groups within a context ("PGWin") appears below.

```
monitor bgp context PGWin 10.1.1.16 group 1
monitor bgp context PGWin 10.1.1.17 group 1
monitor bgp context PGWin 69.2.215.0 group 2
monitor bgp context PGWin 69.2.215.1 group 2
monitor bgp context PGWin 2001:4333:201:1102:103:2a1:: group 3
monitor bgp context PGWin 2001:4333:201:1102:103:2a1:0:1 group 3
```

In the above sample configuration, ICSR failover would occur if both addresses in group 1, 2 or 3 lost connectivity.

For additional information refer to the description of the *monitor bgp*, *monitor diameter* and *monitor authentication-probe* commands in the *Service Redundancy Protocol Configuration Mode Commands* chapter of the *Command Line Interface Reference*.

### Verifying BGP Configuration

Verify your BGP configuration by entering the *show srp monitor bgp* command (Exec Mode).
Modifying the Destination Context for ICSR

To modify the destination context of core service:

**Step 1** Add the BGP router and configure the gateway IP address, neighbor IP address, remote IP address in the destination context where the core network service is configured, by applying the example configuration in Configuring BGP Router and Gateway Address in Destination Context, on page 302.

**Step 2** Configure the service redundancy context with BGP neighbor context and IP address to monitor the BGP link activity by applying the example configuration in Configuring SRP Context for BGP for Destination Context, on page 302.

**Step 3** Set the subscriber mode to **default** by following the steps in Setting Subscriber to Default Mode, on page 302.

**Step 4** Verify your BGP context configuration by following the steps in Verifying BGP Configuration in Destination Context, on page 303.

**Step 5** Save your configuration as described in Verifying and Saving Your Configuration.

---

**Configuring BGP Router and Gateway Address in Destination Context**

Use the following example to create the BGP context and network addresses.

```plaintext
configure
c   context dest_ctxt_name
     router bgp AS_num
     network gw_ip_address
     neighbor neighbor_ip_address remote-as AS_num
end
```

**Notes:**
- **AS_num** is the autonomous systems path number for this BGP router.

**Configuring SRP Context for BGP for Destination Context**

Use the following example to configure the BGP context and IP addresses in the SRP context.

```plaintext
configure
c   context srp_ctxt_name
      service-redundancy-protocol
      monitor bgp context dest_ctxt_name neighbor_ip_address
end
```

**Setting Subscriber to Default Mode**

Use the following example to set the subscriber mode to **default**.

```plaintext
configure
c   context dest_ctxt_name
      subscriber default
end
```
Verifying BGP Configuration in Destination Context

Verify your BGP configuration by entering the show srp monitor bgp command (Exec Mode).

Disabling Bulk Statistics Collection on a Standby System

You can disable the collection of bulk statistics from a system when it is in the standby mode of operation.

**Important**

When this feature is enabled and a system transitions to standby state, any pending accumulated statistical data is transferred at the first opportunity. After that no additional statistics gathering takes place until the system comes out of standby state.

Use the following example to disable the bulk statistics collection on a standby system.

```
configure
bulkstat mode
no gather-on-standby
end
```

Repeat this procedure for both systems.

Verifying the Primary and Backup Chassis Configuration

This section describes how to compare the ICSR configuration on both chassis.

**Step 1**

Enter the show configuration srp command on both chassis (Exec mode).

**Step 2**

Verify that both chassis have the same SRP configuration information.

The output looks similar to following:

```
config
context source
  interface haservice loopback
    ip address 172.17.1.1 255.255.255.255 srp-activate
#exit
  radius attribute nas-ip-address address 172.17.1.1
  radius server 192.168.83.2 encrypted key 01abd002c82b4a2c port 1812
  radius accounting server 192.168.83.2 encrypted key 01abd002c82b4a2c port 1813
  ha-service ha-pdsn
    mn-ha-spi spi-number 256 encrypted secret 6c93f7960b726b6f6c93f7960b726b6f hash-algorithm md5
    fa-ha-spi remote-address 192.168.82.0/24 spi-number 256 encrypted secret 1088bdd6817f64df
    bind address 172.17.1.1
  #exit
#exit
context destination
  ip pool dynamic 172.18.0.0 255.255.0.0 public 0 srp-activate
  ip pool static 172.19.0.0 255.255.240.0 static srp-activate
#exit
context srp
  service-redundancy-protocol
```
Configuring Subscriber State Management Audit Process

This audit is to ensure that two ICSR peers are in synch and identifies any discrepancies prior to any scheduled or unscheduled switchover events.

**Step 1**
Enter the SRP Context mode and enter the `service-redundancy-protocol` command.

**Step 2**
Enter the `audit daily-start-time` command. Specify the daily start time as an hour and minute. For example, a start time of 06 00 indicates that the audit will begin at 6:00 AM.

**Step 3**
Enter the `audit periodicity` command. Specify the interval in minutes for generating SRP audit statistics as an integer from 60 through 1440. For example, a periodicity of 90 indicates that SRP audit statistics will be generated every 90 minutes beginning at the specified start time. Default = 60.

A sample configuration sequence appears below.

```
config
context srp
  service-redundancy-protocol
  audit daily-start-time 06 00
  audit periodicity 90
end
```

Troubleshooting ICSR Operation

**SSD**

StarOS supports an ICSR-specific `show support details` (SSD) command that outputs the results from a series of Exec mode `show` commands. This mini SSD reduces capture time when debugging ICSR timing issues between the Active and Standby chassis, facilitating quicker resolution of the problem.

The `show support details icsr` command produces a mini SSD that contains the output of the following `show` commands:

- show srp info
- show srp checkpoint statistics
- show srp checkpoint statistics verbose
- show srp checkpoint statistics debug-info
- show srp checkpoint statistics sessmgr all
- show srp checkpoint statistics sessmgr all debug-info
The SSD output can be directed to a file that can be stored to /flash or off the chassis. For additional information, see the Command Line Interface Reference.

**show srp details**
The Exec mode *show srp details* command displays comprehensive information used by TAC personnel to troubleshoot ICSR/SRP issues.

## Updating the Operating System

Updating the operating system (StarOS™) on ICSR chassis requires performing an Off-line update of each chassis while it is standby mode. Traffic disruption is minimal since an active chassis will be handling call sessions while the standby chassis is being updated.

The general upgrade sequence is as follows:

- Download the StarOS software image and copy/transfer it to both chassis.
- Save the currently running configurations on both chassis.
- Update the standby backup chassis first.
- Initiate an SRP switchover from the active primary chassis to make the standby backup chassis active.
- Update the standby primary chassis.
- Initiate an SRP switchover from the active backup chassis to make the standby primary chassis active.

The four-part flowchart below shows a more complete view of all the procedures required to complete the StarOS upgrade process.
Enabling the Demux on MIO/UMIO feature changes resource allocations within the system. This directly impacts an upgrade or downgrade between StarOS versions in ICSR configurations. Contact Cisco TAC for procedural assistance prior to upgrading or downgrading your ICSR deployment.

Figure 28: ICSR Software Upgrade – Part 1
Figure 29: ICSR Software Upgrade – Part 2
**Figure 30: ICSR Software Upgrade — Part 3**
Interchassis Session Recovery

Figure 31: ICSR Software Upgrade – Part 4
Both ICSR Chassis

Log into the CLI of the primary and backup and perform the tasks described below.

**Downloading and Transferring the StarOS Build**

**Step 1** Verify that there is enough free space on the /flash device to accommodate the new operating system image file by entering the following Exec mode command:

```
[local] host_name# directory /flash
```

**Step 2** Access to the Cisco support site and download facility is username and password controlled. Download the software image to a network location or physical device (USB stick) from which it can be uploaded to the /flash device.

**Step 3** Transfer the new operating system image file to the /flash device on the MIO/UMIO using one of the following methods:
a) Copy the file from a network location or local device plugged into the MIO/UMIO using the `copy` command.

```
[local] host_name# copy from_url to_url [-noconfirm]
```

b) Transfer the file to the `/flash` device using an FTP client with access to the system. The FTP client must be configured to transfer the file using binary mode.

c) Transfer the file to the `/flash` device using an SFTP client with access to the system.

In release 20.0 and higher Trusted StarOS builds, FTP is not supported.

---

**Step 4**

Verify that the image file was successfully transferred to the `/flash` device by running the Exec mode the following command:

```
[local] host_name# directory /flash
```

**Step 5**

Run the `show version /flash/image_filename` command to verify the build information. Any CRC errors will be displayed in the output of this command. If any errors appear, check the build and re-transfer it onto the chassis. Confirm that the correct image version and build description is displayed.

---

**Standby Backup Chassis**

Log into the backup standby chassis and perform the tasks described below.

**Performing Health Checks**

Health checks are a series of Exec mode `show` commands to determine the readiness of the system to handle a software update.

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td>`show card table all</td>
<td>grep unknown`</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td>`show card table</td>
<td>grep offline`</td>
</tr>
<tr>
<td><strong>Step 3</strong></td>
<td>`show resources</td>
<td>grep Status`</td>
</tr>
<tr>
<td><strong>Step 4</strong></td>
<td><code>show alarm outstanding</code></td>
<td>Review the output for any issues that may preclude performing the software update.</td>
</tr>
</tbody>
</table>

---

**Performing SRP Checks**

Service Redundancy Protocol (SRP) checks verify that the mechanism for monitoring ICSR system status is operational.

<table>
<thead>
<tr>
<th>Step</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td><code>show srp monitor all</code></td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
<td>Review the output for any issues that may preclude performing the software update.</td>
</tr>
</tbody>
</table>
Performing BGP Checks

Border Gateway Protocol (BGP) checks are only required when BGP is used to support redundant interchassis communication. These checks are run per context and per service type.

**Step 1**  
For each BGP-enabled context, run `show ip bgp summary`. Verify that the BGP peers are connected and that IPv4 and IPv6 peers are up. Repeat for all BGP-enabled contexts.

**Step 2**  
Run `show service_name all | grep "Service Status:"`. The service should be "Started". Repeat for all services running on the chassis.

Updating the Boot Record

You must add a new boot stack entry for the recently downloaded software image (.bin) file.

**Step 1**  
Run the Exec mode `show boot` command to verify that there are less than 10 entries in the boot.sys file and that a higher priority entry is available (minimally there is no priority 1 entry in the boot stack).

**Step 2**  
Create a new boot stack entry for the new file group, consisting of the new operating system image file and the currently used CLI configuration file by entering the following Global Configuration command:

```
(host_name) (conf) 
boot system priority number image image_url /flash/filename
filesystem
```

**Step 3**  
Assign the next highest priority to this entry, by using the <N-1> method, wherein you assign a priority number that is one number less than your current highest priority.  
If priority 1 is in use, you must renumber the existing entries to ensure that at least that priority is available. The maximum number of boot stack entries that can be contained in the boot.sys file is 10. If there are already 10 entries in the boot stack, you must delete at least one of these entries (typically, the lowest priority) and, if necessary, renumber some or all of the other entries before proceeding. Use the `no boot system priority` command to delete a boot stack entry.

For information on using the `boot system priority` command, refer to the Adding a New Boot Stack Entry section in this guide.

Synchronizing File Systems

Synchronize the local file systems by entering the following Exec mode command:

```
(host_name)\# filesystem synchronize all
```

Reloading the Chassis

Reboot the chassis by entering the following command:

```
(host_name)\# reload [-noconfirm]
```
As the system reboots, it loads the new operating system software image and its corresponding CLI configuration file using the new boot stack entry configured earlier.

After the system reboots, establish a CLI session and enter the `show version` command to verify that the active software version is correct.

Optional for PDSN: If you are using the IP Pool Sharing Protocol during your upgrade, refer to Configuring IPSP Before the Software Upgrade in the PDSN Administration Guide.

**Updating the Configuration File**

Features in the new operating system may require changes to the configuration file. These changes can be done manually or facilitated by custom scripts prepared by Cisco TAC. Make whatever changes are necessary prior to saving the updated configuration file.

**Verifying the Software Version**

After the system has successfully booted, verify that the new StarOS version is running by executing the Exec mode `show version` command.

**Saving the Configuration File**

Use the Exec mode `save configuration` command to save the currently running configuration to the `/flash` device and to an off-chassis location (external memory device or network URL). The off-chassis copy assures that you will have a fallback, loadable configuration file should a problem be encountered.

**Completing the Update Process**

Repeat the following tasks to complete the upgrade process on the standby secondary chassis:

- Synchronizing File Systems, on page 312
- Performing Health Checks, on page 311
- Performing SRP Checks, on page 311
- Performing BGP Checks, on page 312
Waiting for Session Synchronization

Allow time for session synchronization to occur between the ICSR chassis before proceeding to the next steps.

**Step 1**
Run the `show session recovery status verbose` command on both chassis. Proceed to the next steps only when no errors are seen in the output of this command.

**Step 2**
On the standby chassis, run `show srp checkpoint statistics |more`.

**Step 3**
On active chassis, run `show subs summary |grep Total`.

**Step 4**
Compare the number of subscribers on the active chassis and the number of Current pre-allocated calls: on the standby chassis. They should be similar (within 5%). Allow a few minutes for systems to complete synchronization.

---

**Primary Chassis**

Log into the active primary chassis and complete the tasks described below.

**Initiating an SRP Switchover**

An SRP switchover places the primary chassis in standby mode and makes the backup chassis active. The secondary chassis is now processing sessions with the upgraded software.

**Step 1**
On the primary chassis run the `srp initiate-switchover` command. All existing sessions will be migrated to the backup chassis and it begins servicing new session requests. Allow the switchover process to complete.

**Step 2**
On the primary chassis, run the `show srp info` command. Chassis State should indicate Standby when switchover is complete.

**Step 3**
On the backup chassis, confirm the switchover is complete by running the `show srp info` command. Chassis State should indicate Active when switchover is complete.

---

**Checking AAA Monitor Status on the Newly Active Chassis**

If your network deployment requires communication with AAA servers, log into the newly active chassis and perform an AAA monitor check. You will be checking for the existence of any SNMP traps that indicate the chassis cannot communicate with AAA servers (`starSRPAAAUnreachable`).

**Step 1**
Run the Exec mode command `show snmp trap history |grep starSRPAAAUnreachable`.

**Step 2**
There should be no output for this command, or no very recent SNMP trap notifications (based on the event timestamp).

**Step 3**
If the active chassis cannot communicate with one or more AAA servers, refer to AAA Monitor for additional information.
Completing the Software Update

Log into the standby chassis and repeat the following tasks to complete the upgrade process on the standby primary chassis:

- Updating the Boot Record, on page 312
- Reloading the Chassis, on page 312
- Updating the Configuration File, on page 313
- Verifying the Software Version, on page 313
- Saving the Configuration File, on page 313
- Synchronizing File Systems, on page 312
- Performing Health Checks, on page 311
- Performing SRP Checks, on page 311
- Performing BGP Checks, on page 312
- Waiting for Session Synchronization, on page 314

Initiating an SRP Switchover

An SRP switchover places the primary chassis in active mode and makes the backup chassis active. The primary chassis is now processing sessions with the upgraded software.

Step 1
On the backup chassis run the `srp initiate-switchover` command. All existing sessions will be migrated to the primary chassis which begins servicing new session requests. Allow the switchover process to complete.

Step 2
On the backup chassis, run the `show srp info` command. Chassis State should indicate Standby when switchover is complete.

Step 3
On the primary chassis, confirm the switchover is complete by running the `show srp info` command. Chassis State should indicate Active when switchover is complete.

Making Test Calls

Once the chassis state is verified and subscribers are migrated, perform new call testing to make sure calls are successful.
Fallback Procedure

To revert to the previous configuration and software build, perform the following steps as a user with administrative privileges.

**Step 1**
Run the Exec mode `show boot` command. The topmost lowest numbered entry of the displayed output should be the new configuration with the new software build. The second topmost entry should be the backup configuration.

**Step 2**
Remove the topmost boot entry \( n \), and synchronize the configuration across the management cards.

```
[local] host_name# config
[local] host_name(config)# no boot system priority n
[local] host_name(config)# end
[local] host_name# filesystem synchronize all
```

**Step 3**
Reboot the system to load its previous configuration.

```
[local] host_name# reload
```

**Step 4**
Perform health checks as described in Performing Health Checks, on page 311.
Support Data Collector

The Support Data Collector (SDC) is a system feature that allows scheduled collection of process state, counter, event and attribute data that may be useful when troubleshooting problems at an installation site.

This chapter includes the following sections:

- Overview, page 317
- Configuring SDR Collection, page 318
- Displaying the SDR Collection Configuration, page 318
- Collecting and Storing the SDR Information, page 319
- Managing Record Collection, page 319
- Using SDRs to Diagnose Problems, page 321
- SDR CLI Commands, page 321

Overview

The task of collecting the support data is performed by a background CLI task called the record collector. The administrator configures the SDC via the CLI with the commands to be executed on a periodic basis. The record collector always runs in the background and checks if there are records to be collected.

When it is time to collect support data, the scheduler executes the configured sequence of CLI commands and stores the results in a gunzipped (.gz) file on the hard-disk. This file is called an SDR (Support Data Record), and represents a snapshot of the overall state of the system at that time.

Technical Assistance Center (TAC) personnel and local administrators can review the SDRs on-line or by transferring them off the system. They may also wish to investigate the collector state information. The figure
below shows system tasks that contain state and counter information. Arrows between tasks and processes represent messenger requests and indicate the predominant flow of data.

Figure 32: SDC Tasks and Processes<

Configuring SDR Collection

The Support Data Record (SDR) is an ordered set of the CLI support commands' display output that is stored in a stand-alone compressed file. Each CLI support command output is stored in its own record section. The record section is identified by a record section name and its ASCII command syntax. For example, the record section `show_version` would have a CLI command string of "show version".

The order in which the record section commands appear in the configuration is significant. All of the support record section commands must be configured together as an ordered set. In other words, just specifying one command by itself will result in just that one command output constituting the contents of the entire SDR.

The user may configure a specific set of record sections for the SDR which may or may not include some or all of the default SDR record sections. This configuration is stored in the Global Configuration section of the configuration file. Refer to Configuration Commands (Global Configuration Mode), on page 322 for more detail on the `support record section` command.

Displaying the SDR Collection Configuration

The `show configuration verbose` command displays the default support record sections, if the user has not specified any support record sections. If the user has configured support record sections, then the `show configuration` command displays user-configured support record sections. The support collection schedule configuration also appears in the `show configuration` output under the Global Configuration section.
Collecting and Storing the SDR Information

At the scheduled time, the Support Data Collector (SDC), if active, runs in the background to collect all the record section commands that have been specified. This information is concatenated as one contiguous output. The output is compressed and stored as a file on disk in the `/hd-raid/support/record/` directory.

The periodicity of the SDC is configured by the `support collection schedule` command under Global Configuration Mode. Once the SDR is stored, the SDC waits the sleep-duration interval specified via the `support collection` command before collecting another SDR.

---

**Important**

The period between SDRs is equal to the configured sleep-duration interval + the time taken to collect the previous record.

---

Managing Record Collection

The SDRs are stored together in a self-relative set. This self-relative set is called a Support Record Collection. Each individual SDR is identified with a record-id. The record-id of the most recent SDR is always 0 (zero).
The next older SDR is record-id 1, and so on, for the number of records in the stored collection. For example, if there are five SDRs, they are identified as SDR-0 through SDR-4.

**Figure 33: Support Data Collection Hierarchy**

When a new SDR is created, the numbers all increment by one and the newest SDR is given the value of 0. If the total number of records exceeds a configured maximum, then the oldest SDR is deleted.

Using the example above, when the maximum SDR count of 5 is reached, the SDRs continue to be SDR-0 through SDR-4, with the file timestamps indicating that the files are changing over time.

The time interval between collections may vary by several minutes in relation to the specified sleep-duration. This is because the interval specifies the idle time between scheduled collection runs. Since the actual overhead of the collecting process is not included in the scheduled intervals, the time differences between collections includes this non-deterministic amount of time.
Using a shorter interval to compensate for this behavior is not recommended, since it will only add to the overhead incurred by the collection process and will ultimately impact the overall system performance. The sleep-duration (idle-time) between scheduled collections is an important component of the "self-throttling" mechanism that should not be circumvented by the user.

The Exec Mode show support collection command displays useful information about the Support Data Collector. The output includes information about when the collector last ran, how long it took to run, when it is scheduled to run again, as well as the number of SDRs currently stored, where they are stored, and how much storage space is being used. Refer to Exec Mode Commands, on page 323 for more detail about this command.

Using SDRs to Diagnose Problems

The user can compare the SDRs by examining two or more in sequence. These SDRs are dumped out in their CLI-formatted output display. Comparing the display outputs reveals trends and performance or configuration differences that indicate problem areas.

Once specific record sections have been identified as having problematic characteristics, only the CLI show commands associated with those sections need be monitored and compared to further isolate the problem areas. In addition, individual SDRs may be transferred via system-supported protocols to remote system, or the current collection may be transferred as a set for later analysis.

SDR CLI Commands

You may use the collected support data records to view support data chronologically. If the default list and sequence of sections is inadequate for system monitoring, you can configure your own set of record section commands that make up a particular support record.

Important Refer to the SDR CLI Command Strings appendix for a listing of supported CLI strings (show commands) for record sections. The listing also identifies the CLI strings supported as default record sections. You can obtain the same listing by running the show support collection definitions command.

Important You may enter up to 200 SDR CLI strings in a single record section command. If you attempt to add more than 200 CLI strings, an error message appears. You may also receive an error message if the system is unable to parse all of the requested CLI strings because they are too complicated to parse.

After configuring the SDR you then configure the sleep-duration interval between record collections and the number of historical records to be retained before being overwritten. By default, configuring this collection information makes the collector mechanism active (if not already active).

After one or more collection intervals have passed, the SDR data becomes available for analysis. The administrator can then use CLI commands to examine the SDR information to perform root cause analysis and trend analysis based on how the data has changed over time. The administrator may decide to transfer the SDRs off the system to be analyzed remotely, for example, by Cisco TAC.
For complete descriptions of the CLI commands discussed below, refer to the *Command Line Interface Reference*.

**Configuration Commands (Global Configuration Mode)**

### support record

**support record section**  
`section-name`  
`command"command-string"`  
`[section section-name command"command-string" ...  
no support record | all | section section-name |  
default support record | all | section section-name |  

The **support record section** command configures a specific record section or set of record sections for a support information output command. The order in which record sections are saved is fixed, regardless of the sequence in which the CLI commands were entered.

For example:

```
[local]host_name(config)# support record section show_context command"show context"
```

If the **support record section** command is not explicitly configured by the user, a default set of record section commands are used. These default record section commands are displayed when you run the **show configuration verbose** command. If support record section commands are explicitly configured, they replace the default commands.

---

**Important**  
Refer to the *SDR CLI Command Strings* appendix for a listing of supported CLI strings (**show** commands) for record sections. The listing also identifies the CLI strings included in default record sections.

The **no support record** command removes either a specific section of the record definition or all of the sections. If you specify the **default support record** command, the default record section definition of that specified record section is used. If neither the keyword **all** or **section** is specified, all the record section definitions are removed.

### support collection

**support collection**  
`sleep-duration`  
`hours h | minutes m | | max-records n |  
no support collection  
default support collection  

The **support collection** command modifies and/or enables the support collection process. If support collection has been previously disabled, this command enables the collection activity. If the support collection is currently enabled, this command may be used to modify the sleep-duration interval and/or the maximum number of SDRs that can be collected and stored.

The **sleep duration** keyword specifies the time interval between the collection of support data. It can be specified in hours or minutes with a default of one hour (60 minutes).

The **max-records** keyword specifies the number of SDRs to store as an integer from 1 to 65535. When this value is exceeded, the new SDR overwrites the oldest SDR. The default value is 168.
SDR files will be stored in the `/hd-raid/support/records/` directory.

For example:

```
[local] host_name(config)# support collection sleep-duration minute 30 max-records 50
```

Use the `no support collection` command to explicitly disable the collection of the SDRs. If no record section commands are defined, the support data collector mechanism is also effectively disabled.

Use the `default support collection` command to enable the support data collector using the default record sections.

### Exec Mode Commands

#### show support record

```
show support record  record-id [ to record-id ] [ section section_name ]
```

The `show support record` command displays a collection of SDRs. The SDRs are displayed in order from lowest record-id to highest record-id.

Each SDR is identified by a time index called the record-id. For example, the most recent record is always record-id 0 (filename = sdr.0.gz). The next older record is record-id 1 (filename = sdr.1.gz), and so on.

When a new record is collected it is given a record-id of 0. The previously most recent record is renamed to record-id 1, and so on. The display includes the record-id along with the collection time-stamp.

The `record-id` variable identifies a single SDR. The `to` keyword specifies the endpoint record-id when displaying a range of SDRs.

The `section` keyword displays a particular section of the record.

#### delete support record

```
delete support record  record-id [ to record-id ]
```

The `delete support records` command removes an SDR with a specified record-id or all SDRs in the specified range of record-ids.

#### show support collection

```
show support collection [ definitions ]
```

The `show support collection` command displays information on SDC activity. It display informations such as the start time of the last scheduled collection, the duration of the last scheduled collection, whether the collection is still in progress, etc. In addition this command lists the currently stored set of SDR record-ids, their respective timestamps, and size of each SDR.

```
[local] host_name# show support collection
Record Collection Enabled : yes
Last Collection Start Time : Monday October 21 06:29:05 PDT 2013
Last Collection End Time : Monday October 21 06:29:09 PDT 2013
Est. Collection Next Start : Monday October 21 07:29:13 PDT 2013 (40 minutes)

Support Data Records at /var/tmp/support-records/
```
The optional definitions keyword displays the list of default support record section definitions. This is the list of all valid record section definitions. The display also indicates whether the record section is enabled or disabled by default.

```plaintext
[local] host_name# show support collection definitions
```

The output of this command reflects the sequence in which record sections will be output, regardless of the sequence in which they may have been entered by the user. Refer to the SDR CLI Command Strings appendix for additional information.
Engineering Rules

This appendix provides engineering guidelines for configuring the system to meet network deployment requirements.

- CLI Session Rules, page 325
- ASR 5500 Interface and Port Rules, page 325
- Context Rules, page 326
- Subscriber Rules, page 329
- Service Rules, page 329
- Access Control List (ACL) Engineering Rules, page 330
- ECMP Groups, page 330

CLI Session Rules

Multiple CLI session support is based on the amount of available memory. The Resource Manager reserves enough resources to support a minimum of six CLI sessions at all times. One of the six sessions is further reserved for use exclusively by a CLI session on an MIO/UMIO serial interface.

Additional CLI sessions beyond the pre-reserved limit are permitted if sufficient MIO/UMIO resources are available. If the Resource Manager is unable to reserve resources for a CLI session beyond those that are pre-reserved, users with administrator-privileges are prompted to create the new CLI session, even without reserved resources.

ASR 5500 Interface and Port Rules

The rules discussed in this section pertain to the Ethernet ports used for subscriber traffic on the MIO/UMIO card (ports 10 through 29).

- Give all logical interfaces a unique name to identify the interface from others in the same context. Logical interfaces in different contexts may have the same name.
A single physical port can support multiple logical interfaces when you configure VLAN tags for that physical port. You can use VLAN tagging to bind a single physical port to multiple logical interfaces that reside in different contexts.

Assign all logical interfaces a valid IP address and subnet.

- Give each logical interface within a context a unique IP address(es). Logical interfaces in different contexts can have the same IP address(es).
- If multi-homing is supported on the network, you can assign all logical interfaces a single primary IP address and up to 16 secondary IP addresses.

You can configure a logical interface in only one context, but you can configure multiple interfaces (up to 512) in a single context.

You can apply a maximum of 256 access control list (ACL) rules to a single logical interface.

All ports are identified by their <slot#>/<port#>.

Each physical port for subscriber traffic on an MIO/UMIO card may contain up to a maximum of 1,024 VLAN tags.

A logical interface is limited to using a single VLAN on a single physical port, identified by its <card#[slot#][port#]>

**Packet Data Network (PDN) Interface Rules**

The following engineering rules apply to the interface to the packet data network (PDN):

- Configure the logical interfaces used to facilitate the PDN interface within the egress context.
- The default is to use a single interface within the egress context to facilitate the PDN interface.
- You can configure multiple interfaces in the egress context by using static routes or dynamic routing protocols.
- You may also configure next-hop default gateways.

**Context Rules**

- A maximum of 63 contexts may be configured per chassis. Enabling demux functions on an MIO card reduces the maximum number of contexts to 10.

- Interfaces per Context
  - Prior to Release 15.0: Up to 16 interfaces can be configured within a single context.
  - For Release 15.0 and higher: With the Demux MIO/UMIO feature enabled, up to 64 interfaces can be configured within a single context.
  - 512 Ethernet+PPP+tunnel interfaces
  - 32 ipv6ip tunnel interfaces
  - 511 GRE tunnels (2,048 GRE tunnels per chassis)
• 256 loopback interfaces

• IP Addresses and IP Address Pools
  • Up to 2,000 IPv4 address pools can be configured within a single context.
  • Prior to Release 15.0: Up to 32 IPv6 pools can be configured within a single context.
  • For Release 15.0 and higher: Up to 256 IPv6 pools can be configured within a single context.
  • Up to a combined total of 5,000 IPv4 and IPv6 addresses can be configured per chassis.
  • Each context supports up to 32,000,000 static IP pool addresses. You can configure a maximum total of 96,000,000 static IP pool addresses per chassis. Each static IP pool can contain up to 500,000 addresses.
  • Each context supports up to 16,000,000 dynamic IP pool addresses. You can configure a maximum total of 32,000,000 dynamic IP pool addresses per chassis. Each dynamic IP pool can contain up to 500,000 addresses.

Important: The actual number of IP Pools supported per context and chassis depends on how many addresses are being used and how they are subnetted.

Important: Each address in the pool requires approximately 60 bytes of memory. The amount of memory required, however, depends on a number of factors such as the pool type, and hold-timer usage. Therefore, in order to conserve available memory, you may need to limit the number of pools depending on the number of addresses to be configured and the number of installed application cards.

• The maximum number of simultaneous subscriber sessions is controlled by the installed capacity license for the service(s) supported.

• The maximum number of static address resolution protocol (ARP) entries per context is 128.

• The maximum number of domains per context is 2,048.

• ASN-GW services configured within the same context cannot communicate with each other.

• Routes
  • Up to 1,200 static routes per context (48,000 per chassis).
  • 6,000 pool routes per context (6,000 per chassis)
  • Releases prior to 18.5: 5,000 pool explicit host routes per context (6,000 per chassis)
  • Release 18.5 and higher: 24,000 pool explicit host routes per context (24,000 per chassis)
  • 64 route maps per context

• BGP
  • Releases 12 and 14: 16,000 BGP prefixes can be learned/advertised per context (64,000 per chassis)
  • Releases 15 and 16: 32,000 BGP prefixes can be learned/advertised per context (64,000 per chassis)
• Releases 17, 18 and higher: 64,000 BGP prefixes can be learned/advertised per context (64,000 per chassis)
• 64 EBGP peers can be configured per context (512 per chassis)
• 16 IBGP peers per context
• 512 BGP/AAA monitors per context in support of Interchassis Session Recovery (ICSR)

• OSPF
  • 200 OSPF neighbors per chassis
  • 10,000 OSPF routes per context (64,000 per chassis)

• MPLS
  • 16 label distribution protocol (LDP) sessions per context
  • 8,000 forwarding equivalence class (FEC) entries per context (48,000 per chassis)
  • Up to 8,000 incoming label map (ILM) entries per context (48,000 per chassis)

• VRF
  • Prior to Release 15.0: 250 virtual routing and forwarding (VRF) tables per context (1,024 or 2,048 [release 14.0+] VRFs per chassis)
  • Release 15.0 and higher: 300 virtual routing and forwarding (VRF) tables per context (2,048 VRFs per chassis) [256 VRFs per context with demux functions enabled on the MIO card]
  • APN limit is 2,048 per chassis; VRF limits and APN limits should be identical.
  • 64,000 IP routes

• NEMO (Network Mobility)
  • Prior to Release 15.0: 256K prefixes/framed routes per chassis and up to 8 dynamically learned prefixes per MR (Mobile Router)
  • Release 15.0 and higher: 512K prefixes/framed routes per chassis and up to 16 dynamically learned prefixes per MR (Mobile Router)

• 128 AAA servers per context for a default AAA server group. The servers can be configured as accounting, authentication, charging servers, or any combination thereof.
• You can configure up to 800 AAA server groups per context with following limitations:
  • 128 servers per AAA server group (accounting, authentication, charging server, or any combination thereof)
  • 1,600 servers per context in AAA Server group mode (accounting, authentication, charging server, or any combination thereof)
  • 800 NAS-IP address/NAS identifier (one primary and one secondary per server group) per context

• Up to 12 charging gateway functions (CGFs) for GTPP accounting can be configured per context.
• Up to 16 bidirectional forwarding detection (BFD) sessions per context (64 per chassis)
Important

Refer to Engineering Rules in your product administration guide for additional information on product-specific operating limits.

Subscriber Rules

The following engineering rules apply to subscribers configured within the system:

• Configure a maximum of 2,048 local subscribers per context.

• You may configure attributes for each local subscriber.

• The system creates a default subscriber for each context when the context is made. Configure attributes for each default subscriber. If a AAA-based subscriber is missing attributes in the authentication reply message, the default subscriber attributes in the context where the subscriber was authenticated are used.

Important

Default is not used when local authentication (for local subscribers) is performed.

• Configure default subscriber templates on a per AAA realm (domain aliases configured within a context) basis.

• Configure default subscriber templates on a per PDSN, FA, ASN-GW, or HA service.

• For AAA authenticated subscribers, the selection of local subscriber template to use for setting attributes is in the following order:
  • If the username (NAI) matches any local domain name and the domain name has a local subscriber name configured, that local subscriber template is used.
  • If the first case fails, and if the serving service has a default username configured, that subscriber template is used.
  • If the first two cases fail, the default subscriber template in the AAA context is used.

Service Rules

The following engineering rules apply to services configured within the system:

• Configure a maximum of 256 services (regardless of type) per system.

Caution

Large numbers of services greatly increase the complexity of management and may affect overall system performance. Therefore, you should not configure a large number of services unless your application absolutely requires it. Please contact your Cisco service representative for more information.

• The total number of entries per table and per chassis is limited to 256.
• Although you can use service names that are identical to those configured in different contexts on the same system, this is not a good practice. Services with the same name can lead to confusion and difficulty in troubleshooting problems, and make it difficult to understand the output of `show` commands.

## Access Control List (ACL) Engineering Rules

The following rules apply to Access Control Lists:

• The maximum number of rules per ACL is 128.
• The maximum number of ACL rules applied per port is 128.
• The maximum number of ACL rules applied per context is 1,024.
• The maximum number of ACL rules per IPSec policy is 1.
• The maximum number of IPSec ACL rules per context is 1,024.
• The maximum number of IPSec ACL rules per crypto map is 8.
• The maximum number of ACLs you can configure per context is limited by the number of rules allowed within each ACL. If each ACL contained the maximum number of rules (128), the maximum number of ACLs per context is 8 (128 X 8 ACLs = 1,024 ACL rules per context).
• The maximum number of ACLs applied to an IP access group is 1, whether it is configured for a port or context. Since the maximum number of IP access groups you can apply to an interface or context is 16, the following calculations apply:
  • For each interface/port: 8 rules per ACL multiplied by 16 IP access groups = 128 (the ACL rules limit per port)
  • For each context: 64 rules per ACL multiplied by 16 IP access groups = 1,024 (the ACL rules limit per context)

## ECMP Groups

The maximum number of ECMP groups are as follows:

• For releases prior to 17.0, StarOS supports a maximum of 512 groups.
• For release 17.0 and higher, StarOS supports a maximum of 2048 groups.
This appendix describes system and subsystem tasks running under StarOS on an ASR 5x00 and virtualized platforms. It includes the following sections:

- Overview, page 331
- Primary Task Subsystems, page 332
- Controllers and Managers, page 333
- Subsystem Tasks, page 334

### Overview

For redundancy, scalability and robust call processing, StarOS supports a series of tasks that perform specific functions. These tasks communicate with each other as needed to share control and data signals. As a result, processes can be distributed across multiple tasks thus reducing the overall work-load on any given task and improving system performance. This distributed design provides fault containment that greatly minimizes the impact to processes or sessions due to a failure.

The Exec mode `show task` command displays snapshots of running processes within StarOS. For detailed information about this command, see the *Command Line Interface Reference* and *Statistics and Counters Reference*.

The following sections describe the primary tasks that are implemented by StarOS:

- Primary Task Subsystems, on page 332
- Controllers and Managers, on page 333
Primary Task Subsystems

The individual tasks that run on the CPUs are divided into subsystems. Following is a list of the primary subsystems responsible for call session processing:

- **System Initiation Task (SIT):** This subsystem starts tasks and initializes the system. This includes starting a set of initial tasks at system startup time (static tasks), and starting individual tasks on demand at arbitrary times (dynamic tasks).

- **High Availability Task (HAT):** With the Recovery Control Task (RCT) subsystem, the HAT subsystem maintains the operational state of the system. HAT monitors the various software and hardware components of the system. If there are unusual activities, such as the unexpected termination of another task, the HAT subsystem takes a suitable course of action, such as triggering an event to the RCT subsystem to take corrective action or to report the status. The primary function of the HAT task is to minimize service impacts.

- **Recovery Control Task (RCT):** This subsystem executes a recovery action for any failure that occurs in the system. The RCT subsystem receives signals from the HAT subsystem (and in some cases from the NPU subsystem) and determines what recovery actions are needed.

- **Shared Configuration Task (SCT):** This subsystem provides a facility to set, retrieve, and receive notification of system configuration parameters. The SCT is mainly responsible for storing configuration data for the applications that run on the system.

- **Resource Management (RM):** This subsystem assigns resources, such as CPU loading and memory, for every system task upon start-up. The RM subsystem monitors resource use to verify that allocations are as specified. RM also monitors all sessions and communicates with the Session Controller to enforce capacity licensing limits.

- **Virtual Private Network (VPN):** This subsystem manages the administrative and operational aspects of all VPN-related entities in the system. The functions performed by the VPN subsystem include:
  - Creating separate VPN contexts
  - Starting the IP services within a VPN context
  - Managing IP pools and subscriber IP addresses, and distributing the IP flow information within a VPN context.

All IP operations within StarOS are done within specific VPN contexts. In general, packets are not forwarded across different VPN contexts. The only exception currently is the Session subsystem.

- **Network Processing Unit (npuctrl/npumgr on ASR 5000; npusim on ASR 5500, and knpusim on VPC-DI and VPC-SI):** This subsystem is responsible for the following:
  - Using the database to match address and port numbers to destination tasks for fast-path forwarding of dataframes
  - Receiving and transmitting user data frames to/from various physical interfaces
  - IP forwarding decisions (both unicast and multicast)
• Per-interface packet filtering
• Traffic management and traffic engineering
• Passing user data frames to/from packet processing CPUs
• Modifying, adding, or stripping datalink/network layer headers
• Recalculating checksums
• Maintaining statistics
• Managing external Ethernet interfaces

• **Card/Slot/Port (CSP):** Coordinates the events that occur when any card is inserted, locked, unlocked, removed, shutdown, or migrated. CSP also performs auto-discovery and configures ports on a newly-inserted interface card. It determines how interface cards map to packet processing cards.

The CSP subsystem runs only on the active management card and synchronizes the information it contains with the SCT subsystem on the standby management card. It is started by the SIT subsystem and monitored by the HAT subsystem.

• **Session Manager (SM):** Performs high-touch processing of mobile subscribers' packet-oriented data session flows. High-touch user data processing consists of the following:
  • Payload transformation
  • Filtering and scheduling
  • Statistics collection
  • Policing

---

**Controllers and Managers**

Many of the primary subsystems are composed of controller tasks called Controllers, and subordinated tasks called Managers.

Controllers serve several purposes:

• Monitor the state of their Managers and allow communication between Managers within the same subsystem.

• Enable inter-subsystem communication since they can communicate with the controllers of other subsystems.

• Mask the distributed nature of the software from the user allowing for ease of management.

Managers manage resources and mappings between resources. In addition, some managers are directly responsible for call processing.

For information about the primary subsystems that are composed of critical, controller, and /or manager tasks, see Subsystem Tasks, on page 334.
Subsystem Tasks

The following subsections list and briefly describe StarOS tasks for various subsystems:

- System Initiation Subsystem, on page 334
- High Availability Subsystem, on page 335
- Resource Manager Subsystem, on page 336
- Virtual Private Networking Subsystem, on page 336
- Network Processing Unit Subsystem, on page 338
- Session Subsystem, on page 340
- Platform Processes, on page 350
- Management Processes, on page 354

System Initiation Subsystem

Table 43: System Initiation Subsystem Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>SITMAIN</td>
<td>System Initiation Task – Main</td>
<td>Initiated at system start-up. Reads and provides startup configuration to other SIT components.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Starts SITREAP sub-function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains CPU state information.</td>
</tr>
<tr>
<td>SITPARENT</td>
<td>SIT Parent Sub-function</td>
<td>Starts management cards in either active or standby mode.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Registers tasks with HAT task.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Notifies CSP task of CPU startup completion.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Brings up packet processing cards in standby mode.</td>
</tr>
<tr>
<td>SITREAP</td>
<td>SIT Reap Sub-function</td>
<td>Shuts down tasks as required.</td>
</tr>
</tbody>
</table>
### High Availability Subsystem

#### Table 44: High Availability Subsystem Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>hatcpu</td>
<td>High Availability Task CPU</td>
<td>Performs device initialization and control functions based on the CPUs hardware capabilities.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reports the loss of any task on its CPU to hatsystem sub-function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Controls the LEDs on the packet processing cards.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initializes and monitors the dedicated hardware on packet processing cards. (ASR 5x00 only)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Collects CPU monitoring information periodically and reports to the master hatcpu sub-function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reports the loss of any task on its CPU to the master hatcpu sub-function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs device initialization and control functions because of the CPU's hardware capabilities.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Reports the loss of any task on its CPU to hatsystem sub-function.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Controls the LEDs on the management card. (ASR 5x00 only)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initializes and monitors the dedicated hardware on the management card. (ASR 5x00 only)</td>
</tr>
<tr>
<td>hatsystem</td>
<td>High Availability Task System Controller</td>
<td>Controls all the HAT sub-function tasks in the system. It is initiated on system start-up.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initializes system components (such as the Gigabit Ethernet switches and switch fabric).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Monitors system components such as fans for state changes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Triggers actions for redundancy in the event of fault detection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The HAT subsystem on the redundant management card mirrors the HAT subsystem on the active management card.</td>
</tr>
</tbody>
</table>
Resource Manager Subsystem

Table 45: Resource Manager (RM) Subsystem Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>rmctrl</td>
<td>Resource Manager Controller</td>
<td>Started by the sitparent task on StarOS startup, and monitored by HAT for a failure.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initializes resources such as CPUs and memory.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Requests updated card status from the CSP subsystem and updates the system card table.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Communicates with all rmctrls to request their most recent set of resource data.</td>
</tr>
<tr>
<td>rmctrl</td>
<td>Resource Manager Managers</td>
<td>Started by the sitparent task, and monitored by HATs for failures.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initializes the local resource data and local resource scratch space.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Communicates with the SIT task on the local CPU to get its entire task table and the resources associated with each task.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Gathers current resource utilization for each task.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sends the resource data to the rmctrl task.</td>
</tr>
</tbody>
</table>

Virtual Private Networking Subsystem

Table 46: Virtual Private Networking (VPN) Subsystem Tasks

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>vpnctrl</td>
<td>VPN Controller</td>
<td>Created at system start-up.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Initiates the VPN Manager for each context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Informs the Session Controller task when there are additions or changes to contexts. Only one Session Controller operates at any time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Routes context specific operation information to the appropriate VPN Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs VPN Manager recovery and saves all VPN-related configuration information in SCT.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>vpnmgr</td>
<td>VPN Manager</td>
<td>Started by the VPN Controller for each configured context (one is always present for the local context).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs IP address pool and subscriber IP address management.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs all context specific operations including but not limited to: UCM services, IP interfaces, the Address Resolution Protocol (ARP), IP address pool management, slow path forwarding, NPU flows, port Access Control Lists (ACLs), and logging.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides IP interface address information for each context to the Session Controller.</td>
</tr>
<tr>
<td>bgp</td>
<td>Border Gateway Protocol</td>
<td>Created by the VPN Manager for each context that has enabled the BGP routing protocol (<em>router bgp</em> Context Configuration mode CLI command).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Responsible for learning and redistributing routing information via the BGP protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the BGP peering connections.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applies any defined BGP routing policy.</td>
</tr>
<tr>
<td>ospf</td>
<td>Open Shortest Path First</td>
<td>Created by VPN Manager for each context that has enabled the OSPF routing protocol (<em>router ospf</em> Context Configuration mode CLI command).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Responsible for learning and redistributing routing information via the OSPF protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the OSPF neighboring relationship.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the LSA database.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs SPF calculations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applies any defined OSPF routing policy.</td>
</tr>
<tr>
<td>ospfv3</td>
<td>Open Shortest Path First</td>
<td>Created by VPN Manager for each context that has enabled the OSPFv3 routing protocol (<em>router ospfv3</em> Context Configuration mode CLI command)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Responsible for learning and redistributing routing information via the OSPFv3 protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the OSPFv3 neighboring relationship.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the LSA database.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs OSPFv3 SPF calculations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Applies any defined OSPFv3 routing policy.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>----------</td>
</tr>
</tbody>
</table>
| rip  | Routing Information Protocol | Created by VPN Manager for each context that has enabled the RIP routing protocol (router rip Context Configuration mode CLI command)  
- Responsible for learning and redistributing routing information via the RIP protocol.  
- Maintains the RIP database.  
- Sends periodic RIP update messages.  
- Applies any defined RIP routing policy. |
| zebos | L2 and L3 Switching | Created by VPN Manager for each context.  
- Maintains the routing table (RIB and FIB) for the context.  
- Performs static routing.  
- Interfaces to the kernel for routing & interface updates.  
- Redistributes routing information to dynamic routing protocols.  
- Calculates next hop reachability. |

**Network Processing Unit Subsystem**

*Table 47: Network Processing Unit (NPU) Subsystem Tasks*

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
</table>
| knpusim | Kernel-based NPU Simulator [VPC-DI, VPC-SI] | Created at StarOS start up.  
- Provides port configuration services to the CSP task.  
- Provides interface binding and forwarding services to the VPN Manager.  
- Provides flow insertion and removal services to Session Manager and AAA Manager tasks.  
- Provides recovery services to the NPU Controller. |
<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>npuctrl</td>
<td>NPU Controller</td>
<td>Created at StarOS start-up. Only one NPU Controller operates in the system at any time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Monitors the state of NPU Managers in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Registers to receive notifications when NPU Manager crashes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Controls recovery operation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides a centralized location for CLI commands related to NPU Manager state.</td>
</tr>
<tr>
<td>npumgr</td>
<td>NPU Manager</td>
<td>Created for every packet processing card that is installed and started.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides port configuration services to the CSP task.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides interface binding and forwarding services to the VPN Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides flow insertion and removal services to Session Manager and AAA Manager tasks.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides recovery services to the NPU Controller.</td>
</tr>
<tr>
<td>npusim</td>
<td>NPU Simulator [ASR 5500]</td>
<td>Created for every DPC installed and started.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides port configuration services to the CSP task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides interface binding and forwarding services to the VPN Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides flow insertion and removal services to Session Manager and AAA Manager tasks.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides recovery services to the NPU Controller.</td>
</tr>
</tbody>
</table>
# Session Subsystem

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>sessctrl</td>
<td>Session Controller</td>
<td>Created at StarOS start-up. Only one Session Controller instantiated in the system at any time. Acts as the primary point of contact for the Session Subsystem. Since it is aware of the other subsystems running within the system, the Session Controller acts as a proxy for the other components, or tasks, that make up the subsystem. Starts, configures, and coordinates the efforts of the Session Processing Subsystem sub-managers. Works with Resource Manager to start new Session Managers when all existing Session Managers exceed their capacity. Receives context information from VPN Managers. Distributes IP interface address information to other Session Processing Subsystem sub-managers. Manages Enhanced Charging Service (ECS), Content Filtering and URL Blacklisting services.</td>
</tr>
<tr>
<td>sessmgr</td>
<td>Session Manager</td>
<td>Created by the Session Controller. Provides a subscriber processing system that supports multiple session types. Multiple Session Managers can run on a single CPU and/or can be distributed throughout any CPU present in the system. A single Session Manager can service sessions from multiple A11 Managers, and from multiple contexts. Processes protocols for A10/A11, GRE, R3, R4, R6, GTPU/GTPC, PPP, and Mobile IP. Manages Enhanced Charging Service, Content Filtering and URL Blacklisting services. Session Managers are paired with AAA Managers.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>a11mgr</td>
<td>A11 Manager</td>
<td>Created by the Session Controller for each context in which a PDSN service is configured.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receives the R-P sessions from the PCF and distributes them to different Session Manager tasks for load balancing.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains a list of current Session Manager tasks to aid in system recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The A11 Manager task is also known as the Signaling De-multiplexing task (SDT).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>aaamgr</td>
<td>Authorization, Authentication, and Accounting (AAA) Manager</td>
<td>Paired with Session Managers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs all AAA protocol operations and functions for subscribers and administrative users within the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Acts as a AAA client to AAA servers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Manages GTP Prime (GTP’) messaging with charging gateway functions (CGFs).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Multiple AAA Managers can run on a single CPU and/or can be distributed throughout any CPU present in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AAA operations for the CLI are done through a AAA Manager running on the active management card.</td>
</tr>
<tr>
<td>aaaproxy</td>
<td>Authorization, Authentication, and Accounting (AAA) Proxy Manager</td>
<td>Starts whenever the Global Configuration mode <code>gtp single-source</code> command is configured.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>When GTPP single-sourcing is enabled, aaaproxy generates requests to the accounting server using a single UDP source port number, instead of having each AAA Manager generate independent requests with unique UDP source port numbers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runs on a demux card when session recovery is enabled. If session recovery is not enabled, the Global Configuration mode <code>require demux card</code> command starts aaaproxy on the designated demux card.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Writes CDRs to a file in its VRAM-disk. The enqueued CDRs are then periodically synchronized with a HDD for transfer.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>acsctrl</td>
<td>Active Charging System (ACS) Controller</td>
<td>Active Charging service is defined at the global level and can be utilized through CSS commands from any VPN context. Enable via the Global Configuration mode <strong>active-charging service</strong> CLI command. The ACS controller runs on the primary packet processing card and is responsible for managing the ACS service. Reads and writes ACS configuration information into SCT. The ACS Controller monitors the ACS Manager's recovery process and performs cleanup when redundancy is enabled.</td>
</tr>
<tr>
<td>acsmgr</td>
<td>Active Charging System (ACS) Controller</td>
<td>Created by ACS Controller to perform IP session processing for a specific number of flows. Sends and receives data through Session Managers. Active/Standby acsmgr tasks are created when session recovery (SR) is enabled.</td>
</tr>
<tr>
<td>alcapmgr</td>
<td>Access Link Control Application Part Manager [ASR 5000 only]</td>
<td>Starts when an ALCAP service configuration is detected. There can be multiple instances of this task for load sharing. Runs the ALCAP protocol stack and handles the IuCS-over-ATM associations. Maintains AAL2 node entity databases. Provides nodal functions for IuCS-over-ATM interface on ALCAP protocol.</td>
</tr>
<tr>
<td>cdrmod</td>
<td>Charging Detail Record Module</td>
<td>Responsible for receiving EDR/UDR records from different ACSMGR instances in the system. Responsible for writing the received EDR/UDR records in files using the configured file naming conventions.</td>
</tr>
<tr>
<td>dgmbmgr</td>
<td>Diameter Gmb interface Application Manager</td>
<td>Provides Multimedia Broadcast/Multicast Service (MBMS) feature support for GGSN. It is instantiated when an MBMS policy CLI is configured in the GGSN Service configuration mode. dgmbmgr Maintains the MBMS UE and bearer contexts. Handles the Gmb interface over a Diameter connection to a BMSC Server for MBMS bearer sessions. dgmbmgr recovers by polling all sessmgrs for MBMS session states and recreating the MBMS UE and MBMS bearer context information.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>diamproxy</td>
<td>Diameter Proxy</td>
<td>Created by diactrl (which runs as part of vpnctrl) and the number of diamproxy tasks spawned is based on the configuration to use &quot;multiple&quot; or &quot;single&quot; proxies. In instances that a single proxy is configured, only one diamproxy task is spawned for the entire chassis and runs on demux packet processing cards. When multiple proxies are configured, one diamproxy task is run per packet processing card. Maintains Diameter base connections to all peers configured in the system. Informs applications about any change in the connection status. Acts as a pass-through to the messages from application to the Diameter server. Just acts as a forwarding agent (does not maintain any queues). A single Diameter proxy is used to service multiple Diameter applications.</td>
</tr>
<tr>
<td>egt pemgrr</td>
<td>Enhanced GPRS Tunneling Protocol Egress Manager</td>
<td>Created by the Session Controller for each context in which an egtp-service of interface type sgw-egress or MME is configured. Handles certain EGTP messages from SGW, PGW. Maintains list of current EGTP sessions. Maintains list of current Session Manager tasks which aids in session recovery. Handles GTP Echo messaging. With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>egtpimgr</td>
<td>Enhanced GPRS Tunneling Protocol Ingress Manager</td>
<td>Created by Session Controller for each context in which an egtp-service of interface type sgw-ingress or pgw-ingress is configured. Receives EGTP sessions from MME/S4 SGSN/SGW and distributes them to different Session Manager tasks for load balancing. Maintains list of current EGTP sessions. Maintains list of current Session Manager tasks which aids in session recovery. Handles GTP Echo messaging. With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>gtpcmgr</td>
<td>GPRS Tunneling Protocol Control (GTP-C) Message Manager</td>
<td>Created by the Session Controller for each context in which a GGSN service is configured. Receives the GTP sessions from the SGSN and distributes them to different Session Manager tasks for load balancing. Maintains a list of current Session Manager tasks to aid in system recovery. Verifies validity of GTP-C messages. Maintains a list of current GTP-C sessions. Handles GTP-C Echo messaging to/from SGSN.</td>
</tr>
<tr>
<td>gtpumgr</td>
<td>GPRS Tunneling Protocol User (GTP-U Manager)</td>
<td>Created by the Session Controller for each context in which a GTPU service is configured. Supported for both GTPUv0 and GTPUv1. Maintains a list of the GTPU-services available within the context and performs load-balancing (of only Error-Ind) for them. Supports GTPU Echo handling. Provides Path Failure detection on no response for GTPU echo. Receives Error-Ind and demuxes it to a particular Session Manager. Serves as the Default GTPU listener. GTPUMGR will process GTPU packets with invalid TEID. With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>hamgr</td>
<td>Home Agent (HA) Manager</td>
<td>Created by the Session Controller for each context in which an HA service is configured.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receives Mobile IP sessions from the Foreign Agents (FAs) and distributes them to different Session Manager tasks.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains a list of current Session Manager tasks that aids in system recovery.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Functions as the DemuxMgr – handles all the PMIP signaling packets.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Functions as the Demuxmgr for MIPv6/MIPv4 HA.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>hnbdemux</td>
<td>Home NodeB (HNB) Demux Manager</td>
<td>Started as part of HNB-GW service creation procedure. There is only one hnbdemux in the chassis.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Distributes incoming Iuh connections to HNB Managers in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remains aware of all the active HNB-GW services in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>hnbmgr</td>
<td>Home NodeB (HNB) Manager</td>
<td>Starts when an HNB-GW service configuration is detected. There can be multiple instances of this task for load sharing.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>All HNB Managers have all the Active HNB-GW Services configured and be identical in configuration and capabilities.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runs the SCTP protocol stack.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Handles the SCTP associations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains Home-NodeB databases.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides nodal functions for Iuh interface on SCTP protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>imsimgr</td>
<td>International Mobile Subscriber Identity Manager for MME</td>
<td>Starts when an MME service configuration is detected. There is only one instance of this task:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Selects which SessMgr to use for new subscriber sessions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains and reports MME-related demux statistics on events like Attach by IMSI, Attach by GUTI, etc.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Can interact with the following tasks in the system:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Session Controller</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- MME Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Session Manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>imsimgr</td>
<td>International Mobile Subscriber Identity Manager for SGSN</td>
<td>Started by the Session Controller.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Selects SessMgr, when not done by linkmgr or sgtpcmgr tasks, for calls sessions based on IMSI/P-TMSI.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Load-balances across SessMgrs to select one to which a subscriber will be assigned.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains records for all subscribers on the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains mapping between the IMSI/P-TMSI and SessMgrs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.</td>
</tr>
<tr>
<td>ipsgmgr</td>
<td>IP Services Gateway Manager</td>
<td>Created by the Session Controller.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Server mode, acts as a RADIUS server, and supports Proxy functionality.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>In Snoop mode supports snooping RADIUS Accounting messages.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Load balances requests among different SessMgrs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Activates and deactivates sessions.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>-------</td>
<td>--------------------------------------</td>
<td>------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>l2tpdemux</td>
<td>L2TP Demultiplexor Task</td>
<td>Created by the Session Controller when an LNS service is created. Only one L2TPDemux task is invoked for the entire system. De-multiplexes and forwards new incoming tunnel create requests to L2TPMgs. Maintains information about current active tunnels in all L2TPMgs. Load balances requests among L2TPMgs. With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>l2tpmgr</td>
<td>Layer 2 Tunneling Protocol Manager</td>
<td>Created by the Session Controller for each context in which a LAC or LNS service is configured. Additional managers are created as needed depending on loading. Responsible for all aspects of L2TP processing. Maintains protocol state machines for all L2TP sessions and tunnels. Triggers IPSec encryption for new L2TP tunnels as needed. Works with Session Managers to gracefully bring down tunnels. With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>linkmgr</td>
<td>SS7 Link Manager</td>
<td>Created by the Session Controller when the first SS7RD (routing domain) is activated. Multi-instanced for redundancy and scaling purposes. Provides SS7 and Gb connectivity to the platform. Routes per subscriber signalling across the SS7 (including Iu) and Gb interfaces to the SessMgr.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>magmgr</td>
<td>Mobile Access Gateway (MAG) Manager</td>
<td>Created by the Session Controller when the first MAG service is created in a context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sends and receives PMIP control messages (PBU/PBA).</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Adds an NPU flow to receive MIPv6 PBA packets. This flow is identical to the flow used in the HAMgr.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains the Binding Update List used to keep track of the mobile node's bindings.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Originates PBU-based on trigger received from the Session Manager during error conditions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receives PBA and forwards it to Session Manager.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Supports debugging facility – &quot;magmgr&quot; and &quot;mobile-ipv6&quot;.</td>
</tr>
<tr>
<td>mmgr</td>
<td>SGSN Master Manager</td>
<td>Created upon provisioning of SS7RDs/SCCP-NWs/etc. The Session Controller provides the initial system configuration which includes a detailed description of each distributed protocol layer, its resources sets, and a list of its service user protocol layers and service provider protocol layers.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runs as a single instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Handles nodal SS7, Iu, and Gb functionality.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Implements master linkmgr functionality for SS7 route status aggregation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Implements master linkmgr functionality for RNC and BSC status aggregation.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.</td>
</tr>
<tr>
<td>mmedemux</td>
<td>Mobility Management Entity Demux Manager</td>
<td>Started as part of MME service creation procedure. There is only one mmedemux in the chassis.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Distributes incoming S1-MME SCTP connections to mmemgr tasks in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remains aware of all the active MME services in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>mmemgr</td>
<td>Mobility Management Entity Manager</td>
<td>Starts when an MME service configuration is detected. There can be multiple instances of this task for load sharing. All mmemgrs will have all the Active MME Services configured and will be identical in configuration and capabilities.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Runs the SCTP protocol stack.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Handles the SCTP associations.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains TA List.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Manage eNodeB databases.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Provides nodal functions for S1-MME protocol.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>pccdemux</td>
<td>Policy and Charging Control Bindmux Manager</td>
<td>Started as part of PCC service creation procedure. There is only one instance of BindMux MGR in the chassis.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Handles multiplexing of the sessions across the available pccmgrs along with the session binding functions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Monitors load on pccmgrs.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Distributes incoming IP-CAN connections across pccmgrs in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs session binding; binds IP-CAN/Gateway session with the AF-Session.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ensures all messaging for an IMSI across various interfaces is directed towards the selected pccmgr.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Remains aware of all the active PCC services in the system.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active packet processing card.</td>
</tr>
<tr>
<td>pccmgr</td>
<td>Policy and Charging Control Bindmux Manager</td>
<td>pccmgr is part of a Session Manager instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Handles all PCRF service sessions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Interfaces with PCC-Core while processing different events associated with individual subscriber sessions.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Maintains subscriber information while applying business logic.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Creates calline and corresponding APN session for each subscriber.</td>
</tr>
</tbody>
</table>
**Platform Processes**

**Table 49: Platform Process Tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>sgtpcmgr</td>
<td>SGSN GPRS Tunneling Protocol Control message Manager</td>
<td>Created by the Session Controller for each VPN context in which an SGSN service is configured.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Terminates Gn/Gp and GTP-U interfaces from peer GGSNs and SGSNs for SGSN Services.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Terminates GTP-U interfaces from RNCs for IuPS Services.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Controls standard ports for GTP-C and GTP-U.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Processes and distributes GTP-traffic received from peers on these ports.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs all node level procedures associated with Gn/Gp interface.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>With session recovery (SR) enabled, this demux manager is usually established on one of the CPUs on the first active demux packet processing card.</td>
</tr>
<tr>
<td>srb</td>
<td>Standard Routing Database</td>
<td>Eight srbs are created by the Session Controller when Content Filtering in the Enhanced Charging Service is enabled. A minimum of two packet processing cards are required to initiate these eight tasks.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Receives the static database from the session controller. Each srb task loads two database volumes (one primary and one secondary). The srb task also stores the static DB.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rates and categorizes the URL based on the DB volumes and CSI (Category Set Index) stored on it.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Performs peer loading in case its peer fails. If both the srb task and its peer fail, the session controller performs the loading.</td>
</tr>
</tbody>
</table>

**Platform Processes**

**Table 49: Platform Process Tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>afctrl</td>
<td>ASR 5500 Fabric Controller</td>
<td>Responsible for the overall management of the system fabric. Manages the pool of Rendezvous Destinations and coordinates fabric recovery by the afmgr procslets after a fault. A single afctrl instance runs on the active MIO/UMIO only.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>afmgr</td>
<td>ASR 5500 Fabric Manager</td>
<td>Responsible for the management of fabric resources on a particular card.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>There is one afmgr on every CPU that is responsible for one or more</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fabric access processors (FAPs) or fabric elements (FEs). afmgr supports</td>
</tr>
<tr>
<td></td>
<td></td>
<td>recovery but not migration.</td>
</tr>
<tr>
<td>afio</td>
<td>ASR 5500 Fabric I/O Driver</td>
<td>Responsible for the direct configuration of the fabric chipset. afio</td>
</tr>
<tr>
<td></td>
<td></td>
<td>supports non-messenger interprocess communication (IPC) with the local</td>
</tr>
<tr>
<td></td>
<td></td>
<td>afmgr and with other local afio instances.</td>
</tr>
<tr>
<td>connproxy</td>
<td>TCP/SCTP Connection proxy</td>
<td>Allows applications on any card to share the same TCP/SCTP connection</td>
</tr>
<tr>
<td></td>
<td></td>
<td>to the same remote endpoint instead of opening a new connection for each</td>
</tr>
<tr>
<td></td>
<td></td>
<td>application on the card.</td>
</tr>
<tr>
<td>cspctrl</td>
<td>Card-Slot-Port Controller</td>
<td>Manages physical chassis components.</td>
</tr>
<tr>
<td>cssctrl</td>
<td>Content Server Selection (CSS) Controller</td>
<td>Maintains all global CSS properties which include a list of CSS servers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>that can be bound to a service in a context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CSS defines how traffic will be handled based on the &quot;content&quot; of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>data presented by or sent to a mobile subscriber. CSS encompasses features</td>
</tr>
<tr>
<td></td>
<td></td>
<td>such as load balancing, NAT, HTTP redirection, DNS redirection.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The content server (services) can be either external to the platform or</td>
</tr>
<tr>
<td></td>
<td></td>
<td>integrated within the platform. External CSS servers are configured via</td>
</tr>
<tr>
<td></td>
<td></td>
<td>the Context Configuration mode css server command.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The CSS Controller does not create CSS Managers. CSS Managers are</td>
</tr>
<tr>
<td></td>
<td></td>
<td>stopped and started by VPN Managers. A CSS Manager is automatically</td>
</tr>
<tr>
<td></td>
<td></td>
<td>created for each context.</td>
</tr>
<tr>
<td>cssmgr</td>
<td>Content Server Selection (CSS) Manager</td>
<td>Spawned by the VPN Manager within a StarOS context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Manages the keepalives to a CSS server within the specific VPN context.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fetches the CSS related information for a subscriber.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>If a CSS server goes down, the cssmgr task reprograms the NPUs to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>by-pass the service or redistribute the data among the rest of the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>servers in the service.</td>
</tr>
<tr>
<td>dcardctrl</td>
<td>Daughter Card Controller</td>
<td>Spawns daughter card managers during system initialization and monitors</td>
</tr>
<tr>
<td></td>
<td>[ASR 5x00 only]</td>
<td>daughter card managers during system steady state execution. It also</td>
</tr>
<tr>
<td></td>
<td></td>
<td>spawns daughter card managers whenever a daughter card manager task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>fails.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>dcardmgr</td>
<td>Daughter Card Manager [ASR 5x00 only]</td>
<td>Responsible for managing IPSec Security Associations for AH- and ESP-based sessions. Interfaces with the on-board hardware accelerated cryptographic chip which executes cryptographic algorithms associated with the given IPSec Security Associations.</td>
</tr>
<tr>
<td>dhmgr</td>
<td>Distributed Host Manager</td>
<td>Started automatically on each CPU by SITPARENT. Coordinates establishment of locally terminated TCP, SCTP, and UDP connections on behalf of multi-instanced tasks such as Diameter endpoints among sessmgr tasks.</td>
</tr>
<tr>
<td>drvctrl</td>
<td>Driver Controller</td>
<td>Centralizes access to many of the system device drivers. It also performs temperature and voltage monitoring.</td>
</tr>
<tr>
<td>hdctrl</td>
<td>Hard Drive Controller</td>
<td>Controls and manages the drive array spanning the management cards.</td>
</tr>
<tr>
<td>hwctrl</td>
<td>Hardware Controller</td>
<td>The hwctrl task has several timers that manage polling loops for hardware sensor readings, sensor threshold monitoring, and fan tray monitoring.</td>
</tr>
<tr>
<td>hwmgr</td>
<td>Hardware Manager</td>
<td>The hwmgr task runs on all cards in the chassis to read local accessible hardware sensors and report them back to the hwctrl.</td>
</tr>
<tr>
<td>inetd</td>
<td>InterNET Service Daemon</td>
<td>The subsystem responsible for starting most of the network services. Listens for requests from connecting clients, such as FTP, SFTP, and telnet. When a TCP packet or UDP packet arrives with a particular destination port number, inetd launches the appropriate server program to handle the connection. <strong>Note:</strong> In release 20.0 and higher Trusted StarOS builds, FTP and Telnet are not supported.</td>
</tr>
<tr>
<td>ipsecctrl</td>
<td>IPSec Controller</td>
<td>Started by SIT on system startup regardless of configuration. Starts ipsecmgr tasks based on configuration and maintains its list for task recovery. Receives and maintains user configuration for IPSec. Manages the configured IPSec crypto maps and its assignment to ipsecmgrs. Interfaces with the vpmgr task for required IPSec configuration parameters such as IP Access Lists, IP pools, interface addresses, and interface state notifications.</td>
</tr>
<tr>
<td>ipsecmgr</td>
<td>IPSec Manager</td>
<td>Created by the Session Controller, establishes and manages secure IKEv1, IKEv2 and IPSec data tunnels.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>kvctrl</td>
<td>Key Value Controller</td>
<td>Central key value store (kvstore) function that runs on the management card. Its primary function is to support recovery and distribution functions.</td>
</tr>
<tr>
<td>lagmgr</td>
<td>Link Aggregation Group Manager</td>
<td>Started by npucntl on the demux card's primary CPU (ASR 5000) or MIO (ASR 5500) with a facility level between CSP and npumgr to receive configuration/status notification from npumgr and build global LAG database. Exchanges control packets (LACP and Marker) over configured physical ports with peers to reach agreement on an aggregation of links.</td>
</tr>
<tr>
<td>msgd</td>
<td>Messenger Daemon</td>
<td>Implements the Name Service and related functions for the internal message passing system.</td>
</tr>
<tr>
<td>msgproxy</td>
<td>Message Proxy</td>
<td>The Messenger Proxy process handles broadcast messages send from any single application (referred to as a client) to any facility which has one instance per thread (referred to as the Target Facility). One msgproxy task runs on each CPU complex on the SMC (ASR 5000), DPCs (ASR 5500) and SF Virtual Machine (VPC-DI). Processes incoming broadcast messages from the Client processes, such as sessctrl, distributes them to the correct Target Facility, such as sessmgr, creates the correct responses and sends them back to the correct Client.</td>
</tr>
<tr>
<td>nscontrol</td>
<td>Name Service Controller</td>
<td>As part of the Messenger process, provides a reliable channel for tasks to send control messages to the Messenger Daemon.</td>
</tr>
<tr>
<td>ntpd</td>
<td>Network Time Protocol (NTP) Daemon</td>
<td>Maintains the system time in synchronization with time servers using NTP. Enabled when one or more NTP servers have been configured via the NTP Configuration mode ntp server CLI command.</td>
</tr>
<tr>
<td>rct</td>
<td>Recovery Control Task</td>
<td>Monitors tasks/managers/facilities across the system and performs recovery in the event of a failure.</td>
</tr>
<tr>
<td>sct</td>
<td>Shared Configuration Task</td>
<td>Performs the redundant storage of configuration information and other state information in an in-memory database.</td>
</tr>
<tr>
<td>sft</td>
<td>Switch Fabric Task</td>
<td>Monitors the switch fabric and the gigabit Ethernet control plane.</td>
</tr>
<tr>
<td>sshd</td>
<td>Secure SHEll Daemon</td>
<td>Supports secure login to the StarOS CLI. Enabled via the Context Configuration mode server sshd CLI command.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>----------</td>
</tr>
<tr>
<td>ucm</td>
<td>Utilities Configuration Manager</td>
<td><strong>DHCPD, DNS, FTPD, INETD, NTPD, PING, RLOGIN, SFTP, SFTP-SERVER, SNMPD, SSH, SSHD, TELNET, TELNETD, TFTP, TRACEROUTE</strong>  &lt;br&gt;<strong>Note:</strong> In release 20.0 and higher <strong>Trusted</strong> StarOS builds, FTP and Telnet are not supported.</td>
</tr>
</tbody>
</table>

---

**Management Processes**

**Table 50: Management Process Tasks**

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>bulkstat</td>
<td>Bulk Statistics Manager</td>
<td>Periodically polls and gathers bulk statistics and transfers this data to external management systems.</td>
</tr>
<tr>
<td>evlogd</td>
<td>Event Log Daemon</td>
<td>Handles event logging functions including the interface to external syslogd servers and the internal event logs.</td>
</tr>
<tr>
<td>orbs</td>
<td>ORBEM Service  &lt;br&gt;[ASR 5x00 only]</td>
<td>The orbs task is also known as the ORB Element Manager (ORBEM).  &lt;br&gt;An Element Management System (EMS) requests orbs to perform Element Management Functions on the system using secure IIOP. ORBS then interacts with concerned Controller Tasks to execute the function.  &lt;br&gt;The response/errors from the execution are interpreted, formulated into an EMF response, and handed off to EMS servers.</td>
</tr>
<tr>
<td>orbns</td>
<td>ORBEM Notification Service  &lt;br&gt;[ASR 5x00 only]</td>
<td>Notifies the EMS servers of event occurrences.  &lt;br&gt;Registers such EMS servers and subscribes them to associated event types.  &lt;br&gt;As the events occur, the concerned Controller Task notifies orbs (ORBEM), which then notifies the subscribing EMS servers.</td>
</tr>
<tr>
<td>sesstrc</td>
<td>Session Trace Collection Task</td>
<td>Implements the standards-based session trace functionality.  &lt;br&gt;Manages both CLI and signaling-based subscriber traces. It collects messages to be traced and generates trace files as needed. It uploads trace files to the Trace Collection Entity as needed.</td>
</tr>
<tr>
<td>snmp</td>
<td>Simple Network Management Protocol</td>
<td>Handles inboard SNMP operations if configured, and sends SNMP notifications (traps) if enabled.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>Function</td>
</tr>
<tr>
<td>--------</td>
<td>--------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>threshold</td>
<td>Threshold Server</td>
<td>Handles monitoring of threshold crossing alerts, if configured. Polls the needed statistics/variables, maintains state, and generates log messages/SNMP notification of threshold crossings.</td>
</tr>
</tbody>
</table>
NETCONF and ConfD

This chapter describes the StarOS NETCONF interface and the ConfD engine. It contains the following sections:

- Overview, page 357
- Configuring ConfD, page 358
- Verifying the Configuration, page 360
- Show Support Details (SSD), page 364
- CDB Maintenance, page 364
- Supported StarOS ECS Configuration Commands, page 365

Overview

StarOS provides a northbound NETCONF interface that supports a YANG data model for transferring configuration and operational data with the Cisco Network Service Orchestrator (NSO). It also incorporates a ConfD engine to communicate with the NSO management console.

NETCONF (Network Configuration Protocol) is a network management protocol developed and standardized by the IETF (RFC 6241). It provides mechanisms to install, manipulate, and delete the configuration of network devices. Its operations are realized on top of a simple remote procedure call (RPC) layer. The NETCONF protocol uses XML-based data encoding for the configuration data as well as the protocol messages. The protocol messages are exchanged on top of a secure transport protocol.

ConfD is an on-device management framework that provides a set of interfaces to manage a device. The ConfD framework automatically renders all the management interfaces from a data-model. ConfD implements the full NETCONF specification and runs over SSH with content encoded in XML.

ConfD is configured to allow only authenticated/authorized access through external authentication. ConfD Manager provides a standalone CLI module for ConfD to invoke when authenticating/authorizing any new users. ConfD is configured to allow only authorized access through StarOS authentication. Upon authentication the user is given a privilege level (0-15) which is mapped to StarOS "secure admin", "admin", "operator" and "inspector" as defined in the YANG model. StarOS logs CLI authentication event/status messages for each ConfD authentication request.
On the southbound side, ConfD communicates with a StarOS process called via a set of APIs provided by the ConfD management agent. The ConfD Configuration Database (CDB) is used by ConfD to store objects. StarOS accesses the database through the ConfD supplied APIs. Once the ConfD config database is populated, StarOS continues to allow CLI access to modify the overall configuration. There are no automatic updates to the CDB as a result. The CDB only receives updates via the NETCONF interface. In order to keep the CDB and the StarOS configuration databases in sync, all changes made via CLI access (external to NETCONF) to YANG model supported config objects must be applied to the CDB manually.

YANG is a data modeling language for the NETCONF network configuration protocol. It can be used to model both configuration data as well as state data of network elements. YANG can also be used to define the format of event notifications emitted by network elements and it allows data modelers to define the signature of remote procedure calls that can be invoked on network elements via the NETCONF protocol (RFC 6020). The YANG file is compiled as part of StarOS and incorporates a subset of the existing StarOS supported CLI commands.

In this release the YANG model supports a limited set of ECS configuration commands via NSO. For additional information, refer to NSO user documentation and Supported StarOS ECS Configuration Commands, on page 365.

**Configuring ConfD**

To enable NETCONF protocol in StarOS, you must enable `server confd` and enter the NETCONF Protocol Configuration mode. The NETCONF Protocol Configuration mode supports two optional configuration commands.

**SSH Key Requirement**

NETCONF-ConfD support requires that a V2-RSA SSH key be configured on the local context. If an SSH key is not available, StarOS generates an error message.

*Failure: The ConfD (NETCONF) server requires an RSA key on the local context*

You can run the `show ssh key` command to verify the existence of an SSH key on the system.

If an SSH key is not available, see the Configuring SSH Options section of the Getting Started chapter in this guide.

**NETCONF Protocol Configuration Mode**

The NETCONF protocol is enabled via the Context Configuration mode `server conf` command. This command is restricted to the local context only.

```plaintext
[local] host_name# configure
[local] host_name (config)# context local
[local] host_name (config-ctx)# server confd
[local] host_name (config-confd)# ?
autosave-config - Automatically saves current configuration to the URL specified whenever a change is applied through ConfD interfaces
confd-user - Configures the default login user with full administrator rights for the ConfD server.
end - Exits configuration mode and returns to Exec Mode
exit - Exits current configuration mode, returning to previous mode
no - Enables/Disables the followed option
```
The `autosave-config` and `confd-user` keywords are optional.

To disable NETCONF protocol, run the Global Configuration mode `no server confd` command.

### autosave-config

This NETCONF Protocol Configuration mode command automatically saves the current ConfD configuration to a specified URL whenever a change is applied by NSO through the ConfD interface. By default this command is disabled.

The command syntax is: `autosave-config <url>`, where `<url>` is a URL starting with `/flash`, `/usb1`, `/hd-raid`, `/sftp`, `/<directory>`, `/<filename>`.

For additional information, see the NETCONF Protocol Configuration Mode Commands appendix of the Command Line Interface Reference.

### confd-user

This NETCONF Protocol Configuration mode command associates a username for all CLI operations via NETCONF. The user will be authenticated with verifiable credentials. This username is used for CLI logging purposes only.

The command syntax is: `confd-user <username>`, where `<username>` is an alphanumeric string of 1 to 144 characters.

For additional information, see the NETCONF Protocol Configuration Mode Commands chapter of the Command Line Interface Reference.

### Seeding and Synchronizing the CDB

After enabling `server confd` you may need to initially seed the CDB with a local copy of the configuration database (CDB) managed by ConfD on StarOS. The seeding procedure creates a CDB used by ConfD on the StarOS platform that contains all YANG-model supported configuration commands.

- **Important**
  - If you manually modify a managed object via the StarOS CLI, you must resynchronize the running configuration with the NSO by repeating the procedure described below.

---

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Run Exec mode <code>save configuration &lt;url&gt; confd</code> to save the ConfD supported StarOS configuration data to a file on the /flash device.</td>
</tr>
<tr>
<td>Step 2</td>
<td>Run Exec mode <code>show configuration error</code> to validate the saved configuration. Correct any errors before applying the configuration. Otherwise, ConfD will reject the entire configuration.</td>
</tr>
<tr>
<td>Step 3</td>
<td>Run Exec mode <code>configure confd &lt;url&gt;</code> to apply the ConfD configuration. Once the ConfD configuration is applied, the device is ready to establish NETCONF connections to the NSO management service.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Synchronize the device with your NSO. Refer to NSO user documentation for detailed information on the synchronization process.</td>
</tr>
</tbody>
</table>
Sample Configuration

The following command sequence establishes a COnfD configuration in support of NETCONF protocol.

A type v2-RSA SSH key is required for enabling server confd.

```
configure
  context local
  ssh key
  <encrypted key text>
  len 938 type v2-rsa
  server confd
    confd-user NETCONF
    autosave-config /flash/config.cfg
    #exit
  subscriber default
  exit
  aaa group default
  #exit
  gtpp group default
  #exit
  #exit
end
```

Notes:

- confd-user and autosave-config are optional. Just configuring server confd enables NETCONF support.

Verifying the Configuration

There are two Exec mode show commands that display information about the NETCONF-ConfD configuration.

**show confdmgr Command**

This command displays information about the StarOS ConfD Manager (confdmgr) process.

The syntax for this command is:

```
show confdmgr { confd { cdb | netconf | state } | subscriptions }
```

Notes:

- The **confd** keyword displays the following options:
  - *cdb* displays ConfD CDB information
  - *netconf* displays NETCONF state information
  - *state* displays current ConfD state information

- The **subscriptions** keyword displays ConfD CDB subscription information.

See below for a sample output for **show confdmgr**:

```
[local]<host_name>#$ show confdmgr
State Information
------------------
State            Started
Subscriptions    2
```
The Statistics portion of this output includes the following information:

- **Triggers** – Number of times confdmgr has requested ConfD to dump the CDB contents back into confdmgr which results in a config synchronization by SCT (Shared configuration Task).
- **Notifications** – Number of times ConfD has sent an update to confdmgr.
- **Successful Notifications** – Number of times an update received from ConfD was successfully processed. The number of successes and failures should always equal the total number of notifications.
- **Failed Notifications** – Number of times an update received from ConfD was not processed successfully.
- **Unexpected** – Number of times an unexpected condition was encountered. An error log is generated for each case.

See below for a sample output for `show confdmgr confd cdb`:

```
[local]<host_name>#$ show confdmgr confd cdb
active-charging service acs
exit
context local
  server confd
    autosave-config /tmp/ut_confdmgr_config.txt
    confd-user confd_user
exit
exit
nacm read-default permit
nacm groups group admin

! nacm groups group inspector

! nacm groups group operator

! nacm groups group secure_admin

! nacm rule-list secure_admin

! group [ secure_admin ]
  rule any-access
    action permit

rule secure_admin_server_confd
  module-name cisco-staros-cli-config
  path /context/server/confd
  access-operations create,read,update
  action permit

! v
nacm rule-list inspector

! group [ inspector ]
  rule any-access
    access-operations read
    action permit
```

**ASR 5500 System Administration Guide, StarOS Release 20**
See below for a sample output for `show confdmgr confd netconf`:

```bash
[local]<host_name>#$ show confdmgr confd netconf
netconf-state capabilities capability urn:ietf:params:netconf:base:1.0
netconf-state capabilities capability urn:ietf:params:netconf:base:1.1
netconf-state capabilities capability urn:ietf:params:netconf:capability:writable-running:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:candidate:1.0
netconf-state statistics netconf-start-time 2016-03-30T17:09:49-04:00
netconf-state statistics in-bad-hellos 0
netconf-state statistics in-sessions 0
netconf-state statistics dropped-sessions 0
netconf-state statistics in-rpcs 0
netconf-state datastores datastore candidate
```

See below for a sample output for `show confdmgr confd state`:

```bash
[local]<host_name>#$ show confdmgr confd state
```

NETCONF and ConfD

---

show confdmgr Command

See below for a sample output for `show confdmgr confd netconf`:

```bash
[local]<host_name>#$ show confdmgr confd netconf
netconf-state capabilities capability urn:ietf:params:netconf:base:1.0
netconf-state capabilities capability urn:ietf:params:netconf:base:1.1
netconf-state capabilities capability urn:ietf:params:netconf:capability:writable-running:1.0
netconf-state capabilities capability urn:ietf:params:netconf:capability:candidate:1.0
netconf-state statistics netconf-start-time 2016-03-30T17:09:49-04:00
netconf-state statistics in-bad-hellos 0
netconf-state statistics in-sessions 0
netconf-state statistics dropped-sessions 0
netconf-state statistics in-rpcs 0
netconf-state datastores datastore candidate
```
CLI SSH listen addresses:
IP PORT
---------------
0.0.0.0 2024

confd-state internal cdb datastore running
transaction-id 1461-704882-705350
filename /root/sandbox/confd/install/var/confd/cdb/A.cdb
ram-size "4.44 KiB"
read-locks 0
write-lock-set false
waiting-for-replication-sync false
confd-state internal cdb client
name confdmgr
info 12951/4
type subscriber
subscription
datastore running
priority 0
id 7
path /context
subscription
datastore running
priority 0
id 6
path /active-charging
local<host_name>#

See below for a sample output for show confdmgr subscriptions:
[local]<host_name>#$ show confdmgr subscriptions

Subscriptions:
Path Index Namespace
---------------------------------------------------
/active-charging 6 staros
/context 7 staros
[local]<host_name>#

Subscriptions are configuration points defined in the Yang model for which confdmgr wants to be notified when a change occurs. In this release there are two subscriptions: "/active-charging" and "/context".

show configuration confd Command

The confd keyword filters the output of the show configuration command to display only configuration commands that are supported by the YANG model.

show configuration confd

A sample output appears below.
[local]<host_name>#$ show configuration confd
config
  context local
  server confd
  #exit
active-charging service ecs
  ruledef rd1
    tcp any-match = TRUE
  #exit
  rulebase default
  #exit
  #exit
end
[local]<host_name>#
clear confdmgr statistics

This command clears everything listed in the "Statistics" section of the output of the show confdmgr command, including:

- Triggers
- Notifications
- Successful notifications
- Failed notifications
- Unexpected

Show Support Details (SSD)

The output of all show confdmgr commands has been added to the SSD. To filter the ConfD-related output, run show support details | grep confdmgr.

CDB Maintenance

A local copy of the ConfD Configuration Database (CDB) is managed by ConfD on StarOS. You can show and save all ConfD supported StarOS configuration commands to a URL. The confd keyword has been added to the show configuration and save configuration commands for these purposes.

After saving a ConfD-supported configuration to a URL, you can apply it directly to the CDB via the Exec mode configure confd <url> command. This command applies the contents of the file at the url to the running configuration of ConfD.

Additional detail regarding the above commands is provided below.

clear confdmgr confd cdb

This Exec mode command erases the configuration in the ConfD Configuration Database (CDB) which is used by ConfD to store configuration objects. StarOS accesses the database via ConfD-supplied APIs.

⚠️ Caution

Clearing the CDB is a terminal operation. The CDB must be repopulated afterwards.

🔍 Note

The CDB cannot be erased unless the Context Configuration mode no server confd command is run in the local context to disable ConfD and NETCONF protocol support.

The following is a sample command sequence for clearing the CDB:

```
[local]host_name(config) context local
[local]host_name(config-ctx)# no server confd
```
configure confd <url>

This Exec mode command applies the contents of the configuration script specified by the URL to the current ConfD configuration database (CDB).

A sample command sequence is provided below.

```bash
[local]host_name# configure confd <url>
```

The command syntax for this process is:

```bash
[local]host_name# configure confd <url>
```

The output of the YANG model subset of configuration commands can be viewed via the `show file url <url>` command, where `<url>` is the pathname used to save the configuration. The saved configuration file can then be applied to the CDB using the `configure confd` command.

**Supported StarOS ECS Configuration Commands**

For this release, the following StarOS ECS (Enhanced Charging System) commands are supported for the ConfD-NETCONF interface with NSO:

- `ruledef <ruledef_name>`
  - `ip server-ip-address = *`
  - `tcp ether-port = *`
  - `udp ether-port = *`
• tcp either-port-range = *
• udp ether-port range = *
• tcp-any-match = *
• udp any-match = *
• http url = *
• httpcookie = *
• http x-header = *

• group-of-ruledefs <ruledefs_group_name>
  • add-ruledef priority = *

• qos-group-of-ruledefs <group_name>
  • add-group-of-ruledef <group_of_ruledef_name>

• charging-action <charging_action_name>
  • flow-idle-timeout <seconds>
  • content-id 1
  • service-identifier <service_id>
  • billing-action egcdr

• rulebase <rulebase_name>
  • action priority <priority_number> group-of-ruledefs <ruledefs_group_name> charging-action <charging_action_name>

Note
"*" indicates support for every option following the prior keyword/value.
ICSR Checkpointing

This appendix lists and describes macro- and micro-checkpoints employed by the Interchassis Session Recovery framework. Checkpoints are exchanged between the active and standby ICSR chassis via the Service Redundancy Protocol (SRP).

The following topics are discussed:

- **Overview of Checkpointing**, page 367
- **Macro-checkpoints**, page 367
- **Micro-checkpoints**, page 369

**Overview of Checkpointing**

Interchassis Session Recovery (ICSR) provides a framework for sessmgr instance-level checkpointing within an ICSR framework. A checkpoint is a snapshot of the status of an application. Checkpointing can be used by sessmgr to push instance level information to the peer chassis.

Instance-level checkpointing sends messages to specific sessmgr instances. Each application, such as GGSN, PDSN, P-GW, S-GW or SGSN, is responsibility for encoding and decoding the checkpoint message. The ICSR framework provides the APIs for transport of the instance-level checkpoint information and associated statistics.

Macro-checkpoints contain full session information and micro-checkpoints contain only a few variables. Micro-checkpoints are sent initially from the active chassis to the standby chassis on power up and reload, and periodically thereafter. When a standby chassis receives macro-checkpoints, it clears any existing CRR (Call Recovery Record) or CLP (Call Line Pointer) related to that session, and creates a new CRR or CLP. Macro-checkpoints are also known as full checkpoints (FCs).

To conserve processing cycles and memory, dynamic and periodic updates from an active chassis to a standby chassis are done using micro-checkpoints.

The output of the Exec mode `show srp info` command displays a complete list of SRP checkpoints.

**Macro-checkpoints**

This section lists and briefly describes ICSR macro-checkpoints.
GGSN_APN ID MAPPING

This macro-checkpoint is sent from the active to the standby chassis to map APN names on the standby chassis.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a TCP connection is established between the sessmgrs and they move to READY_STATE.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **Related CLI command:** show session subsystem facility sessmgr instance <instance no> debug-info and show srp micro-checkpoint statistics

INSTANCE LEVEL CHECKPOINT

This macro-checkpoint is generated by ECS (Enhanced Charging System) to send new rules to the standby chassis. It is also used by ECS to delete or modify a rule on the standby chassis.

- **Time based:** Yes
- **Frequency:** 30 minutes
- **Event based:** Yes
- **Events:** Occurs:
  1. When a new rule is added or deleted on the active chassis.
  2. Every 30 minutes if the ECS is registered for periodic micro-checkpointing.
- **Accounting:** —
- **Delta/Cumulative:** —
- **Related CLI command:** show session subsystem facility sessmgr instance <instance no> debug-info and show srp micro-checkpoint statistics

SERVICE_ID MAPPING

This macro-checkpoint is sent from the active to the standby chassis to map Service IDs on the standby chassis.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a TCP connection is established between the sessmgrs and they move to READY_STATE.
VPNMRG_ID MAPPING

This macro-checkpoint is sent from the active to the standby chassis to map VPNs on the standby chassis.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a TCP connection is established between the sessmgrs and they move to READY_STATE.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **Related CLI command:** show session subsystem facility sessmgr instance <instance no> debug-info

Micro-checkpoints

This section lists and briefly describes the characteristics of micro-checkpoints by application category. Micro-checkpoints are listed in alphabetical order under the following categories:

- **Uncategorized,** on page 370
- **DCCA Category,** on page 371
- **ECS Category,** on page 371
- **ePDG Category,** on page 375
- **Firewall/ECS Category,** on page 377
- **GGSN Category,** on page 378
- **Gx Interface Category,** on page 380
- **NAT Category,** on page 380
- **P-GW Category,** on page 383
- **RF Interface Category,** on page 385
- **S6b Interface Category,** on page 387
- **SaMOG Category,** on page 387
Uncategorized

**SESS_UCHKPT_CMD_INVALIDATE_CRR**
This micro-checkpoint is sent to the standby chassis to clear a deleted call. It carries the Call ID and other information that must be deleted on the standby chassis.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs when a call is deleted on the active chassis.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 1
- **Related CLI command**: None

**SESS_UCKKPT_CMD_UPDATE_CLPSTATS**
This micro-checkpoint sends VoLTE data statistics.

- **Time based**: Yes
- **Frequency**: —
- **Event based**: Yes
- **Events**: Occurs during ICSR background checkpointing. A chassis switchover triggers the sending of VoLTE data stats.
- **Accounting**: —
- **Delta/Cumulative**: —
- **CMD-ID**: 4
- **Related CLI command**: None

**SESS_UCHKPT_CMD_UPDATE_IDLESECS**
This micro-checkpoint sends remaining number of seconds before idle timeout.

- **Time based**: Yes
- **Frequency**: —
- **Event based**: No
- **Events**: Occurs during ICSR background checkpointing.
DCCA Category

**SESS_UCHKPT_CMD_DCCA_SESS_INFO**

This micro-checkpoint sends Credit Control (CC) related information.

- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 2
- **Related CLI command:** None

ECS Category

**SESS_UCHKPT_CMD_ACS_CALL_INFO**

This micro-checkpoint sends critical ECS call level data.

- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 179
- **Related CLI command:** None
**SESS_UCHKPT_CMD_ACS_GX_LI_INFO**

This micro-checkpoint sources lawful intercept (LI) related information maintained by ECS.

- **Time based:** Yes
- **Frequency:** —
- **Event based:** Yes
- **Events:** Occurs whenever LI information is created or modified.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 75
- **Related CLI command:** None

**SESS_UCHKPT_CMD_ACS_SESS_INFO**

This micro-checkpoint sends ECS-level bearer-related data

- **Time based:** Yes
- **Frequency:** —
- **Event based:** Yes
- **Events:** Occurs whenever ECS bearer information is created or modified.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 33
- **Related CLI command:** None

**SESS_UCHKPT_CMD_DEL_ACS_CALL_INFO**

This micro-checkpoint notifies that a Release Call event has occurred.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever an ECS Release Call message is processed.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 188
- **Related CLI command:** —
**SESS_UCHKPT_CMD_DEL_ACS_SESS_INFO**

This micro-checkpoint notifies that a Release Bearer event has occurred.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever an ECS Release Bearer message is processed.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 187
- **Related CLI command:** None

**SESS_UCHKPT_CMD_DYNAMIC_CHRG_CA_INFO**

This micro-checkpoint sends dynamic charging action information maintained by ECS.

- **Time based:** Yes
- **Frequency:** —
- **Event based:** Yes
- **Events:** Occurs whenever dynamic charging action information is created or modified.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 141
- **Related CLI command:** None

**SESS_UCHKPT_CMD_DYNAMIC_CHRG_DEL_CA_INFO**

This micro-checkpoint notifies that a dynamic charging action has been deleted.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a dynamic charging action has been deleted.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 183
- **Related CLI command:** None
**SESS_UCHKPT_CMD_DYNAMIC_CHRG_DEL_QG_INFO**

This micro-checkpoint notifies that a dynamic QoS group has been deleted.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a dynamic QoS group has been deleted.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 182
- **Related CLI command:** None

**SESS_UCHKPT_CMD_DYNAMIC_CHRG_QG_INFO**

This micro-checkpoint sends dynamic QoS group related information maintained by ECS.

- **Time based:** Yes
- **Frequency:** —
- **Event based:** Yes
- **Events:** Occurs whenever dynamic QoS group information is created or modified.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 140
- **Related CLI command:** None

**SESS_UCHKPT_CMD_DYNAMIC_RULE_DEL_INFO**

This micro-checkpoint notifies that a dynamic rule has been deleted.

- **Time based:** No
- **Frequency:** —
- **Event based:** Yes
- **Events:** Occurs whenever a dynamic rule has been deleted.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 178
- **Related CLI command:** None
**SESS_UCHKPT_CMD_DYNAMIC_RULE_INFO**

This micro-checkpoint sources predefined and dynamic rule related information maintained by ECS.

- **Time based**: Yes
- **Frequency**: —
- **Event based**: Yes
- **Events**: Occurs whenever a dynamic rule is created or modified.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 43
- **Related CLI command**: None

**ePDG Category**

**SESS_UCHKPT_CMD_DELETE_EPDG_BEARER**

This micro-checkpoint synchronizes deleted ePDG bearers between the active and standby chassis.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: N/A
- **Accounting**: Yes
- **Delta/Cumulative**: Cumulative
- **CMD-ID**: 110
- **Related CLI command**: show srp micro-checkpoint statistics debug-info

**SESS_UCHKPT_CMD_UPDATE_EPDG_BEARER**

This micro-checkpoint synchronizes ePDG bearers between the active and standby chassis.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: No
- **Events**: N/A
- **Accounting**: Yes
- **Delta/Cumulative**: Cumulative
• CMD-ID: 110
• Related CLI command: show srp micro-checkpoint statistics debug-info

SESS_UCHKPT_CMD_UPDATE_EPDG_PEER_ADDR
This micro-checkpoint synchronizes ePDG peer addresses between the active and standby chassis.
• Time based: No
• Frequency: N/A
• Event based: Yes
• Events: —
• Accounting: Yes
• Delta/Cumulative: Cumulative
• CMD-ID: 110
• Related CLI command: show srp micro-checkpoint statistics debug-info

SESS_UCHKPT_CMD_UPDATE_EPDG_REKEY
This micro-checkpoint synchronizes ePDG rekey statistics between the active and standby chassis.
• Time based: Yes
• Frequency: 30 seconds
• Event based: No
• Events: N/A
• Accounting: Yes
• Delta/Cumulative: Cumulative
• CMD-ID: 110
• Related CLI command: show srp micro-checkpoint statistics debug-info

SESS_UCHKPT_CMD_UPDATE_EPDG_STATS
This micro-checkpoint synchronizes session statistics between the active and standby chassis.
• Time based: Yes
• Frequency: 30 seconds
• Event based: No
• Events: N/A
• Accounting: Yes
• Delta/Cumulative: Cumulative
• CMD-ID: 110
• Related CLI command: show srp micro-checkpoint statistics debug-info

Firewall/ECS Category

**SESS_UCHKPT_CMD_SFW_DEL_RULE_INFO**

This micro-checkpoint is sent when a ruledef is deleted for a bearer.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever PCRF sends a command to disable the predefined stateful firewall access rules.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 186
- **Related CLI command:** None

**SESS_UCHKPT_CMD_SFW_RULE_INFO**

This micro-checkpoint notifies the addition of dynamically enabled stateful firewall (SFW) access rules.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever PCRF sends a command to enable the predefined SFW access rules.
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative
- **CMD-ID:** 185
- **Related CLI command:** None
**GGSN Category**

**SESS_UCHKPT_CMD_GGSN_DELETE_SUB_SESS**

This micro-checkpoint sends an update when a secondary bearer is deleted.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs upon secondary bearer deletion
- **Accounting:** —
- **Delta/Cumulative:** —
- **CMD-ID:** 117
- **Related CLI command:** None

**SESS_UCHKPT_CMD_GGSN_UPDATE_RPR**

If RPR (Resilient Packet Ring) is configured in the GGSN service, an RPR timer is started during secondary bearer creation. This checkpoint is sent upon expiry of this timer.

- **Time based:** Yes
- **Frequency:** RPR timer
- **Event based:** Yes
- **Events:** Occurs when the secondary bearer creation RPR timer expires.
- **Accounting:** —
- **Delta/Cumulative:** —
- **CMD-ID:** 118
- **Related CLI command:** —

**SESS_UCHKPT_CMD_GGSN_UPDATE_SESSION**

This micro-checkpoint is sent in a Network or UE initiated update procedure except for updates that result in the following scenarios:

- Creation or deletion of the beare
- TFT change or inter-RAT handovers
- Gn-Gp handoff

Parameters associated with this micro-checkpoint are shown below.
ICSR Checkpointing

GGSN Category

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs for a network initiated or UE initiated update.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 171
- **Related CLI command**: show srp checkpoint statistics active verbose, and show session subsystem facility sessmgr instance <instance_number> debug-info.

**SESS_UCHKPT_CMD_GGSN_UPDATE_STATS**

This micro-checkpoint periodically sends session statistics.

- **Time based**: Yes
- **Frequency**: Every five minutes
- **Event based**: No
- **Events**: N/A
- **Accounting**: Yes
- **Delta/Cumulative**: Cumulative
- **CMD-ID**: 116
- **Related CLI command**: None

**SESS_UCHKPT_CMD_UPDATE_COA_PARAMS**

This micro-checkpoint updates input and output ACL parameters.

- **Time based**: —
- **Frequency**: —
- **Event based**: Yes
- **Events**: COA (Change of Authorization) response
- **Accounting**: —
- **Delta/Cumulative**: —
- **CMD-ID**: 83
- **Related CLI command**: None
Gx Interface Category

**SESS_UCHKPT_CMD_ACS_VOLUME_USAGE**

This micro-checkpoint sends volume usage over Gx accounting buckets.

- **Time based**: Yes
- **Frequency**: 4 seconds for aamgr micro-checkpoint and 18 seconds for GR micro-checkpoint
- **Event based**: No
- **Events**: Send along with macro-checkpoint
- **Accounting**: Yes
- **Delta/Cumulative**: Cumulative
- **CMD-ID**: 79
- **Related CLI command**: None

---

**SESS_UCHKPT_CMD_UPDATE_SGX_INFO**

This micro-checkpoint sends Gx session-related information.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Triggered on receiving CCA-I/U or RAR from PCRF.
- **Accounting**: Yes
- **Delta/Cumulative**: Cumulative
- **CMD-ID**: 137
- **Related CLI command**: None

---

NAT Category

**SESS_UCHKPT_CMD_GR_UPDATE_NAT_REALM_PORT_INFO1**

This micro-checkpoint is sent when a port chunk is allocated or deallocated for a subscriber sharing a NAT IP address with other subscribers. The port chunk is allocated or deallocated while data is being received for that subscriber.

- **Time based**: No
- **Frequency**: N/A
• Event based: Yes
• Events: Triggered when a new NAT port chunk is allocated or deleted.
• Accounting: No
• Delta/Cumulative: N/A
• CMD-ID: 105
• Related CLI command: None

**SESS_UCHKPT_CMD_GR_UPDATE_NAT_REALMS**

This micro-checkpoint is sent when a NAT IP address is allocated to or deallocated from a subscriber.

For an on-demand case, it is triggered when the first packet matching a particular NAT realm is received and the NAT IP address is allocated to the subscriber.

If this is not an on-demand case, the NAT IP address is allocated during call setup and this micro-checkpoint is sent.

• Time based: No
• Frequency: N/A
• Event based: Yes
• Events: Triggered when a NAT IP address is allocated to or deallocated from a subscriber.
• Accounting: No
• Delta/Cumulative: N/A
• CMD-ID: 45
• Related CLI command: None

**SESS_UCHKPT_CMD_NAT_SIP_ALG_CALL_INFO**

This micro-checkpoint is sent when a new SIP flow is created or deleted for a subscriber (while SIP data is being passed via the subscriber).

• Time based: No
• Frequency: N/A
• Event based: Yes
• Events: Triggered when a new SIP flow is created or deleted.
• Accounting: No
• Delta/Cumulative: N/A
• CMD-ID: 98
• Related CLI command: None
**SESS_UCHKPT_CMD_NAT_SIP_ALG_CONTACT_PH_INFO**

This micro-checkpoint is sent when a received SIP packet is analyzed and pinholes are created in the NAT firewall.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Triggered when a SIP packet creates pinholes in the NAT firewall.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 97
- **Related CLI command:** None

**SESS_UCHKPT_CMD_UPDATE_DSK_FLOW_CHKPT_INFO**

This micro-checkpoint is sent when a new NAT flow is created or deleted for a subscriber (while data is being passed via the subscriber).

This checkpoint is sent from a timer but it is not timer based. The timer is used to pace (10 micro-checkpoints) whenever the timer fires (granularity = 2 sec). This only occurs if there are new flows that need to be micro-checkpointed. Otherwise, no micro-micro-checkpoints are sent.

- **Time based:** No
- **Frequency:** See explanation above.
- **Event based:** Yes
- **Events:** Triggered when a new NAT flow is created or deleted.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 96
- **Related CLI command:** None

**SESS_UCHKPT_CMD_UPDATE_NAT_BYPASS_FLOW_INFO**

This micro-checkpoint is sent when NAT is enabled for a subscriber but bypass-nat (no NATing) is configured for this flow (based on a rule-match), and a new bypass flow is created.

This checkpoint is sent when the flow is both added and deleted.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
• **Events**: Triggered when a new flow with bypass-nat enabled is created or deleted.
• **Accounting**: No
• **Delta/Cumulative**: N/A
• **CMD-ID**: 60
• **Related CLI command**: None

### P-GW Category

**SESS_UCHKPT_CMD_PGW_DELETE_SUB_SESS**

Reserved for future use.

**SESS_UCHKPT_CMD_PGW_OVRCHRG_PRTCTN_INFO**

This micro-checkpoint indicates that the S-GW has set the Overcharging Protection bit in the MBR.
• **Time based**: No
• **Frequency**: N/A
• **Event based**: Yes
• **Events**: Triggered when the S-GW sets the Over Charging Protection Bit.
• **Accounting**: No
• **Delta/Cumulative**: N/A
• **CMD-ID**: 159
• **Related CLI command**: None

**SESS_UCHKPT_CMD_PGW_SGWRESTORATION_INFO**

This micro-checkpoint indicates the interval that a call will remain up when the S-GW is down.
• **Time based**: No
• **Frequency**: N/A
• **Event based**: Yes
• **Events**: Triggered when the S-GW goes into Restoration mode.
• **Accounting**: No
• **Delta/Cumulative**: N/A
• **CMD-ID**: 158
• **Related CLI command**: None
**SESS_UCHKPT_CMD_PGW_UBR_MBR_INFO**

This micro-checkpoint is sent at the end of a UBR (Update Bearer Request) or MBR (Modify Bearer Request) except when the UBR / MBR procedure results in the following scenarios:

- TFT change
- Bearer update or modification for a collapsed call
- Pure P to collapsed or collapsed to Pure P change
- Inter-technology handoff, for example, WiFi to LTE

Parameters associated with this micro-checkpoint are shown below.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** yes
- **Events:** Occurs as a result of a UBR or MBR procedure.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 193
- **Related CLI command:** show srp checkpoint statistics active verbose and show session subsystem facility sessmgr instance `<instance_number>` debug-info.

**SESS_UCHKPT_CMD_PGW_UPDATE_APN_AMBR**

Reserved for future use.

**SESS_UCHKPT_CMD_PGW_UPDATE_INFO**

Reserved for future use.

**SESS_UCHKPT_CMD_PGW_UPDATE_LI_PARAM**

This micro-checkpoint indicates the state of Lawful Intercept (LI) for this call.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Triggered when there is a change in the LI state for this call.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 151


- **Related CLI command:** None

**SESS_UCHKPT_CMD_PGW_UPDATE_PDN_COMMON_PARAM**
Reserved for future use.

**SESS_UCHKPT_CMD_PGW_UPDATE_QOS**
Reserved for future use.

**SESS_UCHKPT_CMD_PGW_UPDATE_SGW_CHANGE**
Reserved for future use.

**SESS_UCHKPT_CMD_PGW_UPDATE_STATS**
This micro-checkpoint periodically sends session statistics.

- **Time based:** Yes
- **Frequency:** Every five minutes
- **Event based:** No
- **Events:** N/A
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative
- **CMD-ID:** 65
- **Related CLI command:** None

---

**Rf Interface Category**

**SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_QCI_RF**
This micro-checkpoint indicates a change in the SDF+QCI-based Rf accounting buckets.

- **Time based:** Yes
- **Frequency:** 4 seconds for aamgr checkpoint and 18 seconds for GR checkpoint
- **Event based:** No
- **Events:** N/A
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative
**SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_QCI_RF_WITH_FC**

This micro-checkpoint indicates complete SDF+QCI-based Rf accounting buckets.

- **Time based:** Yes
- **Frequency:** 4 seconds for aamgr checkpoint and 18 seconds for GR checkpoint
- **Event based:** No
- **Events:** Sent along with macro-checkpoint.
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative

**SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_RATING_GROUP_RF**

This micro-checkpoint indicates a change in the SDF-based Rf accounting buckets.

- **Time based:** Yes
- **Frequency:** 4 seconds for aamgr checkpoint and 18 seconds for GR checkpoint
- **Event based:** No
- **Events:** N/A
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative

**SESS_UCHKPT_CMD_ACS_ACCOUNTING_TYPE_RATING_GROUP_RF_WITH_FC**

This micro-checkpoint indicates complete SDF-based Rf accounting buckets.

- **Time based:** Yes
- **Frequency:** 4 seconds for aamgr checkpoint and 18 seconds for GR checkpoint;
- **Event based:** No
- **Events:** Sent along with macro-checkpoint.
- **Accounting:** Yes
- **Delta/Cumulative:** Cumulative
S6b Interface Category

SESS_UCHKPT_CMD_S6B_INFO
This micro-checkpoint sends the Restoration Priority Indicator when reauthorization occurs over the S6b interface.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs when an Sb6 reauthorization results in a change in value of the Restoration Priority Indicator.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 163
- **Related CLI command**: None

SaMOG Category

SESS_UCHKPT_CMD_CGW_DELETE_BEARER
Reserved for future use.

SESS_UCHKPT_CMD_CGW_DELETE_PDN
This micro-checkpoint indicates a PDN connection has been deleted.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs whenever SaMOG sends a Delete-Session-Req or upon receiving a Delete-Bearer-Request.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 169
- **Related CLI command**: show subscriber samog-only full
SESS_UCHKPT_CMD_CGW_UPDATE_BEARER_QOS

This micro-checkpoint indicates a QoS update for the bearer.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs when a change in Bearer QoS is received from the P-GW due to a reauthorization (AAR Received from AAA Server) or Update-Bearer-Request.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 167
- **Related CLI command**: show subscriber samog-only full

SESS_UCHKPT_CMD_CGW_UPDATE_PDN

This micro-checkpoint indicates a PDN update for a change in APN-AMBR.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs when a change in APN-AMBR is received from the P-GW due to a reauthorization (AAR Received from AAA Server) or Update-Bearer-Request.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 168
- **Related CLI command**: show subscriber samog-only full

SESS_UCHKPT_CMD_CGW_UPDATE_STATS

Reserved for future use.

SESS_UCHKPT_CMD_CGW_UPDATE_UE_PARAM

Reserved for future use.

SESS_UCHKPT_CMD_SAMOG_ACCT_INTERIM_INFO

This micro-checkpoint is sent for a SaMOG session on receipt of an Accounting Req (INTERIM-UPDATE) from the WLC
**SESS_UCHKPT_CMD_SAMOG_ACCT_START_INFO**

This micro-checkpoint is sent for a SaMOG session on receipt of an Accounting Req (START) from the WLC (Wireless LAN Controller).

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs when a Account Req (START) request is received from the WLC.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 174
- **Related CLI command:** show subscriber samog-only full

**SESS_UCHKPT_CMD_SAMOG_EOGRE_TUNNEL_INFO**

This micro-checkpoint is sent for an Inter-RG handoff for EoGRE subscriber sessions. This checkpoint updates the VMAC Address and WLC EoGRE tunnel end-point address.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs whenever a DHCP-Discover message is received over a different EoGRE tunnel.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 201
- **Related CLI command:** show subscriber samog-only full
**SESS_UCHKPT_CMD_SAMOG_GTPV1_UPDATE_PDN_INFO**

This micro-checkpoint is sent for a SaMOG session upon receipt of an Update-PDP-Context-Req from the GGSN to update the PDN information.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs after successful SaMOG processing of an Update-PDP-Context-Req from the GGSN.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 191
- **Related CLI command:** show subscriber samog-only full

---

**SESS_UCHKPT_CMD_SAMOG_HANDOFF_AUTHEN_INFO**

This micro-checkpoint is sent for a SaMOG session that is Re-authenticating the subscriber while the subscriber session is in Handoff state.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs on completion of Re-Authentication for an existing SaMOG subscriber session currently in Handoff state.
- **Accounting:** No
- **Delta/Cumulative:** N/A
- **CMD-ID:** 176
- **Related CLI command:** show subscriber samog-only full

---

**SESS_UCHKPT_CMD_SAMOG_HANDOFF_INIT_INFO**

This micro-checkpoint is sent for a SaMOG session on receipt of an Accounting Req (STOP) from the WLC (Wireless LAN Controller).

SaMOG will delay handoff as it expects an Accounting Req (START) from the subscriber.

- **Time based:** No
- **Frequency:** N/A
- **Event based:** Yes
- **Events:** Occurs when a Account Req (STOP) request is received from the WLC.
**SESS_UCHKPT_CMD_SAMOG LI PROV_INFO**

This micro-checkpoint is sent for a SaMOG session that is on lawful intercept (LI) Active-Camp-on mode.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs after an LI trigger is received after SaMOG session has been created.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 175
- **Related CLI command**: show subscriber samog-only full

**SESS_UCHKPT_CMD_SAMOG MIPv6 TIMER_INFO**

This micro-checkpoint updates the Binding Cache Life timer and MIPv6 biding status for a SaMOG session.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
- **Events**: Occurs whenever a PMIPv6 PBU is received with a lifetime of zero from the WLC.
- **Accounting**: No
- **Delta/Cumulative**: N/A
- **CMD-ID**: 189
- **Related CLI command**: show subscriber samog-only full

**SESS_UCHKPT_CMD_SAMOG MULTI_ROUND_AUTHEN_INFO**

This micro-checkpoint is sent for a SaMOG session when SaMOG is waiting on the UE after sending an Access-Challenge while Re-authenticating the subscriber session.

- **Time based**: No
- **Frequency**: N/A
- **Event based**: Yes
• **Events:** Occurs after SaMOG sends an Access-Challenge for an existing SaMOG subscriber session during Re-authentication.

• **Accounting:** No

• **Delta/Cumulative:** N/A

• **CMD-ID:** 184

• **Related CLI command:** show subscriber samog-only full

**SESS_UCHKPT_CMD_SAMOG_REAUTHEN_INFO**

This micro-checkpoint is sent for a SaMOG session when subscriber Re-authentication is completed.

• **Time based:** No

• **Frequency:** N/A

• **Event based:** Yes

• **Events:** Occurs on completion of Re-Authentication for an existing SaMOG subscriber session.

• **Accounting:** No

• **Delta/Cumulative:** N/A

• **CMD-ID:** 172

• **Related CLI command:** show subscriber samog-only full

**SESS_UCHKPT_CMD_SAMOG_REAUTHOR_INFO**

This micro-checkpoint is sent for a SaMOG session when subscriber Re-authorization is completed.

• **Time based:** No

• **Frequency:** N/A

• **Event based:** Yes

• **Events:** Occurs on receiving and successfully processing AAR from the AAA Server to re-authorize the subscriber

• **Accounting:** No

• **Delta/Cumulative:** N/A

• **CMD-ID:** 173

• **Related CLI command:** show subscriber samog-only full
ASR 5500 SDR CLI Command Strings

This appendix identifies the CLI command strings that can be entered for a record section via the support record section command in the Global Configuration Mode. The string must be entered within double quotation marks (" ") to be recognized. The table below also indicates default and non-default strings.

For detailed command string information, refer to the Command Line Interface Reference or the online Help for the command.

The table below also indicates default and non-default strings. It reflects the output sequence of the show support collection definitions command.

Table 51: ASR 5500 SDR CLI Command Strings

<table>
<thead>
<tr>
<th>No.</th>
<th>Default SDR</th>
<th>Command String</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Enabled</td>
<td>&quot;show version verbose&quot;</td>
</tr>
<tr>
<td>1</td>
<td>Enabled</td>
<td>&quot;show clock&quot;</td>
</tr>
<tr>
<td>2</td>
<td>Enabled</td>
<td>&quot;show clock universal&quot;</td>
</tr>
<tr>
<td>3</td>
<td>Enabled</td>
<td>&quot;show configuration&quot;</td>
</tr>
<tr>
<td>4</td>
<td>Enabled</td>
<td>&quot;show_profile&quot;</td>
</tr>
<tr>
<td>5</td>
<td>Enabled</td>
<td>&quot;show context&quot;</td>
</tr>
<tr>
<td>6</td>
<td>Enabled</td>
<td>&quot;show boot&quot;</td>
</tr>
<tr>
<td>7</td>
<td>Enabled</td>
<td>&quot;show boot initial-config&quot;</td>
</tr>
<tr>
<td>8</td>
<td>Enabled</td>
<td>&quot;show system uptime&quot;</td>
</tr>
<tr>
<td>9</td>
<td>Disabled</td>
<td>&quot;show license information&quot;</td>
</tr>
<tr>
<td>10</td>
<td>Disabled</td>
<td>&quot;show license history&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>11</td>
<td>Disabled</td>
<td>&quot;show hardware inventory&quot;</td>
</tr>
<tr>
<td>12</td>
<td>Disabled</td>
<td>&quot;show hardware version&quot;</td>
</tr>
<tr>
<td>13</td>
<td>Disabled</td>
<td>&quot;show card hardware&quot;</td>
</tr>
<tr>
<td>14</td>
<td>Disabled</td>
<td>&quot;show card dhaccel hardware counters&quot;</td>
</tr>
<tr>
<td>15</td>
<td>Enabled</td>
<td>&quot;show hd raid verbose&quot;</td>
</tr>
<tr>
<td>16</td>
<td>Enabled</td>
<td>&quot;debug hdctrl mdstat&quot;</td>
</tr>
<tr>
<td>17</td>
<td>Enabled</td>
<td>&quot;debug hdctrl history&quot;</td>
</tr>
<tr>
<td>18</td>
<td>Disabled</td>
<td>&quot;debug hdctrl issas&quot;</td>
</tr>
<tr>
<td>19</td>
<td>Disabled</td>
<td>&quot;debug hdctrl mapping&quot;</td>
</tr>
<tr>
<td>20</td>
<td>Enabled</td>
<td>&quot;show hd iocnt all&quot;</td>
</tr>
<tr>
<td>21</td>
<td>Disabled</td>
<td>&quot;show hd logs all&quot;</td>
</tr>
<tr>
<td>22</td>
<td>Enabled</td>
<td>&quot;show hd smart all&quot;</td>
</tr>
<tr>
<td>23</td>
<td>Enabled</td>
<td>&quot;debug hdctrl state&quot;</td>
</tr>
<tr>
<td>24</td>
<td>Enabled</td>
<td>&quot;debug hdctrl client list&quot;</td>
</tr>
<tr>
<td>25</td>
<td>Disabled</td>
<td>&quot;show card info&quot;</td>
</tr>
<tr>
<td>26</td>
<td>Enabled</td>
<td>&quot;show card diag&quot;</td>
</tr>
<tr>
<td>27</td>
<td>Enabled</td>
<td>&quot;show card table all&quot;</td>
</tr>
<tr>
<td>28</td>
<td>Enabled</td>
<td>&quot;show port table all&quot;</td>
</tr>
<tr>
<td>29</td>
<td>Enabled</td>
<td>&quot;show port info&quot;</td>
</tr>
<tr>
<td>30</td>
<td>Enabled</td>
<td>&quot;show port utilization table&quot;</td>
</tr>
<tr>
<td>31</td>
<td>Enabled</td>
<td>&quot;show data-path congestion&quot;</td>
</tr>
<tr>
<td>32</td>
<td>Disabled</td>
<td>&quot;show npu details&quot;</td>
</tr>
<tr>
<td>33</td>
<td>Disabled</td>
<td>&quot;show lagmgr details&quot;</td>
</tr>
<tr>
<td>34</td>
<td>Enabled</td>
<td>&quot;show fans&quot;</td>
</tr>
<tr>
<td>35</td>
<td>Disabled</td>
<td>&quot;show hardware version fans&quot;</td>
</tr>
<tr>
<td>36</td>
<td>Enabled</td>
<td>&quot;show power chassis&quot;</td>
</tr>
<tr>
<td>37</td>
<td>Enabled</td>
<td>&quot;show temperature&quot;</td>
</tr>
<tr>
<td>38</td>
<td>Disabled</td>
<td>&quot;show timing&quot;</td>
</tr>
<tr>
<td>39</td>
<td>Disabled</td>
<td>&quot;show alarm audible&quot;</td>
</tr>
<tr>
<td>40</td>
<td>Disabled</td>
<td>&quot;show alarm central-office&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>------------</td>
<td>---------------------------------------</td>
</tr>
<tr>
<td>41</td>
<td>Disabled</td>
<td>&quot;show alarm outstanding&quot;</td>
</tr>
<tr>
<td>42</td>
<td>Disabled</td>
<td>&quot;show alarm statistics&quot;</td>
</tr>
<tr>
<td>43</td>
<td>Enabled</td>
<td>&quot;show cpu table&quot;</td>
</tr>
<tr>
<td>44</td>
<td>Disabled</td>
<td>&quot;show cpu info verbose&quot;</td>
</tr>
<tr>
<td>45</td>
<td>Enabled</td>
<td>&quot;show cpu errors verbose&quot;</td>
</tr>
<tr>
<td>46</td>
<td>Enabled</td>
<td>&quot;show cpu performance verbose&quot;</td>
</tr>
<tr>
<td>47</td>
<td>Disabled</td>
<td>&quot;show resources&quot;</td>
</tr>
<tr>
<td>48</td>
<td>Disabled</td>
<td>&quot;show task table&quot;</td>
</tr>
<tr>
<td>49</td>
<td>Disabled</td>
<td>&quot;show task memory&quot;</td>
</tr>
<tr>
<td>50</td>
<td>Disabled</td>
<td>&quot;show task memory max&quot;</td>
</tr>
<tr>
<td>51</td>
<td>Disabled</td>
<td>&quot;show task resources&quot;</td>
</tr>
<tr>
<td>52</td>
<td>Disabled</td>
<td>&quot;show task resources max&quot;</td>
</tr>
<tr>
<td>53</td>
<td>Enabled</td>
<td>&quot;show crash list&quot;</td>
</tr>
<tr>
<td>54</td>
<td>Enabled</td>
<td>&quot;show crash all&quot;</td>
</tr>
<tr>
<td>55</td>
<td>Disabled</td>
<td>&quot;show persistdump list&quot;</td>
</tr>
<tr>
<td>56</td>
<td>Disabled</td>
<td>&quot;show persistdump display&quot;</td>
</tr>
<tr>
<td>57</td>
<td>Enabled</td>
<td>&quot;show snmp trap history verbose&quot;</td>
</tr>
<tr>
<td>58</td>
<td>Disabled</td>
<td>&quot;show snmp trap statistics verbose&quot;</td>
</tr>
<tr>
<td>59</td>
<td>Enabled</td>
<td>&quot;show logs&quot;</td>
</tr>
<tr>
<td>63</td>
<td>Disabled</td>
<td>&quot;show messenger settings&quot;</td>
</tr>
<tr>
<td>64</td>
<td>Enabled</td>
<td>&quot;show messenger nameservice&quot;</td>
</tr>
<tr>
<td>65</td>
<td>Enabled</td>
<td>&quot;show messenger statistics&quot;</td>
</tr>
<tr>
<td>66</td>
<td>Enabled</td>
<td>&quot;show messenger bounces&quot;</td>
</tr>
<tr>
<td>67</td>
<td>Disabled</td>
<td>&quot;debug limits checkup detailed&quot;</td>
</tr>
<tr>
<td>68</td>
<td>Disabled</td>
<td>&quot;show plugin&quot;</td>
</tr>
<tr>
<td>69</td>
<td>Disabled</td>
<td>&quot;show module&quot;</td>
</tr>
<tr>
<td>70</td>
<td>Disabled</td>
<td>&quot;show ppp statistics&quot;</td>
</tr>
<tr>
<td>71</td>
<td>Disabled</td>
<td>&quot;show rsvp statistics&quot;</td>
</tr>
<tr>
<td>72</td>
<td>Enabled</td>
<td>&quot;show session disconnect-reasons verbose&quot;</td>
</tr>
<tr>
<td>73</td>
<td>Disabled</td>
<td>&quot;show apn statistics all&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>74</td>
<td>Disabled</td>
<td>&quot;show ipsg statistics&quot;</td>
</tr>
<tr>
<td>75</td>
<td>Disabled</td>
<td>&quot;show pdsn-service all&quot;</td>
</tr>
<tr>
<td>76</td>
<td>Disabled</td>
<td>&quot;show hsgw-service all&quot;</td>
</tr>
<tr>
<td>77</td>
<td>Disabled</td>
<td>&quot;show hsgw-service statistics all&quot;</td>
</tr>
<tr>
<td>78</td>
<td>Disabled</td>
<td>&quot;show epdg-service all counters&quot;</td>
</tr>
<tr>
<td>79</td>
<td>Disabled</td>
<td>&quot;show epdg-service statistics&quot;</td>
</tr>
<tr>
<td>80</td>
<td>Disabled</td>
<td>&quot;show fa-service all&quot;</td>
</tr>
<tr>
<td>81</td>
<td>Disabled</td>
<td>&quot;show ha-service all&quot;</td>
</tr>
<tr>
<td>82</td>
<td>Disabled</td>
<td>&quot;show mag-service all&quot;</td>
</tr>
<tr>
<td>83</td>
<td>Disabled</td>
<td>&quot;show mipv6ha-service all&quot;</td>
</tr>
<tr>
<td>84</td>
<td>Disabled</td>
<td>&quot;show lma-service all&quot;</td>
</tr>
<tr>
<td>85</td>
<td>Disabled</td>
<td>&quot;show dhcp-service all&quot;</td>
</tr>
<tr>
<td>86</td>
<td>Disabled</td>
<td>&quot;show sgsn-service all&quot;</td>
</tr>
<tr>
<td>87</td>
<td>Disabled</td>
<td>&quot;show sgsn sessmgr all memory statistics&quot;</td>
</tr>
<tr>
<td>88</td>
<td>Disabled</td>
<td>&quot;show operator-policy all&quot;</td>
</tr>
<tr>
<td>89</td>
<td>Disabled</td>
<td>&quot;show call-control-profile all&quot;</td>
</tr>
<tr>
<td>90</td>
<td>Disabled</td>
<td>&quot;show apn-profile all&quot;</td>
</tr>
<tr>
<td>91</td>
<td>Disabled</td>
<td>&quot;show imei-profile all&quot;</td>
</tr>
<tr>
<td>92</td>
<td>Disabled</td>
<td>&quot;show gprs-service all&quot;</td>
</tr>
<tr>
<td>93</td>
<td>Disabled</td>
<td>&quot;show iups-service all&quot;</td>
</tr>
<tr>
<td>94</td>
<td>Disabled</td>
<td>&quot;show sgtp-service all&quot;</td>
</tr>
<tr>
<td>95</td>
<td>Disabled</td>
<td>&quot;show map-service all&quot;</td>
</tr>
<tr>
<td>96</td>
<td>Disabled</td>
<td>&quot;show gs-service all&quot;</td>
</tr>
<tr>
<td>97</td>
<td>Disabled</td>
<td>&quot;show ggsn-service all&quot;</td>
</tr>
<tr>
<td>98</td>
<td>Disabled</td>
<td>&quot;show ggsn-service sgsn-table&quot;</td>
</tr>
<tr>
<td>105</td>
<td>Disabled</td>
<td>&quot;show lac-service all&quot;</td>
</tr>
<tr>
<td>106</td>
<td>Disabled</td>
<td>&quot;show ins-service all&quot;</td>
</tr>
<tr>
<td>107</td>
<td>Disabled</td>
<td>&quot;show pdnclosedrp-service all&quot;</td>
</tr>
<tr>
<td>108</td>
<td>Enabled</td>
<td>&quot;show subscriber summary&quot;</td>
</tr>
<tr>
<td>109</td>
<td>Enabled</td>
<td>&quot;show connproxy sockets all&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>110</td>
<td>Disabled</td>
<td>&quot;show session progress&quot;</td>
</tr>
<tr>
<td>111</td>
<td>Disabled</td>
<td>&quot;show session subsystem data-info verbose&quot;</td>
</tr>
<tr>
<td>112</td>
<td>Disabled</td>
<td>&quot;show session subsystem full data-info&quot;</td>
</tr>
<tr>
<td>113</td>
<td>Disabled</td>
<td>&quot;show session subsystem facility sessmgr all debug-info&quot;</td>
</tr>
<tr>
<td>114</td>
<td>Disabled</td>
<td>&quot;show sesctrl config-reconciliation statistics&quot;</td>
</tr>
<tr>
<td>115</td>
<td>Disabled</td>
<td>&quot;show rp statistics&quot;</td>
</tr>
<tr>
<td>116</td>
<td>Disabled</td>
<td>&quot;show mipfa statistics&quot;</td>
</tr>
<tr>
<td>117</td>
<td>Disabled</td>
<td>&quot;show mipha statistics&quot;</td>
</tr>
<tr>
<td>118</td>
<td>Disabled</td>
<td>&quot;show mipv6ha statistics&quot;</td>
</tr>
<tr>
<td>119</td>
<td>Disabled</td>
<td>&quot;show lma-service statistics&quot;</td>
</tr>
<tr>
<td>120</td>
<td>Disabled</td>
<td>&quot;show mag-service statistics&quot;</td>
</tr>
<tr>
<td>121</td>
<td>Disabled</td>
<td>&quot;show cli configuration-monitor&quot;</td>
</tr>
<tr>
<td>122</td>
<td>Enabled</td>
<td>&quot;show srp info&quot;</td>
</tr>
<tr>
<td>123</td>
<td>Enabled</td>
<td>&quot;show srp checkpoint statistics&quot;</td>
</tr>
<tr>
<td>124</td>
<td>Disabled</td>
<td>&quot;show srp checkpoint statistics verbose&quot;</td>
</tr>
<tr>
<td>125</td>
<td>Disabled</td>
<td>&quot;show srp checkpoint statistics sessmgr all&quot;</td>
</tr>
<tr>
<td>126</td>
<td>Disabled</td>
<td>&quot;show srp checkpoint statistics ipsecmgr all&quot;</td>
</tr>
<tr>
<td>127</td>
<td>Enabled</td>
<td>&quot;show srp checkpoint statistics sessmgr all write-list-stats&quot;</td>
</tr>
<tr>
<td>128</td>
<td>Disabled</td>
<td>&quot;show srp monitor&quot;</td>
</tr>
<tr>
<td>129</td>
<td>Enabled</td>
<td>&quot;show srp monitor all&quot;</td>
</tr>
<tr>
<td>130</td>
<td>Disabled</td>
<td>&quot;show srp monitor diameter debug&quot;</td>
</tr>
<tr>
<td>131</td>
<td>Enabled</td>
<td>&quot;show srp statistics&quot;</td>
</tr>
<tr>
<td>132</td>
<td>Disabled</td>
<td>&quot;show srp call-loss statistics&quot;</td>
</tr>
<tr>
<td>133</td>
<td>Disabled</td>
<td>&quot;show srp audit-statistics&quot;</td>
</tr>
<tr>
<td>134</td>
<td>Disabled</td>
<td>&quot;show gtpe statistics verbose&quot;</td>
</tr>
<tr>
<td>135</td>
<td>Enabled</td>
<td>&quot;show gtpu statistics verbose&quot;</td>
</tr>
<tr>
<td>136</td>
<td>Enabled</td>
<td>&quot;show gtpu debug-info&quot;</td>
</tr>
<tr>
<td>137</td>
<td>Enabled</td>
<td>&quot;show gmm-sm statistics verbose&quot;</td>
</tr>
<tr>
<td>138</td>
<td>Enabled</td>
<td>&quot;show sgtpc statistics verbose&quot;</td>
</tr>
<tr>
<td>139</td>
<td>Enabled</td>
<td>&quot;show sgtpu statistics&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>140</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all sctp asp all status peer-server all peer-server-process all verbose&quot;</td>
</tr>
<tr>
<td>141</td>
<td>Enabled</td>
<td>&quot;show ss7-routing-domain all sctp asp all statistics gen&quot;</td>
</tr>
<tr>
<td>142</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all m3ua status peer-server all&quot;</td>
</tr>
<tr>
<td>143</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all m3ua statistics peer-server all peer-server-process all&quot;</td>
</tr>
<tr>
<td>144</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all qsaal statistics linkset all link all&quot;</td>
</tr>
<tr>
<td>145</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all sscf statistics linkset all link all&quot;</td>
</tr>
<tr>
<td>146</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all mtp3 status linkset all link all&quot;</td>
</tr>
<tr>
<td>147</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all mtp3 statistics gen&quot;</td>
</tr>
<tr>
<td>148</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all mtp3 statistics linkset all link all&quot;</td>
</tr>
<tr>
<td>149</td>
<td>Disabled</td>
<td>&quot;show ss7-routing-domain all routes&quot;</td>
</tr>
<tr>
<td>150</td>
<td>Disabled</td>
<td>&quot;show sc cp-network all status all&quot;</td>
</tr>
<tr>
<td>151</td>
<td>Disabled</td>
<td>&quot;show global-title-translation association&quot;</td>
</tr>
<tr>
<td>152</td>
<td>Disabled</td>
<td>&quot;show global-title-translation address-map&quot;</td>
</tr>
<tr>
<td>153</td>
<td>Enabled</td>
<td>&quot;show egtpe peers&quot;</td>
</tr>
<tr>
<td>154</td>
<td>Disabled</td>
<td>&quot;show egtpe statistics interface mme&quot;</td>
</tr>
<tr>
<td>155</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface sgsn&quot;</td>
</tr>
<tr>
<td>156</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface sgw-ingress&quot;</td>
</tr>
<tr>
<td>157</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface sgw-egress&quot;</td>
</tr>
<tr>
<td>158</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface pgw-ingress&quot;</td>
</tr>
<tr>
<td>159</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface cgw-egress&quot;</td>
</tr>
<tr>
<td>160</td>
<td>Enabled</td>
<td>&quot;show egtpe statistics interface epdg-egress&quot;</td>
</tr>
<tr>
<td>161</td>
<td>Disabled</td>
<td>&quot;show egt-service all&quot;</td>
</tr>
<tr>
<td>162</td>
<td>Disabled</td>
<td>&quot;show gtpu-service all&quot;</td>
</tr>
<tr>
<td>163</td>
<td>Disabled</td>
<td>&quot;show pgw-service all&quot;</td>
</tr>
<tr>
<td>164</td>
<td>Disabled</td>
<td>&quot;show sgw-service all&quot;</td>
</tr>
<tr>
<td>165</td>
<td>Disabled</td>
<td>&quot;show saegw-service all&quot;</td>
</tr>
<tr>
<td>166</td>
<td>Disabled</td>
<td>&quot;show henbgw-access-service statistics&quot;</td>
</tr>
<tr>
<td>167</td>
<td>Disabled</td>
<td>&quot;show henbgw-network-service statistics&quot;</td>
</tr>
<tr>
<td>168</td>
<td>Disabled</td>
<td>&quot;show mme-service all&quot;</td>
</tr>
<tr>
<td>169</td>
<td>Disabled</td>
<td>&quot;show mme-service enodeb-association full all&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>170</td>
<td>Disabled</td>
<td>&quot;show mme-service statistics debug&quot;</td>
</tr>
<tr>
<td>171</td>
<td>Disabled</td>
<td>&quot;show mme-service db statistics&quot;</td>
</tr>
<tr>
<td>172</td>
<td>Disabled</td>
<td>&quot;show sgs-service all&quot;</td>
</tr>
<tr>
<td>173</td>
<td>Disabled</td>
<td>&quot;show sgs-service vlr-status full&quot;</td>
</tr>
<tr>
<td>174</td>
<td>Disabled</td>
<td>&quot;show sgs-service statistics all&quot;</td>
</tr>
<tr>
<td>175</td>
<td>Enabled</td>
<td>&quot;show sgw-service statistics all&quot;</td>
</tr>
<tr>
<td>176</td>
<td>Disabled</td>
<td>&quot;show saegw-service statistics all verbose&quot;</td>
</tr>
<tr>
<td>177</td>
<td>Disabled</td>
<td>&quot;show saegw-service statistics all function sgw verbose&quot;</td>
</tr>
<tr>
<td>178</td>
<td>Disabled</td>
<td>&quot;show saegw-service statistics all function pgw verbose&quot;</td>
</tr>
<tr>
<td>179</td>
<td>Enabled</td>
<td>&quot;show pgw-service statistics all&quot;</td>
</tr>
<tr>
<td>180</td>
<td>Disabled</td>
<td>&quot;show sccp statistics&quot;</td>
</tr>
<tr>
<td>181</td>
<td>Disabled</td>
<td>&quot;show tcap statistics&quot;</td>
</tr>
<tr>
<td>182</td>
<td>Disabled</td>
<td>&quot;show map statistics&quot;</td>
</tr>
<tr>
<td>183</td>
<td>Disabled</td>
<td>&quot;show sms statistics&quot;</td>
</tr>
<tr>
<td>184</td>
<td>Disabled</td>
<td>&quot;show pdg-service statistics&quot;</td>
</tr>
<tr>
<td>185</td>
<td>Disabled</td>
<td>&quot;show hnbgw sessmgr all memory statistics&quot;</td>
</tr>
<tr>
<td>186</td>
<td>Disabled</td>
<td>&quot;show hnbgw sessmgr all internal statistics&quot;</td>
</tr>
<tr>
<td>187</td>
<td>Disabled</td>
<td>&quot;show hnbgw disconnect-reasons&quot;</td>
</tr>
<tr>
<td>188</td>
<td>Disabled</td>
<td>&quot;show cs-network statistics&quot;</td>
</tr>
<tr>
<td>189</td>
<td>Disabled</td>
<td>&quot;show ps-network statistics&quot;</td>
</tr>
<tr>
<td>190</td>
<td>Disabled</td>
<td>&quot;show hnbgw statistics&quot;</td>
</tr>
<tr>
<td>191</td>
<td>Disabled</td>
<td>&quot;show hnbgw counters&quot;</td>
</tr>
<tr>
<td>192</td>
<td>Disabled</td>
<td>&quot;show demux-mgr statistics hnbmgr full&quot;</td>
</tr>
<tr>
<td>193</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics bngmgr all&quot;</td>
</tr>
<tr>
<td>194</td>
<td>Disabled</td>
<td>&quot;show alcap statistics&quot;</td>
</tr>
<tr>
<td>195</td>
<td>Disabled</td>
<td>&quot;show pdg-service statistics micro-tunnel&quot;</td>
</tr>
<tr>
<td>196</td>
<td>Disabled</td>
<td>&quot;show pdg-service statistics transport&quot;</td>
</tr>
<tr>
<td>197</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics a11mgr all&quot;</td>
</tr>
<tr>
<td>198</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics famgr all&quot;</td>
</tr>
<tr>
<td>199</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics hmsgr all&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>200</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics l2tpmgr all&quot;</td>
</tr>
<tr>
<td>201</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics ipsgmgr all&quot;</td>
</tr>
<tr>
<td>202</td>
<td>Enabled</td>
<td>&quot;show demuxmgr statistics sgtpcmgr all&quot;</td>
</tr>
<tr>
<td>203</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics imsimgr all&quot;</td>
</tr>
<tr>
<td>204</td>
<td>Enabled</td>
<td>&quot;show demuxmgr statistics gtpcmgr all&quot;</td>
</tr>
<tr>
<td>205</td>
<td>Enabled</td>
<td>&quot;show demuxmgr statistics egtpipnmgr all&quot;</td>
</tr>
<tr>
<td>206</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics egtpepgmgr all&quot;</td>
</tr>
<tr>
<td>207</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics pdgdmgr all&quot;</td>
</tr>
<tr>
<td>208</td>
<td>Enabled</td>
<td>&quot;show demuxmgr statistics gtpumgr all&quot;</td>
</tr>
<tr>
<td>209</td>
<td>Disabled</td>
<td>&quot;show bcmcs statistics all&quot;</td>
</tr>
<tr>
<td>210</td>
<td>Enabled</td>
<td>&quot;show linkmgr all parser statistics all&quot;</td>
</tr>
<tr>
<td>211</td>
<td>Disabled</td>
<td>&quot;show gtpp accounting servers&quot;</td>
</tr>
<tr>
<td>212</td>
<td>Disabled</td>
<td>&quot;show gtpp statistics verbose&quot;</td>
</tr>
<tr>
<td>213</td>
<td>Disabled</td>
<td>&quot;show gtpp counters all&quot;</td>
</tr>
<tr>
<td>214</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server&quot;</td>
</tr>
<tr>
<td>215</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server statistics verbose&quot;</td>
</tr>
<tr>
<td>216</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server local file statistics verbose&quot;</td>
</tr>
<tr>
<td>217</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server local file counters all&quot;</td>
</tr>
<tr>
<td>218</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server streaming file statistics verbose&quot;</td>
</tr>
<tr>
<td>219</td>
<td>Disabled</td>
<td>&quot;show gtpp storage-server streaming file counters all&quot;</td>
</tr>
<tr>
<td>220</td>
<td>Disabled</td>
<td>&quot;show gtpp group all&quot;</td>
</tr>
<tr>
<td>221</td>
<td>Enabled</td>
<td>&quot;show hd-storage-policy statistics all verbose&quot;</td>
</tr>
<tr>
<td>222</td>
<td>Enabled</td>
<td>&quot;show hd-storage-policy counters all verbose&quot;</td>
</tr>
<tr>
<td>223</td>
<td>Disabled</td>
<td>&quot;show dhcp statistics verbose&quot;</td>
</tr>
<tr>
<td>224</td>
<td>Disabled</td>
<td>&quot;show npu table&quot;</td>
</tr>
<tr>
<td>225</td>
<td>Disabled</td>
<td>&quot;show npu sf hw-info&quot;</td>
</tr>
<tr>
<td>228</td>
<td>Enabled</td>
<td>&quot;show npu asr5500&quot;</td>
</tr>
<tr>
<td>229</td>
<td>Disabled</td>
<td>&quot;show l2tp statistics&quot;</td>
</tr>
<tr>
<td>230</td>
<td>Enabled</td>
<td>&quot;show fabric asr5500&quot;</td>
</tr>
<tr>
<td>231</td>
<td>Enabled</td>
<td>&quot;show vpn subsystem facility vpmgr&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>232</td>
<td>Enabled</td>
<td>&quot;show session recovery status verbose&quot;</td>
</tr>
<tr>
<td>233</td>
<td>Enabled</td>
<td>&quot;show clock all&quot;</td>
</tr>
<tr>
<td>234</td>
<td>Disabled</td>
<td>&quot;show sntp statistics verbose&quot;</td>
</tr>
<tr>
<td>235</td>
<td>Disabled</td>
<td>&quot;show llc statistics verbose&quot;</td>
</tr>
<tr>
<td>236</td>
<td>Disabled</td>
<td>&quot;show bssgp statistics verbose&quot;</td>
</tr>
<tr>
<td>237</td>
<td>Disabled</td>
<td>&quot;show bssap+ statistics verbose&quot;</td>
</tr>
<tr>
<td>238</td>
<td>Disabled</td>
<td>&quot;show network-service-entity ip-config&quot;</td>
</tr>
<tr>
<td>239</td>
<td>Disabled</td>
<td>&quot;show network-service-entity fr-config&quot;</td>
</tr>
<tr>
<td>240</td>
<td>Disabled</td>
<td>&quot;show gprsns statistics sns-msg-stats&quot;</td>
</tr>
<tr>
<td>241</td>
<td>Disabled</td>
<td>&quot;show radius authentication servers detail&quot;</td>
</tr>
<tr>
<td>242</td>
<td>Disabled</td>
<td>&quot;show radius accounting servers detail&quot;</td>
</tr>
<tr>
<td>243</td>
<td>Enabled</td>
<td>&quot;show radius counters all&quot;</td>
</tr>
<tr>
<td>244</td>
<td>Enabled</td>
<td>&quot;debug console card cpu tail 4000 only&quot;</td>
</tr>
<tr>
<td>245</td>
<td>Enabled</td>
<td>&quot;show rct stats&quot;</td>
</tr>
<tr>
<td>246</td>
<td>Enabled</td>
<td>&quot;show heartbeat stats hatcpus&quot;</td>
</tr>
<tr>
<td>247</td>
<td>Disabled</td>
<td>&quot;show ntp associations all&quot;</td>
</tr>
<tr>
<td>248</td>
<td>Disabled</td>
<td>&quot;show npu details&quot;</td>
</tr>
<tr>
<td>249</td>
<td>Disabled</td>
<td>&quot;show active-charging service all&quot;</td>
</tr>
<tr>
<td>250</td>
<td>Disabled</td>
<td>&quot;show active-charging tcp-proxy statistics all verbose debug-info&quot;</td>
</tr>
<tr>
<td>251</td>
<td>Disabled</td>
<td>&quot;show active-charging edr-udr-file flow-control-counters verbose debug-only&quot;</td>
</tr>
<tr>
<td>252</td>
<td>Disabled</td>
<td>&quot;show active-charging service statistics&quot;</td>
</tr>
<tr>
<td>253</td>
<td>Disabled</td>
<td>&quot;show active-charging analyzer statistics&quot;</td>
</tr>
<tr>
<td>254</td>
<td>Disabled</td>
<td>&quot;show active-charging dns-learnt-ip-addresses statistics sessmgr all verbose&quot;</td>
</tr>
<tr>
<td>255</td>
<td>Disabled</td>
<td>&quot;show active-charging analyzer statistics name ip verbose&quot;</td>
</tr>
<tr>
<td>256</td>
<td>Disabled</td>
<td>&quot;show active-charging analyzer statistics name ipv6 verbose&quot;</td>
</tr>
<tr>
<td>257</td>
<td>Disabled</td>
<td>&quot;show active-charging analyzer statistics name tcp verbose&quot;</td>
</tr>
<tr>
<td>258</td>
<td>Disabled</td>
<td>&quot;show active-charging analyzer statistics name http verbose&quot;</td>
</tr>
<tr>
<td>259</td>
<td>Disabled</td>
<td>&quot;show active-charging charging-action statistics&quot;</td>
</tr>
<tr>
<td>260</td>
<td>Disabled</td>
<td>&quot;show active-charging rulebase statistics&quot;</td>
</tr>
<tr>
<td>261</td>
<td>Disabled</td>
<td>&quot;show active-charging ruledef statistics all charging&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>262</td>
<td>Enabled</td>
<td>&quot;show active-charging ruledef statistics all firewall wide&quot;</td>
</tr>
<tr>
<td>263</td>
<td>Disabled</td>
<td>&quot;show active-charging regex status all&quot;</td>
</tr>
<tr>
<td>264</td>
<td>Disabled</td>
<td>&quot;show active-charging regex statistics memory summary&quot;</td>
</tr>
<tr>
<td>265</td>
<td>Disabled</td>
<td>&quot;show active-charging regex statistics ruledef summary&quot;</td>
</tr>
<tr>
<td>266</td>
<td>Disabled</td>
<td>&quot;show active-charging edr-format statistics&quot;</td>
</tr>
<tr>
<td>267</td>
<td>Disabled</td>
<td>&quot;show active-charging subsystem all debug-only&quot;</td>
</tr>
<tr>
<td>268</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats cumulative all-flows&quot;</td>
</tr>
<tr>
<td>269</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats cumulative http&quot;</td>
</tr>
<tr>
<td>270</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats cumulative ip&quot;</td>
</tr>
<tr>
<td>271</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats cumulative tcp&quot;</td>
</tr>
<tr>
<td>272</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats cumulative udp&quot;</td>
</tr>
<tr>
<td>273</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats max-simultaneous-flows all-flows&quot;</td>
</tr>
<tr>
<td>274</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats max-simultaneous-flows http&quot;</td>
</tr>
<tr>
<td>275</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats max-simultaneous-flows ip&quot;</td>
</tr>
<tr>
<td>276</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats max-simultaneous-flows tcp&quot;</td>
</tr>
<tr>
<td>277</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats max-simultaneous-flows udp&quot;</td>
</tr>
<tr>
<td>278</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats duration-based all-flows&quot;</td>
</tr>
<tr>
<td>279</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats duration-based tcp&quot;</td>
</tr>
<tr>
<td>280</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats duration-based udp&quot;</td>
</tr>
<tr>
<td>281</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show flow-stats lifetime-based all-flows&quot;</td>
</tr>
<tr>
<td>282</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show p2p detection-params sct&quot;</td>
</tr>
<tr>
<td>283</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show rule-optimization-information&quot;</td>
</tr>
<tr>
<td>284</td>
<td>Disabled</td>
<td>&quot;debug sessmgr charging-service show-stats all&quot;</td>
</tr>
<tr>
<td>285</td>
<td>Disabled</td>
<td>&quot;debug acsmgr show memory usage&quot;</td>
</tr>
<tr>
<td>286</td>
<td>Disabled</td>
<td>&quot;debug aaamgr show memory usage&quot;</td>
</tr>
<tr>
<td>287</td>
<td>Disabled</td>
<td>&quot;show active-charging credit-control statistics debug-info&quot;</td>
</tr>
<tr>
<td>288</td>
<td>Disabled</td>
<td>&quot;show active-charging credit-control session-states&quot;</td>
</tr>
<tr>
<td>289</td>
<td>Disabled</td>
<td>&quot;show active-charging credit-control statistics&quot;</td>
</tr>
<tr>
<td>290</td>
<td>Disabled</td>
<td>&quot;show diameter endpoints all&quot;</td>
</tr>
<tr>
<td>291</td>
<td>Disabled</td>
<td>&quot;show diameter endpoints all debug-info&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>-----------------------------------------------------</td>
</tr>
<tr>
<td>292</td>
<td>Disabled</td>
<td>&quot;show diameter route table debug-info&quot;</td>
</tr>
<tr>
<td>293</td>
<td>Disabled</td>
<td>&quot;show diameter peers full debug&quot;</td>
</tr>
<tr>
<td>294</td>
<td>Disabled</td>
<td>&quot;show diameter aaa-statistics&quot;</td>
</tr>
<tr>
<td>295</td>
<td>Disabled</td>
<td>&quot;show diameter aaa-statistics all&quot;</td>
</tr>
<tr>
<td>296</td>
<td>Disabled</td>
<td>&quot;show diameter aaa-statistics debug-info&quot;</td>
</tr>
<tr>
<td>297</td>
<td>Disabled</td>
<td>&quot;show diameter accounting servers debug-info&quot;</td>
</tr>
<tr>
<td>298</td>
<td>Disabled</td>
<td>&quot;show diameter authentication servers debug-info&quot;</td>
</tr>
<tr>
<td>299</td>
<td>Disabled</td>
<td>&quot;show diameter statistics&quot;</td>
</tr>
<tr>
<td>300</td>
<td>Disabled</td>
<td>&quot;show diameter statistics debug-info&quot;</td>
</tr>
<tr>
<td>301</td>
<td>Disabled</td>
<td>&quot;show diameter statistics proxy&quot;</td>
</tr>
<tr>
<td>302</td>
<td>Disabled</td>
<td>&quot;show diameter statistics proxy debug-info&quot;</td>
</tr>
<tr>
<td>303</td>
<td>Disabled</td>
<td>&quot;show diameter dynamic-dictionary all contents&quot;</td>
</tr>
<tr>
<td>304</td>
<td>Disabled</td>
<td>&quot;show active-charging edr-udr-file statistics&quot;</td>
</tr>
<tr>
<td>305</td>
<td>Disabled</td>
<td>&quot;show active-charging firewall statistics debug-info&quot;</td>
</tr>
<tr>
<td>306</td>
<td>Disabled</td>
<td>&quot;show active-charging nat statistics&quot;</td>
</tr>
<tr>
<td>307</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics asngwmgr all&quot;</td>
</tr>
<tr>
<td>308</td>
<td>Disabled</td>
<td>&quot;show asngw-service all&quot;</td>
</tr>
<tr>
<td>309</td>
<td>Disabled</td>
<td>&quot;show asngw-service statistics verbose&quot;</td>
</tr>
<tr>
<td>310</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics asnpcmgr all&quot;</td>
</tr>
<tr>
<td>311</td>
<td>Disabled</td>
<td>&quot;show asnpc-service all&quot;</td>
</tr>
<tr>
<td>312</td>
<td>Disabled</td>
<td>&quot;show asnpc-service statistics verbose&quot;</td>
</tr>
<tr>
<td>313</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics phsgwmgr all&quot;</td>
</tr>
<tr>
<td>314</td>
<td>Disabled</td>
<td>&quot;show phsgw-service all&quot;</td>
</tr>
<tr>
<td>315</td>
<td>Disabled</td>
<td>&quot;show phsgw-service statistics verbose&quot;</td>
</tr>
<tr>
<td>316</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics phspcmgr all&quot;</td>
</tr>
<tr>
<td>317</td>
<td>Disabled</td>
<td>&quot;show phspc-service all&quot;</td>
</tr>
<tr>
<td>318</td>
<td>Disabled</td>
<td>&quot;show phspc-service statistics verbose&quot;</td>
</tr>
<tr>
<td>319</td>
<td>Disabled</td>
<td>&quot;show demuxmgr statistics magmgr all&quot;</td>
</tr>
<tr>
<td>320</td>
<td>Disabled</td>
<td>&quot;show active-charging content-filtering category policy-id all&quot;</td>
</tr>
<tr>
<td>321</td>
<td>Disabled</td>
<td>&quot;show content-filtering category database all verbose&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>322</td>
<td>Disabled</td>
<td>&quot;show content-filtering category database facility srdbmgr all verbose&quot;</td>
</tr>
<tr>
<td>323</td>
<td>Disabled</td>
<td>&quot;show content-filtering category statistics&quot;</td>
</tr>
<tr>
<td>324</td>
<td>Disabled</td>
<td>&quot;show content-filtering category statistics facility srdbmgr all&quot;</td>
</tr>
<tr>
<td>325</td>
<td>Disabled</td>
<td>&quot;show active-charging content-filtering category statistics&quot;</td>
</tr>
<tr>
<td>326</td>
<td>Disabled</td>
<td>&quot;show active-charging content-filtering server-group statistics verbose&quot;</td>
</tr>
<tr>
<td>327</td>
<td>Disabled</td>
<td>&quot;show active-charging url-blacklisting statistics&quot;</td>
</tr>
<tr>
<td>328</td>
<td>Disabled</td>
<td>&quot;show url-blacklisting database all&quot;</td>
</tr>
<tr>
<td>329</td>
<td>Disabled</td>
<td>&quot;show url-blacklisting database facility acsmgr all&quot;</td>
</tr>
<tr>
<td>330</td>
<td>Disabled</td>
<td>&quot;show active-charging tethering-detection database&quot;</td>
</tr>
<tr>
<td>331</td>
<td>Disabled</td>
<td>&quot;show active-charging tethering-detection database sessmgr all&quot;</td>
</tr>
<tr>
<td>332</td>
<td>Disabled</td>
<td>&quot;show active-charging tethering-detection statistics&quot;</td>
</tr>
<tr>
<td>333</td>
<td>Disabled</td>
<td>&quot;show ims-authorization service statistics&quot;</td>
</tr>
<tr>
<td>334</td>
<td>Disabled</td>
<td>&quot;show ims-authorization policy-control statistics&quot;</td>
</tr>
<tr>
<td>335</td>
<td>Disabled</td>
<td>&quot;show ims-authorization policy-control statistics debug-info&quot;</td>
</tr>
<tr>
<td>336</td>
<td>Disabled</td>
<td>&quot;show local-policy statistics summary&quot;</td>
</tr>
<tr>
<td>337</td>
<td>Disabled</td>
<td>&quot;show rohe statistics&quot;</td>
</tr>
<tr>
<td>338</td>
<td>Disabled</td>
<td>&quot;show dns client statistics&quot;</td>
</tr>
<tr>
<td>339</td>
<td>Disabled</td>
<td>&quot;show hss-peer-service service all&quot;</td>
</tr>
<tr>
<td>340</td>
<td>Disabled</td>
<td>&quot;show ipms status all&quot;</td>
</tr>
<tr>
<td>341</td>
<td>Disabled</td>
<td>&quot;show ipms status debug-info&quot;</td>
</tr>
<tr>
<td>342</td>
<td>Disabled</td>
<td>&quot;show kvstore&quot;</td>
</tr>
<tr>
<td>343</td>
<td>Disabled</td>
<td>&quot;show kvstore verbose&quot;</td>
</tr>
<tr>
<td>344</td>
<td>Disabled</td>
<td>&quot;show kvstore kvclient&quot;</td>
</tr>
<tr>
<td>345</td>
<td>Disabled</td>
<td>&quot;show kvstore kvmgr&quot;</td>
</tr>
<tr>
<td>346</td>
<td>Disabled</td>
<td>&quot;show pcc-service all&quot;</td>
</tr>
<tr>
<td>347</td>
<td>Disabled</td>
<td>&quot;show pcc-service statistics all&quot;</td>
</tr>
<tr>
<td>348</td>
<td>Disabled</td>
<td>&quot;show pcc-policy service all&quot;</td>
</tr>
<tr>
<td>349</td>
<td>Disabled</td>
<td>&quot;show pcc-policy service statistics all&quot;</td>
</tr>
<tr>
<td>350</td>
<td>Disabled</td>
<td>&quot;show pcc-quota service all&quot;</td>
</tr>
<tr>
<td>351</td>
<td>Disabled</td>
<td>&quot;show pcc-quota service statistics all&quot;</td>
</tr>
<tr>
<td>No.</td>
<td>Default SDR</td>
<td>Command String</td>
</tr>
<tr>
<td>-----</td>
<td>-------------</td>
<td>----------------</td>
</tr>
<tr>
<td>352</td>
<td>Disabled</td>
<td>&quot;show pcc-af service all&quot;</td>
</tr>
<tr>
<td>353</td>
<td>Disabled</td>
<td>&quot;show pcc-af service statistics all&quot;</td>
</tr>
<tr>
<td>354</td>
<td>Disabled</td>
<td>&quot;show pcc-sp-endpoint all&quot;</td>
</tr>
<tr>
<td>355</td>
<td>Disabled</td>
<td>&quot;show pcc-sp-endpoint statistics all&quot;</td>
</tr>
<tr>
<td>356</td>
<td>Disabled</td>
<td>&quot;show event-notif server all&quot;</td>
</tr>
<tr>
<td>357</td>
<td>Disabled</td>
<td>&quot;show event-notif statistics&quot;</td>
</tr>
<tr>
<td>358</td>
<td>Disabled</td>
<td>&quot;show demux-mgr statistics bindmux all&quot;</td>
</tr>
<tr>
<td>359</td>
<td>Disabled</td>
<td>&quot;show congestion-control configuration&quot;</td>
</tr>
<tr>
<td>360</td>
<td>Disabled</td>
<td>&quot;show congestion-control statistics mme full&quot;</td>
</tr>
<tr>
<td>361</td>
<td>Disabled</td>
<td>&quot;show congestion-control statistics imsimgr all full&quot;</td>
</tr>
<tr>
<td>362</td>
<td>Enabled</td>
<td>&quot;show ge-switch counters (second sample)&quot;</td>
</tr>
<tr>
<td>363</td>
<td>Enabled</td>
<td>&quot;ethtool -S cpeth&quot;</td>
</tr>
<tr>
<td>364</td>
<td>Enabled</td>
<td>&quot;Standby SMC Ophir Mac counters (second sample)&quot; [ASR 5000 only]</td>
</tr>
<tr>
<td>365</td>
<td>Disabled</td>
<td>&quot;show cli history&quot;</td>
</tr>
<tr>
<td>366</td>
<td>Disabled</td>
<td>&quot;card-cpu boxer summary&quot;</td>
</tr>
<tr>
<td>367</td>
<td>Disabled</td>
<td>&quot;show sls-service all&quot;</td>
</tr>
<tr>
<td>368</td>
<td>Disabled</td>
<td>&quot;show sls-service peers all&quot;</td>
</tr>
<tr>
<td>369</td>
<td>Disabled</td>
<td>&quot;show sls-service statistics all&quot;</td>
</tr>
</tbody>
</table>

Notes:
- Enabled = Included in default record section
- Disabled = Not included in default record section