Important Information

Latest Software
We recommend that you install the most recent software release to stay up-to-date with the latest functional improvements, stability fixes, security enhancements and protection against new and evolving attacks.

Latest Documentation
The latest version of this document is at: (http://supportcontent.checkpoint.com/documentation_download?ID=27964)
To learn more, visit the Check Point Support Center (http://supportcenter.checkpoint.com).
For more about this release, see the R76SP home page (http://supportcontent.checkpoint.com/solutions?id=sk94686).

Revision History

<table>
<thead>
<tr>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 May 2014</td>
<td>General updates and corrections.</td>
</tr>
</tbody>
</table>
|                    | Corrected file path in the Configuring SGMs ("Configuring SGMs
|                    | (asg_blade_config)" on page 101) - Troubleshooting section and updated the
|                    | asg_blade_config CLI syntax. |
|                    | Corrected UIDs in Monitoring the System with SNMP (on page 83). |
| 20 February 2014   | First release of this document |

Feedback
Check Point is engaged in a continuous effort to improve its documentation.
Please help us by sending your comments (mailto:cp_techpub_feedback@checkpoint.com?subject=Feedback on 61000/41000 Security System R76SP Administration Guide).
## Important Information

Working with System Status

- Networking Monitoring
  - Monitoring Service Traffic (asg profile)
  - Monitoring the 61000/41000 Security System (asg_archive)
  - Working with Interface Status (asg if)
  - Showing Bond Interfaces (asg_bond)
  - Showing Traffic Information (asg_ifconfig)
  - Working with Routing Tables (asg_route)
  - Showing Multicast Information
  - VPN Packet Tracking (bcstats)
  - Showing SSM Traffic Statistics (asg_traffic_stats)
  - Showing SGM Forwarding Statistics (asg_blade_stats)
  - Multi-blade capture (tcpdump -mcap -view)
  - Traceroute (asg_traceret)

- Hardware Monitoring and Control
  - Showing Chassis and Component State (asg stat)
  - Monitoring Chassis and Component Status (asg_monitor)
  - Monitoring Performance (asg perf)
  - Monitoring SGM Resources (asg_resource)
  - Searching for a Connection (asg_search)
  - Configuring Alerts for SGM and Chassis Events (asg_alert)
  - Collecting System Diagnostics (asg_diag)
  - Monitoring Hardware Components (asg_hw_monitor)
  - Chassis Control (asg_chassis_ctrl)

- Security Monitoring
  - SYN Defender (sim synatk, sim6 synatk, asg synatk)
  - F2F Quota (asg f2lq, fwacct f2lq stats)
  - Showing the Number of Firewall and SecureXL Connections (asg_conns)
  - Packet drop monitoring (HLINK_1)

- Other Monitoring Commands
  - Showing System Serial Numbers
  - Showing the 61000/41000 Security System Version (ver)
  - Looking a Log Files (asg log)
  - Looking at the Auditlog File (asg_auditlog)
  - Working with the firewall Database Configuration (asg_config)
  - Showing Software and Firmware versions (asg_version)
  - Showing System Messages (asg_varlog)
  - Monitoring the System with SNMP
  - Monitoring Virtual Systems (cpha_vsx_util_monitor)

- System Configuration

- Administration
- Working with Global Commands
- Check Point global commands
- Global Operating System Commands
- Global Commands Generated by CMM
- General global commands
- Synchronize SGM Time (asg_ntp_sync_config)
- Configuring SGMs (asg_blade_config)
- Backing Up and Restoring an SGM (backup_system)
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restore Procedures</td>
<td>103</td>
</tr>
<tr>
<td>Configuring SGM state (asg_sgm_admin)</td>
<td>104</td>
</tr>
<tr>
<td>Image Management</td>
<td>105</td>
</tr>
<tr>
<td>Global Image Management - (snapshot)</td>
<td>105</td>
</tr>
<tr>
<td>Image Management for Specified SGMs (g_snapshot)</td>
<td>106</td>
</tr>
<tr>
<td>High Availability</td>
<td>107</td>
</tr>
<tr>
<td>Chassis High Availability Active/Standby Mode</td>
<td>107</td>
</tr>
<tr>
<td>Setting Chassis Weights (chassis high-availability factors)</td>
<td>109</td>
</tr>
<tr>
<td>Chassis High Availability Active/Active Mode</td>
<td>110</td>
</tr>
<tr>
<td>Changing the High Availability Mode</td>
<td>110</td>
</tr>
<tr>
<td>Admin Down on First Join (down_on_first_join)</td>
<td>111</td>
</tr>
<tr>
<td>Chassis ID Configuration</td>
<td>111</td>
</tr>
<tr>
<td>Configuring a Unique IP address per Chassis (UIPC)</td>
<td>112</td>
</tr>
<tr>
<td>asg_sync_manager</td>
<td>113</td>
</tr>
<tr>
<td>Verifying the High Availability Configuration</td>
<td>115</td>
</tr>
<tr>
<td>Monitoring, Logs and Auditing</td>
<td>115</td>
</tr>
<tr>
<td>Redirecting Alerts and Logs to External syslog server (asg_syslog)</td>
<td>115</td>
</tr>
<tr>
<td>Monitoring Management Interfaces Link State</td>
<td>118</td>
</tr>
<tr>
<td>Log Server Distribution (asg_log_servers)</td>
<td>120</td>
</tr>
<tr>
<td>Configuring a Dedicated Logging Port</td>
<td>121</td>
</tr>
<tr>
<td>Command Auditing</td>
<td>122</td>
</tr>
<tr>
<td>Port Mirroring (SPAN Port)</td>
<td>123</td>
</tr>
<tr>
<td>Configuring Port Mirroring on a Security Gateway</td>
<td>123</td>
</tr>
<tr>
<td>Configuring Port Mirroring for a VSX Gateway</td>
<td>124</td>
</tr>
<tr>
<td>Security</td>
<td>126</td>
</tr>
<tr>
<td>Generic Routing Encapsulation – GRE (asg_gre)</td>
<td>126</td>
</tr>
<tr>
<td>Role Based Administration (RBA)</td>
<td>127</td>
</tr>
<tr>
<td>RADIUS Authentication</td>
<td>128</td>
</tr>
<tr>
<td>VSX Provisioning</td>
<td>130</td>
</tr>
<tr>
<td>Clean Installation</td>
<td>130</td>
</tr>
<tr>
<td>Reconfigure (vsx_util reconfigure)</td>
<td>133</td>
</tr>
<tr>
<td><strong>Network Management</strong></td>
<td>133</td>
</tr>
<tr>
<td>Working with IPv6</td>
<td>133</td>
</tr>
<tr>
<td>Enabling/Disabling IPv6 Support (ipv6-state)</td>
<td>133</td>
</tr>
<tr>
<td>Configuring the 6in4 Internet Transition Mechanism</td>
<td>137</td>
</tr>
<tr>
<td>Working with the Bridge Mode</td>
<td>138</td>
</tr>
<tr>
<td>Configuring Bridge Interfaces</td>
<td>139</td>
</tr>
<tr>
<td>Disabling BPDU Forwarding</td>
<td>139</td>
</tr>
<tr>
<td>Configuring Link Aggregation (Bonding)</td>
<td>140</td>
</tr>
<tr>
<td>Creating a Bonding Group</td>
<td>140</td>
</tr>
<tr>
<td>Setting a Bonding Mode</td>
<td>141</td>
</tr>
<tr>
<td>Setting a Polling interval</td>
<td>142</td>
</tr>
<tr>
<td>Setting the Slave Interface to On</td>
<td>142</td>
</tr>
<tr>
<td>Enslaving Interfaces</td>
<td>142</td>
</tr>
<tr>
<td>Removing Slaves from a Bond</td>
<td>142</td>
</tr>
<tr>
<td>Deleting a Bonding Group</td>
<td>142</td>
</tr>
<tr>
<td>Configuring VLANs</td>
<td>143</td>
</tr>
<tr>
<td>Configuring Dynamic Routing - Unicast</td>
<td>144</td>
</tr>
<tr>
<td>Configuring OSPF on an Interface</td>
<td>144</td>
</tr>
<tr>
<td>Configuring BGP</td>
<td>144</td>
</tr>
<tr>
<td>Changing the Default VMAC (asg_unique_mac_utility)</td>
<td>145</td>
</tr>
<tr>
<td>Verifying the New MAC Address</td>
<td>147</td>
</tr>
<tr>
<td>Changing the Management Interface</td>
<td>147</td>
</tr>
<tr>
<td>Configuring Policy Based Routing</td>
<td>148</td>
</tr>
<tr>
<td>ECMP Configuration</td>
<td>148</td>
</tr>
<tr>
<td>Enhanced Failover of ECMP Static Routes</td>
<td>149</td>
</tr>
<tr>
<td>Working with the ARP Table (asg_arp)</td>
<td>151</td>
</tr>
<tr>
<td>Verbose Mode Output</td>
<td>153</td>
</tr>
<tr>
<td>Verifying MAC Addresses</td>
<td>153</td>
</tr>
</tbody>
</table>
Security Group (asg security_group) .......................................................... 204
Working with the Distribution Mode ......................................................... 204
Automatic Distribution Configuration (Auto-Topology) .......................... 205
Setting and Showing the Distribution Configuration ............................ 207
Configuring the Interface Distribution Mode (set distribution interface) .. 208
Showing Distribution Status .................................................................. 209
Running a Verification Test (show distribution verification) ................. 210
NAT and the Correction Layer on Security Gateway .............................. 210
NAT and the Correction Layer on a VSX Gateway ................................. 210
Hybrid System ...................................................................................... 213
GARP Chunk Mechanism .................................................................... 214

Hardware Components ........................................................................... 215
Chassis Management Module (CMM) CLI ............................................. 215
Security Switch Module (SSM) CLI ......................................................... 217
SSM60 CLI ......................................................................................... 217
SSM160 CLI ....................................................................................... 218
Security Gateway Modules ................................................................... 222
Identifying SGMs in the Chassis (asg_detection) ................................. 222
SGM260 LEDs ................................................................................... 222
SGM220 LEDs .................................................................................. 224
Security Switch Module LEDs .............................................................. 225

Software Blades Support ....................................................................... 226
Software Blades Updates ....................................................................... 226
IPS Bypass under Load ........................................................................ 226
IPS Cluster Failover Management ......................................................... 227

Troubleshooting .................................................................................... 228
Collecting System Information (asg_info) ........................................... 228
Verifiers ............................................................................................... 232
MAC Verification (mac_verifier) ........................................................... 232
L2 Bridge Verifier (asg_br_verifier) ....................................................... 232
Port Connectivity Verification (asg_pingable_hosts) ......................... 234
Verifying VSX Gateway Configuration (asg vsx_verify) ................... 236
Reseting SIC (g_cpconfig sic init) ........................................................... 239
Reseting SIC on a Security Gateway or VSX Gateway (VS0) ............... 239
Reset SIC for non-VS0 Virtual Systems ................................................ 239
Troubleshooting SIC reset .................................................................. 240

Troubleshooting Hardware ................................................................... 240
Security Gateway Module (SGM) .......................................................... 240
Chassis Management Module (CMM) ................................................... 242
Security Switch Module (SSM) .............................................................. 243
Fans .................................................................................................... 244
Power Supply Unit (PSU) ..................................................................... 244
Debug files ........................................................................................... 245
Terms

**Active/Standby**
A High Availability cluster where only one member handles connections.

**Administrator**
A SmartDashboard or SmartDomain Manager user with permissions to manage Check Point security products and the network environment.

**Affinity**
The assignment of a specified process, Firewall instance, VSX Virtual System, interface or IRQ with one or more CPU cores.

**Bond**
A virtual interface that contains two or more physical interfaces for redundancy and load sharing.

**BPDU**
Bridge Protocol Data Unit. Data messages that are sent between switches in an extended LAN that uses a Spanning Tree Protocol (STP) topology.

**Bridge Mode**
A Security Gateway or Virtual System that works as layer-2 bridge device for easy deployment in an existing topology.

**CCP**
Cluster Control Protocol. Proprietary Check Point protocol that manages synchronization between High Availability between cluster members.

**Chassis**
The container that contains all the components of a 61000/41000 Security System.

**Cluster**
Two or more Security Gateways connected to each other for High Availability and/or Load Sharing.

**Cluster Member**
A Security Gateway that is part of a cluster.

**ClusterXL**
Check Point software-based cluster solution for Security Gateway redundancy and Load Sharing.

**CMM**
Chassis Management Module. Hardware component that controls and monitors Chassis operation. This includes fan speed, Chassis and module temperature, and component hot-swapping.

**CoreXL**

**Failover**
A redundancy operation, where one cluster member automatically takes over for a failed member.

**Firewall**
The software and hardware that protects a computer network by analyzing the incoming and outgoing network traffic (packets).

**Firewall Instance**
On a Security Gateway with CoreXL enabled, the Firewall kernel is replicated multiple times. Each replicated copy, or firewall instance, runs on one processing core. These instances handle traffic concurrently, and each instance is a complete and independent inspection kernel.

**GARP**
Gratuitous Address Resolution Protocol. An ARP request or reply that is not normally required by the ARP specification (RFC 826).

**Hybrid System**
A 61000/41000 Security System that includes SGMs that have different quantities of CPU cores and configured CoreXL instances.

**Link Aggregation**
A technology that joins multiple physical interfaces together into one virtual interface, known as a bond interface. Also known as interface bonding.

**Management Server**
A Security Management Server or a Multi-Domain Security Management Multi-Domain Server that manages one or more Security Gateways and security policies.

**Multi Domain Log Server**
Physical server that contains the log database for all Domains.
Multi-Domain Security Management
A centralized management solution for large-scale, distributed environments with many different network Domain Management Servers.

Multi-Domain Server
A physical server that contains system information and policy databases for all Domains in an enterprise environment.

Packet
A formatted unit of data that moves on computer networks.

PEM
Power Entry Module. Hardware component that supplies DC power to the Chassis with EMC filtering and over-current protection.

Permissions Profile
A predefined group of SmartConsole access permissions assigned to Domains and administrators. This feature lets you configure complex permissions for many administrators with one definition.

Policy
A collection of rules that control network traffic and enforce organization guidelines for data protection and access to resources through the use of packet inspection.

Primary Multi-Domain Server
The first Multi-Domain Server that you define and log into in a High Availability deployment.

PSU
Power Supply Unit. Hardware component that supplies AC power to the chassis with filtering and over-current protection.

Secondary Multi-Domain Server
All Multi-Domain Servers in a High Availability deployment created after the Primary Multi-Domain Server.

Security Gateway
A computer or appliance that inspects traffic and enforces Security Policies for connected network resources.

Security Management Server
The application that manages, stores, and distributes the security policy to Security Gateways.

SGM

SIC
Secure Internal Communication. The process by which networking components authenticate over SSL between themselves and the Security Management Server, as the Internal Certificate Authority (ICA), for secure communication. The Security Management Server issues a certificate, which components use to validate the identity of others.

SmartDashboard
A Check Point client used to create and manage the security policy.

SmartUpdate
SmartConsole client used to centrally upgrade and manage Check Point software and licenses.

SMO
Single Management Object. A Check Point technology that manages the 61000/41000 Security System as one large Security Gateway with one management IP address. All management tasks, are handled by one SGM (the SMO Master), which updates all other SGMs. All management tasks, such as Security Gateway configuration, policy installation, remote connections and logging are handled by the SMO master.

SMO Master
The physical SGM that handles management tasks for all SGMs in a 61000/41000 Security System environment. By default, the SGM with the lowest ID number assigned this role.

SNMP
Simple Network Management Protocol. A protocol used to monitor the activity of hardware and software in a network.

SNMP Counter
An SNMP object with an integer value that increases by one when a specified event occurs. Counters are typically used as performance metrics, such as network throughput, dropped packets, or error events.

SNMP Trap
A notification of an event generated by an SNMP-enabled device and sent to the SNMP server.
**SSM**
Security Switch Module. Hardware component that manages the flow of network traffic to and from the Security Gateway Modules.

**Standby Domain Management Server**
All Domain Management Servers for a Domain that are not designated as the active Domain Management Server.

**Standby Multi-Domain Server**
All Multi-Domain Servers in a High Availability deployment that cannot manage global policies and objects. Standby Multi-Domain Servers are synchronized with the active Multi-Domain Server.

**Traffic**
The flow of data between network resources.

**Virtual Device**
A logical object that emulates the functionality of a type of physical network object.

**Virtual Router**
A virtual device that functions as a physical router.

**Virtual Switch**
Also vSwitch. A software abstraction of a physical Ethernet switch that can connect to physical switches through physical network adapters, to join virtual networks with physical networks.

**Virtual System**
A virtual device that implements the functionality of a Security Gateway.

**VLAN**
Virtual Local Area Network. Open servers or appliances connected to a virtual network, which are not physically connected to the same network.

**VLAN Trunk**
A connection between two switches that contains multiple VLANs.

**VPN**
Virtual Private Network. A secure, encrypted connection between networks and remote clients on a public infrastructure, to give authenticated remote users and sites secured access to an organization's network and resources.

**VSLS**
Virtual System Load Sharing. A VSX cluster technology that assigns Virtual System traffic to different active cluster members.

**VSX**
**Definition:** Virtual System Extension - Check Point virtual networking solution, hosted on a single computer or cluster containing virtual abstractions of Check Point Security Gateways and other network devices. These virtual devices provide the same functionality as their physical counterparts.

**VSX Gateway**
Physical server that hosts VSX virtual networks, including all **virtual devices** that provide the functionality of physical network devices. It holds at least one Virtual System, which is called VS0.

**Warp Link**
An interface between a Virtual System and a Virtual Switch or Virtual Router that is created automatically in a VSX topology.
Chapter 1

Working with System Status

In This Section:
  Networking Monitoring ............................................................... 12
  Hardware Monitoring and Control.................................................. 38
  Security Monitoring .................................................................... 68
  Other Monitoring Commands.......................................................... 76

Networking Monitoring

Monitoring Service Traffic (asg profile)

Use the asg profile command to monitor traffic per service that passes through the 61000/41000 Security System. This information is equivalent to SmartView Monitor traffic monitoring. This command has minimal performance impact.

Syntax

```
asg profile [ --delay <timeout> ] [ -b <sgm_ids> ] [ -v | -p | -g ]
[ --rel ] [ --tcp | --udp ] [ --ipv6 | --ipv4 ]
asg profile -m
asg profile --enable
asg profile --disable
asg profile --help
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--delay &lt;timeout&gt;</td>
<td>Information refresh interval (seconds).</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>- No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>- One SGM</td>
</tr>
<tr>
<td></td>
<td>- A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>- A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>- One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>- The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-v</td>
<td>-p</td>
</tr>
<tr>
<td></td>
<td>-v - Show verbose service statistics.</td>
</tr>
<tr>
<td></td>
<td>-p - Show service statistics for these paths:</td>
</tr>
<tr>
<td></td>
<td>- Acceleration (Accelerated by a SecureXL device)</td>
</tr>
<tr>
<td></td>
<td>- Medium</td>
</tr>
<tr>
<td></td>
<td>- Firewall</td>
</tr>
<tr>
<td></td>
<td>-g - Show graph view of BPS per service</td>
</tr>
<tr>
<td>--rel</td>
<td>Show the results as a percentage. For the -v -p and default view.</td>
</tr>
</tbody>
</table>
**Example**

> asg profile -m

Aggregated statistics of SGMs: 1_1 Virtual Systems: 0

<table>
<thead>
<tr>
<th>Service</th>
<th>Throughput</th>
<th>Packet rate</th>
<th>Connection rate</th>
<th>Concurrent connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>8116/udp cp-cluster</td>
<td>116.2 K</td>
<td>112</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22/tcp ssh</td>
<td>4.5 K</td>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33628/tcp</td>
<td>2.0 K</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33630/tcp</td>
<td>1.2 K</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33632/tcp</td>
<td>1.2 K</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33633/tcp</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>33636/tcp</td>
<td>400</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>67/udp bootps</td>
<td>288</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>257/tcp set</td>
<td>48</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total tcp</td>
</tr>
<tr>
<td>Total udp</td>
</tr>
<tr>
<td>Total other</td>
</tr>
<tr>
<td>Total</td>
</tr>
</tbody>
</table>

Time: Sun Jul 07 14:34:30 IDT 2013
SGMs: 1_1 1_2
VSs: 0 1
Choose one of the following option: (Bold options are current view)
- `n` Normal View
  - `a` Absolute Values
  - `r` Relative Values
- `v` Verbose View
- `V` Move to a different Virtual System Shay said he will add this
- `p` Path View
- `g` Graph View
- `O` Online
- `H` History
- `S` Move to next sgm
- `b` Back one menu
- `e` Exit

Choose one of these options:

- `--tcp` Show TCP statistics only
- `--udp` Show UDP statistics only

Choose one of these options:

- `--ipv4` Show ipv4 statistics only.
- `--ipv6` Show ipv6 statistics only.

Run in a convenient interactive menu mode.
Enable statistics collection.
Disable statistics collection.
Show command syntax and help information.
Notes
This example shows the normal (not verbose) view with absolute values. The highest throughput and packet rate is from the service 8116/udp cp-cluster. To show this view, type a.

Monitoring the 61000/41000 Security System (asg_archive)

The asg_archive utility collects 61000/41000 Security System status and activity information in real time, which is periodically saved to a history file. The system refreshes the data and saves history files automatically based on predefined time intervals for each status information type. You can change the refresh time intervals based on your requirements.

The asg_archive utility shows current and historical statistics for each SGM or VSX Virtual System. You can easily change the SGM and/or Virtual System that shows. You can enable or disable data collection globally for all status types or for specified status types. You can also assign the data collection process to a specified CPU to help prevent negative performance impact.

Syntax

asg_archive
asg_archive --height <max_lines>
asg_archive --{enable|--disable}
asg_archive --status
asg_archive --config [collectors {enable|disable} [seconds]]
asg_archive --refresh [timeout]
asg_archive --cpu [cpu_id]
asg_archive --remote <path>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Parameter</td>
<td>Shows the System Status and the Options menu.</td>
</tr>
<tr>
<td>--height</td>
<td>Set the maximum number of lines in the output.</td>
</tr>
<tr>
<td>--enable</td>
<td>Start all data collectors, except those that were manually disabled with</td>
</tr>
<tr>
<td></td>
<td>asg_archive --config.</td>
</tr>
<tr>
<td>--disable</td>
<td>Disable all information collectors.</td>
</tr>
<tr>
<td>--status</td>
<td>Show if asg_archive is enabled or disabled.</td>
</tr>
<tr>
<td>--config</td>
<td>Show or set the configuration of information collectors:</td>
</tr>
<tr>
<td></td>
<td>collector - Name of the information collector, as shown in the asg_archive</td>
</tr>
<tr>
<td></td>
<td>--config output. Enclose the name in double quotes.</td>
</tr>
<tr>
<td></td>
<td>timeout - Enter a refresh period, in seconds, for the specified collector. If</td>
</tr>
<tr>
<td></td>
<td>you do not enter a refresh, the default value is applied automatically.</td>
</tr>
<tr>
<td>--refresh</td>
<td>Show or set the default refresh time, in seconds, which applies when no value</td>
</tr>
<tr>
<td></td>
<td>is specified with the --config parameter.</td>
</tr>
<tr>
<td>--cpu &lt;cpu_id&gt;</td>
<td>Show or select the default CPU assigned to the data collection process. This</td>
</tr>
<tr>
<td></td>
<td>can help prevent unnecessary performance impact caused by this command.</td>
</tr>
<tr>
<td>--remote &lt;path&gt;</td>
<td>Read archive files from a specified remote Security Gateway. Specify the path</td>
</tr>
<tr>
<td></td>
<td>to this Security Gateway.</td>
</tr>
<tr>
<td>--help</td>
<td>Show the command syntax and help text. This option automatically closes the</td>
</tr>
<tr>
<td></td>
<td>interactive mode and goes back to the command line.</td>
</tr>
</tbody>
</table>
Working with the Interactive Mode.

When you run `asg_archive`, the system enters the interactive mode and shows a menu. You select an option and the applicable status information shows on the upper portion of the screen. Some menu item have sub-menus with more choices. Use the arrow keys to scroll through the status information. The menu is always available on the lower portion of the screen. This example shows the memory status (option 3-m).

```
+-----------------------------------+----------+----------+----------+----------+
| Resource Table                    |          |          |          |
+-----------------------------------+----------+----------+----------+----------+
| SGM ID   | Resource Name | Usage  | Threshold| Total    |
+-----------------------------------+----------+----------+----------+----------+
| 1_01    | Memory        | 20%    | 50%      | 31.3G    |
|         | HD: /         | 22%    | 80%      | 19.4G    |
|         | HD: /var/log  | 1%     | 80%      | 58.1G    |
|         | HD: /boot     | 19%    | 80%      | 288.6M   |
+-----------------------------------+----------+----------+----------+----------+
```

Time: Tue Jan 14 12:13:30 IST 2014
SGMs: 1_1 1_2 1_3 1_4 1_5 2_1 2_2 2_3 2_4 2_5
VSs: 0 1 2

Choose one of the following option: (Bold options are current view)
1) System Status
2) Performance
3) Hardware & Resources
   m) Memory
   f) FW Memory Allocation
   c) CPU Usage
   t) Top Process
   h) Hardware
4) SXL Statistics
5) Diagnostic
6) Logs
7) SYN Attack
8) Network
O) Online
H) History
S) Move to next SGM
V) Move to next VS
b) Back one menu
e) Exit

To select a menu item, enter the number or letter to the left of the item. The letters are case sensitive. If there is a sub-menu, the first option automatically shows in the upper section of the screen. To select a different option, enter the applicable letter. Some options open another sub-menu.

The numbered options show status and system information. The letter options, at the bottom of the menu, are operations that control the information display.

<table>
<thead>
<tr>
<th>Menu Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td><strong>Online</strong> - Shows the current status for the selected item</td>
</tr>
<tr>
<td>H</td>
<td><strong>History</strong> - Shows status historical status information saved in the history files. Select the sub-menu item to show the specified history file.</td>
</tr>
<tr>
<td>S</td>
<td><strong>Move to next SGM</strong> - Use this option to show the SGMs in sequential order.</td>
</tr>
<tr>
<td>V</td>
<td><strong>Move to next Virtual System</strong> - Use this option to show the different Virtual Systems in sequential order.</td>
</tr>
<tr>
<td>b</td>
<td><strong>Back one menu</strong> - Go back to the main menu or a higher sub-menu.</td>
</tr>
<tr>
<td>e</td>
<td><strong>Exit</strong> - Close the interactive mode and go back to the command line.</td>
</tr>
</tbody>
</table>
Working with Interface Status (asg if)

Description

Use this command to show information for interfaces for the 61000/41000 Security System. The command output shows:

- IPv4, IPv6, and MAC address
- Interface type
- State
- Currently defined interface speed
- MTU
- Duplex status

You can also use this command to do these interface management tasks:

- Set the interface speed
- Enable or disable the interface

Syntax

```
# asg if -h
# asg if [-i <interface> [-v] [enable|disable] [set_speed {0|1000|10000}] [-ip]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show command syntax.</td>
</tr>
<tr>
<td>-i &lt;interface&gt;</td>
<td>Interface status for the specified interface or a comma-separated list of interfaces. If this parameter is not specified, the status for all interfaces shows.</td>
</tr>
<tr>
<td>-v</td>
<td>Verbose - Shows detailed output.</td>
</tr>
<tr>
<td>enable</td>
<td>disable</td>
</tr>
<tr>
<td>set_speed</td>
<td>Set interface port speed.</td>
</tr>
<tr>
<td>-ip</td>
<td>Interface IPv4 or IPv6 address.</td>
</tr>
</tbody>
</table>

Global view of all interfaces (asg if)

You use the asg if command to show the current status of all defined interfaces on the system.

```
> asg if
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
|Interface Data   |Interface Data   |Interface Data   |Interface Data   |Interface Data   |Interface Data   |Interface Data   |Interface Data   |
|Interface        |IPv4 Address     |Info             |State            |Speed            |MTU              |Duplex           |
|                 |MAC Address      |            |(ch1)            |         |          |        |
|bond1            |17.17.17.10      |Bond Master     |(down)           |NA       |NA        |NA      |
|                 |00:1c:7f:81:05:fe|                |slaves:          |         |          |        |
|                 |                  |                |eth1-05(down)    |         |          |        |
|eth1-05          |                  |Bond slave      |(down)           |10G      |1500      |Full   |
|                 |                  |                 |master:          |         |          |        |
|                 |                  |                 |bond1(down)      |         |          |        |
|eth2-05          |00:1c:7f:81:05:fe|Bond slave      |(down)           |10G      |1500      |Full   |
|                 |                  |                 |master:          |         |          |        |
|                 |                  |                 |bond1(down)      |         |          |        |
|bond1.201        |18.18.18.10      |Vlan            |(down)           |NA       |NA        |NA      |
|                 |00:1c:7f:81:05:fe|                |             |         |          |        |
|br0              |-                  |Bridge Mast     |(up)            |NA       |NA        |NA      |
```
To add a comment to an interface, run:

`set interface <if_name> comment <comment_text>`

**Verbose mode**

The verbose mode shows extended information, including information retrieved from the switch. You can use the verbose mode for one interface or a comma-separated list of interfaces. This operation can take a few seconds for each interface.

```
# asg if -i eth1-01 -v
```

Collecting information, may take few seconds

<table>
<thead>
<tr>
<th>Interfaces Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
</tr>
<tr>
<td>MAC Address</td>
</tr>
<tr>
<td>IPv6 Address (local)</td>
</tr>
<tr>
<td>eth2-Sync</td>
</tr>
<tr>
<td>Sync</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>eth1-Sync</td>
</tr>
<tr>
<td>ETH</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Notes

- This sample output shows:
  - This sync interface is a bond-Master
  - Interfaces are up or down
- To add a comment to an interface, run:
  > `set interface <if_name> comment <comment_text>`

**Verbose mode**

The verbose mode shows extended information, including information retrieved from the switch. You can use the verbose mode for one interface or a comma-separated list of interfaces. This operation can take a few seconds for each interface.

```
# asg if -i eth1-01 -v
```

Collecting information, may take few seconds

<table>
<thead>
<tr>
<th>Interfaces Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
</tr>
<tr>
<td>MAC Address</td>
</tr>
<tr>
<td>IPv6 Address (local)</td>
</tr>
<tr>
<td>eth2-Sync</td>
</tr>
<tr>
<td>Sync</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>eth1-Sync</td>
</tr>
<tr>
<td>ETH</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Notes

- This sample output shows:
  - This sync interface is a bond-Master
  - Interfaces are up or down
- To add a comment to an interface, run:
  > `set interface <if_name> comment <comment_text>`

**Verbose mode**

The verbose mode shows extended information, including information retrieved from the switch. You can use the verbose mode for one interface or a comma-separated list of interfaces. This operation can take a few seconds for each interface.

```
# asg if -i eth1-01 -v
```

Collecting information, may take few seconds

<table>
<thead>
<tr>
<th>Interfaces Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
</tr>
<tr>
<td>MAC Address</td>
</tr>
<tr>
<td>IPv6 Address (local)</td>
</tr>
<tr>
<td>eth2-Sync</td>
</tr>
<tr>
<td>Sync</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>eth1-Sync</td>
</tr>
<tr>
<td>ETH</td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

Notes

- This sample output shows:
  - This sync interface is a bond-Master
  - Interfaces are up or down
- To add a comment to an interface, run:
  > `set interface <if_name> comment <comment_text>`
Enable/Disable interface ports

You can use the asg if command to enable or disable interface ports for specified <SGMs>.

To disable an interface port, run:

```
# asg if -i eth1-01 disable
```

You are about to perform port state disable on eth1-01 on blades: all

Are you sure? (Y - yes, any other key - no) y

Port state disable on eth1-01 requires auditing
Enter your full name: y
Enter reason for port state disable on eth1-01 [Maintenance]: y
WARNING: Port state disable on eth1-01 on blades: all, User: y, Reason: y interface eth1-01 is disabled

To enable an interface port, run:

```
# asg if -i eth1-01 enable
```

You are about to perform port state enable on eth1-01 on blades: all

Are you sure? (Y - yes, any other key - no) y

Port state enable on eth1-01 requires auditing
Enter your full name: y
Enter reason for port state enable on eth1-01 [Maintenance]: y
WARNING: Port state enable on eth1-01 on blades: all, User: y, Reason: y interface eth1-01 is enabled

Connecting to a specific SGM (blade)

When you connect to the 61000/41000 Security System, you are actually connected to one of the SGMs. You can use the blade command to open a connection to a different Security Gateway Module. You must run blade in the Expert mode, which establishes a new SSH connection over the Sync interface.

**Syntax**

```
blade [<chassis_id>_]<sgm_id>
```

**Example**

```
# blade 1_03
```

**Output**

Moving to blade 1_3

**Notes**

- When you only enter the SGM ID, the default Chassis is assumed.
- To go back to the last SGM, enter `exit`.
- You can run more than one `blade` command to open many SSH sessions.

**Set Port Speed**

You can set the port speed for one interface port or a comma-separated list of ports.
# asg if -i eth1-01,eth2-01 set_speed 10000
You are about to perform port speed change to 10000 on eth1-01 eth2-01 on blades: all

Are you sure? (Y - yes, any other key - no) y

Port speed change to 10000 on eth1-01 eth2-01 requires auditing
Enter your full name: y
Enter reason for port speed change to 10000 on eth1-01 eth2-01 [Maintenance]: y
WARNING: Port speed change to 10000 on eth1-01 eth2-01 on blades: all, User: y, Reason: y
Interface eth1-01 speed was set to 10G
Interface eth2-01 speed was set to 10G

**Showing Bond Interfaces (asg_bond)**

The `asg_bond` command shows bond interfaces and runs LACP packet tests:

- MAC address consistency for each Chassis
- Slave state consistency for all SGMs
- Database consistency for all SGMs
- Make sure that the LACP aggregator ID between bond and slaves are compatible
- Verification of the LACP packet between neighbors and key comparison

You can run this command for specified bonds or for all bonds.

**Syntax**

`asg_bond [v] [-i <filter>] [-help | -h]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h or --help</td>
<td>Show command syntax.</td>
</tr>
<tr>
<td>-i &lt;filter&gt;</td>
<td>Enter a bond name or a string. The output shows all bonds that match the bond name or those names that contain the text string.</td>
</tr>
<tr>
<td>-v</td>
<td>Run LACP packet test for the specified interfaces.</td>
</tr>
</tbody>
</table>

**Global List of all Bonds**

You can use this command without parameters to show all currently defined bonds.

```
# asg_bond
```

```
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
| Name            | Address         | Mode            | Slaves          | Result          | Comments        |
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
| bond1           | (MAC) 00:1c:7f:81:02:fe | LACP 802.3ad    | eth1-02         | OK              |                 |
|                 | (IPv4) 13.13.1.10 | Load Sharing    | eth1-03         |                 |                 |
|                 |                 |                 | eth2-03         |                 |                 |
|                 |                 |                 | eth2-02         |                 |                 |
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
| bond3           | (MAC) 00:1c:7f:82:04:fe | XOR             | eth2-04         | OK              |                 |
|                 | (IPv4) 23.23.1.10 | Load Sharing    | eth1-04         |                 |                 |
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
| bond5           | (MAC) 00:1c:7f:81:07:fe | Round-Rubin     | eth1-07         | OK              |                 |
|                 | (IPv4) 33.33.1.10 | Load Sharing    | eth2-07         |                 |                 |
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
| bond7           | (MAC) 00:00:00:00:00:00:fe | Active-Backup | OK              |                 | - No slaves exist |
|                 | (IPv4)         | High Availability |                 |                 |                 |
+-----------------+-----------------+-----------------+-----------------+-----------------+-----------------+
```
Filtering a Bond Interface

This example shows the command output for the specified bond.

```
# asg_bond -i bond5
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Mode</th>
<th>Slaves</th>
<th>Result</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond5</td>
<td>(MAC) 00:1c:7f:81:07:fe</td>
<td>Round-Rubin</td>
<td>eth1-07</td>
<td>OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(IPv4) 33.33.1.10</td>
<td>Load Sharing</td>
<td>eth2-07</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes

You can also specify a substring that is part of a bond name to show all bonds that contain the substring.

Verification Test

This example shows the verification test results for all bonds, including one with an error.

```
> asg_bond -v
Listening for LACP packets [............................] [ OK ]
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Address</th>
<th>Mode</th>
<th>Slaves</th>
<th>Result</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond1</td>
<td>(MAC) 00:1c:7f:81:02:fe</td>
<td>LACP 802.3ad</td>
<td>eth1-02</td>
<td>Failed</td>
<td>eth1-02 missing LACP pkts</td>
</tr>
<tr>
<td></td>
<td>(IPv4) 13.13.1.10</td>
<td>Load Sharing</td>
<td>eth1-03</td>
<td></td>
<td>eth1-03 missing LACP pkts</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>eth2-03</td>
<td></td>
<td>eth2-03 missing LACP pkts</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>eth2-02</td>
<td></td>
<td>eth2-02 missing LACP pkts</td>
</tr>
<tr>
<td>bond3</td>
<td>(MAC) 00:1c:7f:82:04:fe</td>
<td>XOR</td>
<td>eth2-04</td>
<td>OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(IPv4) 23.23.1.10</td>
<td>Load Sharing</td>
<td>eth1-04</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bond5</td>
<td>(MAC) 00:1c:7f:81:07:fe</td>
<td>Round-Rubin</td>
<td>eth1-07</td>
<td>OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(IPv4) 33.33.1.10</td>
<td>Load Sharing</td>
<td>eth2-07</td>
<td></td>
<td></td>
</tr>
<tr>
<td>bond7</td>
<td>(MAC) 00:00:00:00:00:00:fe</td>
<td>Active-Backup</td>
<td></td>
<td>OK</td>
<td>- No slaves exist</td>
</tr>
<tr>
<td></td>
<td></td>
<td>High Availability</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes

- The comments column shows a description of problems detected by the verification tests.
- Bond7 shows an incomplete definition with no slaves configured.

Setting the Minimum Number of Slaves in a Bond

Bond interfaces can be monitored by `asg stat`. A Bond interface is considered down when the number of slaves in the bond that are **UP** is less than `min_slaves` value. You can change the `min_slaves` value in `gclish`.

Syntax

```
set chassis high-availability bond <bond port> min_slaves
```

Example

```
> set chassis high-availability bond bond1 min_slaves 2
```

Notes

- The default value for `min_slaves` is 1.
- The bond is considered Down if the number of slaves in **UP** state is below `min_slaves` value.

Showing Traffic Information (asg_ifconfig)

The `asg_ifconfig` command collects traffic statistics from all or a specified range of SGMs. The combined output shows the traffic distribution between SGMs and their interfaces (calculated during a certain period).
The `asg_ifconfig` command has three modes:

- **Native**
  Default setting. When the `analyze` or `banalyze` option is not specified the command behaves similar to the native Linux `ifconfig` command, except that the output shows statistics for all interfaces on all SGMs and shows statistics for interfaces on the local SGM.

- **Analyze**
  Shows accumulated traffic information and traffic distribution between SGMs.

- **Banalyze**
  Shows accumulated traffic information and traffic distribution between interfaces

**Note:**
- The `analyze` and `banalyze` parameters cannot be used together.
- If you run this command in a Virtual System context, you can only see the output that applies to that context.

**Syntax**

```
asg_ifconfig
asg_ifconfig [-b <sgm_ids>] [<interface>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>The name of the interface</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>One SGM</td>
</tr>
<tr>
<td></td>
<td>A comma-separated list of SGMs (1_1, 1_4)</td>
</tr>
<tr>
<td></td>
<td>A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-d delay</td>
<td>Delay, in seconds, between data samples (default = 5).</td>
</tr>
<tr>
<td>-v</td>
<td>Verbose mode: Shows traffic distribution between interfaces.</td>
</tr>
<tr>
<td>-a</td>
<td>Shows total traffic volume. By default (without -a), the average traffic volume per second shows.</td>
</tr>
<tr>
<td>-h</td>
<td>Shows help information and exit.</td>
</tr>
<tr>
<td>analyze</td>
<td>Shows accumulated traffic information. Use the -v, -a and -d &lt;delay&gt; parameters to show traffic distribution between interfaces.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>banalyze</td>
<td>Shows accumulated traffic information. Use the <code>-v</code>, <code>-a</code> and <code>-d &lt;delay&gt;</code> parameters to show traffic distribution between interfaces. You can use these parameters to sort the traffic distribution table: <code>-rp X packets</code>, <code>-rb X bytes</code>, <code>-rd X dropped packets</code>, <code>-tp X packets</code>, <code>-tb X bytes</code>, <code>-td X dropped packet</code> For example, if you sort with the <code>-rb</code> option, the higher values appear at the top of the RX bytes column in the traffic distribution table:</td>
</tr>
<tr>
<td>SGM ID</td>
<td>RX packets</td>
</tr>
<tr>
<td>1_03</td>
<td>70%</td>
</tr>
<tr>
<td>1_02</td>
<td>20%</td>
</tr>
<tr>
<td>1_01</td>
<td>10%</td>
</tr>
</tbody>
</table>

By default, the traffic distribution table is not sorted.

### Native Usage

This example shows the total traffic sent and received by eth2-01 for all SGMs on Chassis 1 (Active Chassis). By default, the average traffic volume per second shows.

> asg_ifconfig -b chassis1 eth2-01

**as1_02:**

eth2-01 Link encap:Ethernet  HWaddr 00:1C:7F:81:01:EA  
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1  
RX packets:94 errors:0 dropped:0 overruns:0 frame:0  
TX packets:63447 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:5305 (5.1 KiB)  TX bytes:5688078 (5.4 MiB)

**1_03:**

eth2-01 Link encap:Ethernet  HWaddr 00:1C:7F:81:01:EA  
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1  
RX packets:137 errors:0 dropped:0 overruns:0 frame:0  
TX packets:26336 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:7591 (7.4 KiB)  TX bytes:2355386 (2.2 MiB)

**1_04:**

eth2-01 Link encap:Ethernet  HWaddr 00:1C:7F:81:01:EA  
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1  
RX packets:124 errors:0 dropped:0 overruns:0 frame:0  
TX packets:3098 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:6897 (6.7 KiB)  TX bytes:378990 (370.1 KiB)

**1_05:**

eth2-01 Link encap:Ethernet  HWaddr 00:1C:7F:81:01:EA  
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1  
RX packets:79 errors:0 dropped:0 overruns:0 frame:0  
TX packets:26370 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:0  
RX bytes:4507 (4.4 KiB)  TX bytes:2216546 (2.1 MiB)
Using the Analyze Option

This example shows accumulated traffic volume statistics for eth2-Sync per SGM and the total for all SGMs. The traffic distribution for each SGM also shows. The -a option shows the total traffic volume instead of the average volume per second.

> asg_ifconfig eth2-Sync analyze -v -a
Command is executed on SGMs: chassis_active

1_01:
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:01:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:225018 bytes:36970520 (37.0 MiB) dropped:0
TX: packets:3522445 bytes:1381032583 (1.4 GiB) dropped:0

1_02:
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:02:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:221395 bytes:35947248 (35.9 MiB) dropped:0
TX: packets:4674143 bytes:1850315554 (1.9 GiB) dropped:0

1_03:
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:03:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:10 bytes:644 (644.0 b) dropped:0
TX: packets:67826313 bytes:7345458105 (7.3 GiB) dropped:0

1_04:
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:04:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:13 bytes:860 (860.0 b) dropped:0
TX: packets:7164109 bytes:2740761091 (2.7 GiB) dropped:0

1_05:
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:05:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:203386 bytes:19214238 (19.2 MiB) dropped:0
TX: packets:7164109 bytes:2740761091 (2.7 GiB) dropped:0

*= Accumulative *=
eth2-Sync   Link encap:Ethernet  HWaddr 00:1C:7F:01:04:FE
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
RX: packets:649822 bytes:92133510 (92.1 MiB) dropped:0
TX: packets:151676227 bytes:20805043393 (20.8 GiB) dropped:0

*= Traffic Distribution *=

<table>
<thead>
<tr>
<th>SGM ID</th>
<th>RX packets</th>
<th>RX bytes</th>
<th>RX dropped</th>
<th>TX packets</th>
<th>TX bytes</th>
<th>TX dropped</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_01</td>
<td>34.6%</td>
<td>40.1%</td>
<td>0.0%</td>
<td>2.3%</td>
<td>6.6%</td>
<td>0.0%</td>
</tr>
<tr>
<td>1_02</td>
<td>34.1%</td>
<td>39.0%</td>
<td>0.0%</td>
<td>3.1%</td>
<td>8.9%</td>
<td>0.0%</td>
</tr>
<tr>
<td>1_03</td>
<td>0.0%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>44.7%</td>
<td>35.3%</td>
<td>0.0%</td>
</tr>
<tr>
<td>1_04</td>
<td>0.0%</td>
<td>0.0%</td>
<td>0.0%</td>
<td>45.2%</td>
<td>36.0%</td>
<td>0.0%</td>
</tr>
<tr>
<td>1_05</td>
<td>31.3%</td>
<td>20.9%</td>
<td>0.0%</td>
<td>4.7%</td>
<td>13.2%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>
**Working with Routing Tables (asg_route)**

**Description**

`asg_route` is an advanced utility that collects and shows routing information on all SGMs. It also makes sure that route information in the 61000/41000 Security System database is the same as the operating system routing table. This can cause routing errors if not corrected. The command also makes sure that routing information is consistent between SGMs.

This command lets you filter and customize the collected information based on different criteria, such as:

- Specified SGMs or Chassis
- Virtual Systems
- IPv4 and IPv6 addresses
- Dynamic routing protocols
- Static routes
- Source-based routes
- Inactive routes

You can run a summary report that shows the number of routes in different categories and protocols. The summary report also makes sure that the routing information is the same for all SGMs.

**Basic Syntax**

```
asg_route -h
asg_route -v
asg_route [-a] [-b blade_string] [ipv6] [--vs <vs_ids>] [<inactive>] [<filter>]
asg_route [-a] [-b blade_string] [ipv6] [--vs <vs_ids>] comp_os_db
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show command syntax, help information and examples.</td>
</tr>
<tr>
<td>-v</td>
<td>Collect route information from all SGMs and save to a file at: /var/log/asg_route/all_routes</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Show only routes for the specified SGMs.</td>
</tr>
<tr>
<td>-ipv6</td>
<td>Show IPv6 routes only (default shows IPv4 routes only).</td>
</tr>
<tr>
<td>-a</td>
<td>Show all SGMs, including those that are in the admin down state.</td>
</tr>
<tr>
<td>--vs &lt;vs_ids&gt;</td>
<td>Show the routing table only for the specified Virtual System. This option is available only for VSX environments.</td>
</tr>
<tr>
<td>&lt;inactive&gt;</td>
<td>Optional inactive route filter parameters.</td>
</tr>
<tr>
<td>&lt;filter&gt;</td>
<td>Optional advanced routing parameters.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| --compare-os-db | Compares the routing data in the database with the operating system and shows:  
  - All routes in the database that are in the operating system routing table  
  - All routes in the operating system routing table that are not in the database  
  The <vs_ids> can be:  
  - No <vs_ids> (default) - Shows the current Virtual System context.  
  - One Virtual System.  
  - A comma-separated list of Virtual Systems (1,2,4,5).  
  - A range of Virtual Systems (VS 3-5).  
  - all - Shows all Virtual Systems.  
  Note: This parameter is only relevant in a VSX environment.                                                                                           |

**Note:**

You can combine many basic options on one line, but you can only use one `advanced_filter` option.

**Using an SGM Filter**

This example shows a simple filter for one SGM. The route type is shown as a one letter code in the left-hand column. The route type codes show at the end of the list.

```plaintext
> asg_route -b 1_01
Collecting routing information, may take few seconds...
===============================================
```

Fetching Routes info from SGMs:
1_01

Routes:
- C 127.0.0.0/8  is directly connected, lo
- C 130.0.0.0/24  is directly connected, eth1-CIN
- C 172.23.9.0/24  is directly connected, eth1-Mgmt4
- C 192.0.2.0/24  is directly connected, Sync
- S 0.0.0.0/0  via 172.23.9.4, eth1-Mgmt4, cost 0

Types:  
- C - Connected,  
- S - Static,  
- R - RIP,  
- B - BGP,  
- O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)  
- A - Aggregate,  
- K - Kernel Remnant,  
- H - Hidden,  
- P - Suppressed  
- SBR - Source-Based Routes

**Example 2**

The next example shows a complex SGM filter that includes 4 SGMs. Note that the results show route inconsistencies between the 61000/41000 Security System database and the operating system.

```plaintext
> asg_route -b 1_1,2_1 2_3
Collecting routing information, may take few seconds...
==============================================
```

Fetching Routes info from SGMs:
1_01,2_01,2_02,2_03

---

Status:  
- DB Routes info is NOT identical on all SGMs  
- OS Routes info is NOT identical on all SGMs
---

Identical DB Routes: (21 records)
- C 10.33.86.0/24  is directly connected, bond2.160
- C 10.33.87.0/24  is directly connected, bond2.163
- C 10.33.89.0/24  is directly connected, bond2.165
- C 127.0.0.0/8  is directly connected, lo
C 192.0.2.0/24 is directly connected, Sync
C 192.168.15.128/25 is directly connected, eth1-Mgmt4
C 192.168.33.0/24 is directly connected, bond1.33
C 192.168.34.0/24 is directly connected, bond1.34
C 198.51.100.0/25 is directly connected, eth1-CIN
C 198.51.100.128/25 is directly connected, eth2-CIN
C 2.2.2.0/24 is directly connected, bond2.166
S 0.0.0.0/0 via 192.168.33.1, bond1.33, cost 0
S 16.0.0.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.1.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.2.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.3.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.4.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.5.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.6.0/24 via 10.33.86.16, bond2.160, cost 0
S 16.0.8.0/24 via 10.33.86.16, bond2.160, cost 0
S 194.29.40.138/32 via 192.168.15.254, eth1-Mgmt4, cost 0

Inconsistent DB Routes:
1_01:

2_01:
R 10.33.96.0/24 via 192.168.33.96, bond1.33, cost 2, tag 13142
R 15.0.2.0/24 via 192.168.33.96, bond1.33, cost 2, tag 13142

2_02:

2_03:
R 10.33.96.0/24 via 192.168.33.96, bond1.33, cost 2, tag 13142
R 15.0.2.0/24 via 192.168.33.96, bond1.33, cost 2, tag 13142

Types: C - Connected, S - Static, R - RIP, B - BGP,
O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)
A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed
SBR - Source-Based Routes

Using the Summary Option (--summary)
The --summary parameter shows this summary information:

• Total number of routes by route type
• Summary of routes that are the same on the database and the operating system routing table
• Summary of routes where the database and operating system are different
• OSPF interfaces and neighbors
• BGP peers
Example:

```bash
> asg_route --summary
Collecting routing information, may take few seconds...

OSPF interfaces -
--* 6 blades: 1_02 1_03 1_04 2_01 2_02 2_03 --*

<table>
<thead>
<tr>
<th>Name</th>
<th>IP Address</th>
<th>Area ID</th>
<th>State</th>
<th>DR Interface</th>
<th>BDR Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond1.34</td>
<td>192.168.34.86</td>
<td>0.0.0.86</td>
<td>DR</td>
<td>192.168.34.86</td>
<td>0.0.0.0</td>
</tr>
<tr>
<td>bond2.163</td>
<td>10.33.87.1</td>
<td>0.0.0.91</td>
<td>BDR</td>
<td>10.33.87.88</td>
<td>10.33.87.1</td>
</tr>
</tbody>
</table>

Status: OK

OSPF neighbors -
--* 6 blades: 1_02 1_03 1_04 2_01 2_02 2_03 --*

<table>
<thead>
<tr>
<th>Neighbor</th>
<th>Pri</th>
<th>State</th>
<th>Address</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.33.87.88</td>
<td>1</td>
<td>FULL/DR</td>
<td>10.33.87.88</td>
<td>10.33.87.1</td>
</tr>
</tbody>
</table>

Status: OK

BGP peers -
--* 1 blade: 1_02 (DR Manager) --*

<table>
<thead>
<tr>
<th>PeerID</th>
<th>AS</th>
<th>State</th>
<th>ActRts</th>
<th>Routes</th>
<th>InUpds</th>
<th>OutUpds</th>
<th>Uptime</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.33.96</td>
<td>86</td>
<td>Active</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>00:00:00</td>
</tr>
</tbody>
</table>

--* 5 blades: 1_03 1_04 2_01 2_02 2_03 --*

<table>
<thead>
<tr>
<th>PeerID</th>
<th>AS</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.33.96</td>
<td>86</td>
<td>Idle</td>
</tr>
<tr>
<td>192.168.34.33</td>
<td>161</td>
<td>Idle</td>
</tr>
<tr>
<td>192.168.33.94</td>
<td>162</td>
<td>Idle</td>
</tr>
<tr>
<td>192.168.34.94</td>
<td>162</td>
<td>Idle</td>
</tr>
</tbody>
</table>

Status: OK

Fetching Summary info from SGMs:
1_02,1_03,1_04,2_01,2_02,2_03

Status: DB Summary info is NOT identical on all SGMs

OS Summary info is identical on all SGMs

Identical DB Summary: (7 records)

<table>
<thead>
<tr>
<th>Total</th>
<th>aggregate</th>
<th>connected</th>
<th>igrp</th>
<th>ospf</th>
<th>rip</th>
<th>static</th>
</tr>
</thead>
<tbody>
<tr>
<td>628</td>
<td>0</td>
<td>11</td>
<td>0</td>
<td>602</td>
<td>2</td>
<td>10</td>
</tr>
</tbody>
</table>

Inconsistent DB Summary:

<table>
<thead>
<tr>
<th>1_02:</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>4294967293</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1_03:</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1_04:</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2_01:</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td></td>
</tr>
</tbody>
</table>
Comparing the OS Routing Table with the Database (--compare-os-db)

You can use the --compare-os-db option to compare the routing data in the database with the operating system routing table. The output shows:

- All routes in the database that are in the operating system routing table
- All routes in the operating system routing table that are not in the database

Example:

```
# asg_route --compare-os-db
Collecting routing information, may take few seconds...
```

Fetching Routes info from SGMs:

```
1_01
>> Found inconsistency between routes in DB & OS
```

DB Routes that does not exists in OS: (7 records)
```
<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next Hop</th>
<th>Metric</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.33.92.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.145.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.146.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.147.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.148.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.149.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
<tr>
<td>12.1.150.0/24</td>
<td>10.33.87.88, bond2.163</td>
<td>2:0</td>
<td>bond2.163</td>
</tr>
</tbody>
</table>
```

OS Routes that does not exist in DB: (6 records)
```
<table>
<thead>
<tr>
<th>Prefix</th>
<th>Next Hop</th>
<th>Metric</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.9.9.9</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>gated</td>
</tr>
<tr>
<td>12.3.0.0/24</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>proto gated</td>
</tr>
<tr>
<td>12.3.1.0/24</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>proto gated</td>
</tr>
<tr>
<td>12.3.2.0/24</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>proto gated</td>
</tr>
<tr>
<td>12.3.3.0/24</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>proto gated</td>
</tr>
<tr>
<td>12.3.4.0/24</td>
<td>10.33.87.88, dev bond2.163</td>
<td>0:0</td>
<td>proto gated</td>
</tr>
</tbody>
</table>
```

Using the Advanced Filters

Advanced filters let you customize the routing table display to show only the routes that you want to see. This release includes these advanced filter criteria:

<table>
<thead>
<tr>
<th>Advanced Filter Criterion</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--route-filter</td>
<td>Shows active routes filtered by a specified parameter</td>
</tr>
<tr>
<td>--inactive-filter</td>
<td>Shows inactive routes filtered by a specified parameter</td>
</tr>
<tr>
<td>--dynamic-filter</td>
<td>Shows specified OSPF and BGP route information and makes sure that there are no inconsistencies between SGMs</td>
</tr>
</tbody>
</table>

Each advanced filter type has many different parameters that you can use to show a precisely filtered route list.
Advanced Filter Syntax and Parameters

You can combine many basic options on one line, but you can only use one advanced filter option at a time.

`asg_route [basic_options] -n |--dyn-route <parameter>`

### Dynamic Route Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ospf</code></td>
<td>Shows OSPF interfaces and neighbors</td>
</tr>
<tr>
<td><code>rip</code></td>
<td>Shows RIP interfaces and neighbors</td>
</tr>
<tr>
<td><code>bgp</code></td>
<td>Shows BGP peers</td>
</tr>
</tbody>
</table>

`asg_route [basic_options] -r | --route <parameter>`

### Advanced Filter Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aggregate</code></td>
<td>Shows active aggregate routes</td>
</tr>
<tr>
<td><code>bgp</code></td>
<td>Shows BGP peers</td>
</tr>
<tr>
<td><code>Destination &lt;address&gt;</code></td>
<td>Shows routes to the specified destination</td>
</tr>
<tr>
<td><code>direct</code></td>
<td>Shows directly connected routes</td>
</tr>
<tr>
<td><code>exact &lt;ip_address/mask&gt;</code></td>
<td>Shows a route from the specified IP address</td>
</tr>
<tr>
<td><code>subnets &lt;ip_address/mask&gt;</code></td>
<td>Shows routes to the specified network and subnets</td>
</tr>
<tr>
<td><code>ospf</code></td>
<td>Shows OSPF interfaces and neighbors</td>
</tr>
<tr>
<td><code>static</code></td>
<td>Shows static routes</td>
</tr>
<tr>
<td><code>rip</code></td>
<td>Shows RIP interfaces and neighbors</td>
</tr>
<tr>
<td><code>all</code></td>
<td>Shows all routes (Including inactive routes)</td>
</tr>
</tbody>
</table>

`asg_route [basic_options] -i | --inactive <parameter>`

### Inactive Route Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>aggregate</code></td>
<td>Shows active aggregate routes</td>
</tr>
<tr>
<td><code>bgp</code></td>
<td>Shows BGP routes</td>
</tr>
<tr>
<td><code>direct</code></td>
<td>Shows directly connected routes</td>
</tr>
<tr>
<td><code>ospf</code></td>
<td>Shows routes received from OSPF</td>
</tr>
<tr>
<td><code>static</code></td>
<td>Shows static routes</td>
</tr>
<tr>
<td><code>rip</code></td>
<td>Shows RIP Routes</td>
</tr>
<tr>
<td><code>all</code></td>
<td>Shows all routes (Including inactive routes)</td>
</tr>
</tbody>
</table>
**Advanced Filter Examples**

**Example 1 - BGP routes for all SGMs.**

```bash
> asg_route -b all --route-filter bgp
Collecting routing information, may take few seconds...

Fetches Routes info from SGMs:
1_01

Routes:
B 10.33.88.0/24  via 192.168.34.33, bond1.34, cost -1
B 10.33.94.0/24  via 192.168.33.94, bond1.33, cost -1
B 10.34.94.0/24  via 192.168.34.94, bond1.34, cost -1

Types: C - Connected, S - Static, R - RIP, B - BGP,
O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)
A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed
SBR - Source-Based Routes
```

**Example 2 - Dynamic Routing filter for OSPF neighbors**

```bash
> asg_route --dynamic-filter ospf_neighbors
Collecting routing information, may take few seconds...

OSPF neighbors - 

```
<table>
<thead>
<tr>
<th>Neighbor</th>
<th>Pri</th>
<th>State</th>
<th>Address</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.33.94.1</td>
<td>1</td>
<td>FULL/BDR</td>
<td>192.168.33.94</td>
<td>192.168.33.86</td>
</tr>
<tr>
<td>10.33.87.88</td>
<td>1</td>
<td>FULL/BDR</td>
<td>10.33.87.88</td>
<td>10.33.87.1</td>
</tr>
</tbody>
</table>
```

Status: OK

**Example 3 - Inactive OSPF Routes**

```bash
> asg_route --inactive-filter ospf
Collecting routing information, may take few seconds...

Fetches Routes info from SGMs:
1_01

Routes:
O H i 10.33.87.0/24  is an unusable route
O H i 192.168.33.0/24  is an unusable route
O H i 192.168.34.0/24  is an unusable route
O E i 194.29.40.138/32  via 10.33.87.88, bond2.163, cost 2:0

Types: C - Connected, S - Static, R - RIP, B - BGP,
O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)
A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed
SBR - Source-Based Routes
```

**Notes:**
- Do not use the \(-v\) argument with an advanced filter. If you use \(-v\), the command ignores the advanced filter and shows all routes.
Showing Multicast Information

Showing Multicast Routing - asg_mroute

Description
The asg_mroute command shows this multicast routing information in a tabular format:

- **Source** - Source IP address
- **Dest** - Destination address
- **Iif** - Source interface
- **Oif** - Outbound interface

You can filter the output for specified interfaces and SGMs.

Syntax
```
asg_mroute -h
asg_mroute [-d <destination_route>] [-s <source_route>] [-i <source_interface>] [-b <sgm_ids>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show command syntax.</td>
</tr>
<tr>
<td>-d</td>
<td>Destination multicast group IP address.</td>
</tr>
<tr>
<td>-s</td>
<td>Source IP address.</td>
</tr>
<tr>
<td>-i</td>
<td>Source interface name.</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;.</td>
</tr>
</tbody>
</table>

The <sgm_ids> can be:
- No <sgm_ids> specified or all shows all SGMs and Chassis
- One SGM
- A comma-separated list of SGMs (1_1,1_4)
- A range of SGMs (1_1-1_4)
- One Chassis (Chassis1 or Chassis2)
- The active Chassis (chassis_active)

Example: Show all multicast routes

This example shows all multicast routes for all interfaces and SGMs.

```
> asg_mroute
<table>
<thead>
<tr>
<th>Multicast Routing (All SGMs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>12.12.12.1</td>
</tr>
<tr>
<td>22.22.22.1</td>
</tr>
</tbody>
</table>
```

When no optional parameters are specified, all routes, interfaces and SGMs are shown.
Example: Show only specified interfaces or SGMs

This example shows routes for the specified source IP address, Interface and destination IP address.

```
> asg_mroute  -s 22.22.22.1 -i eth1-02  -d 225.0.90.91
+------------------------------------------------------------|
| Multicast Routing (All SGMs)                                |
+------------------------------------------------------------|
| Source | Dest   | Iif            | Oif            |
+--------+--------+----------------+----------------|
| 22.22.22.1 | 225.0.90.91 | eth1-02       | eth2-01        |
+------------------------------------------------------------|
```

Showing PIM Information - (asg_pim)

Description

The `asg_pim` command shows this PIM information in a tabular format:

- **Source** - Source IP address
- **Dest** - Destination IP address
- **Mode** - currently only dense mode is supported in GAUDI
- **Flags** - Local source and MFC state indicators
- **In. intf** - Source interface
- **RPF** - Reverse Path Forwarding indicator
- **Out int** - Outbound interface
- **State** - Outbound interface state

You can filter the output for specified interfaces and SGMs.

Syntax

```
asg_pim -h
asg_pim [-b <sgm_ids>] [-i <interface>] [-n <neighbor>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show command syntax.</td>
</tr>
</tbody>
</table>
| -b        | Works with SGMs and/or Chassis as specified by `<sgm_ids>`.
  The `<sgm_ids>` can be:
  - No `<sgm_ids>` specified or all shows all SGMs and Chassis
  - One SGM
  - A comma-separated list of SGMs (1_1,1_4)
  - A range of SGMs (1_1-1_4)
  - One Chassis (Chassis1 or Chassis2)
  - The active Chassis (chassis_active)
| -i        | Show only the specified source interface. |
| -n        | Show only the specified PIM neighbor. This parameter is relevant only with the `neighbors` option. |
| neighbors | Runs verification test to make sure that PIM neighbors are the same on all SGMs and shows this information:
  - **Verification** - Results of verification test.
  - **Neighbor** - PIM neighbor.
  - **Interface** - Interface name.
  - **Holdtime** - Time in seconds to hold a connection open during peer negotiation.
  - **Expires** - Minimum and Maximum expiration values for all SGMs. |
Example: Show PIM information for all interfaces and SGMs

This example shows PIM information and multicast routes for all interfaces and SGMs.

```plaintext
> asg_pim
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| PIM (All SGMs)                          |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| source  | dest    | Mode    | Flags   | In. intf  | RPF       | Out. intf | State   |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 12.12.1 | 225.0.90.90 | Dense-Mode | M | eth1-01 | none |             |         |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | M | eth1-02 | none | eth1-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | M | eth1-02 | none | eth1-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | L | eth1-02 | none | eth2-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | L | eth1-02 | none | eth2-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+

Flags: L - Local source, M - MFC State
```

- When no optional parameters are specified, all routes, interfaces and SGMs are shown.
- In this version, only the Dense Mode is supported.

Example: Show PIM information for all interfaces and SGMs

```plaintext
> asg_pim -i eth1-02 -b all
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| PIM (All SGMs)                          |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| SGM_01  |         |         |         |           |           |           |         |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| source  | dest    | Mode    | Flags   | In. intf  | RPF       | Out. intf | State   |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | M | eth1-02 | none | eth1-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | M | eth1-02 | none | eth1-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | L | eth1-02 | none | eth2-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+
| 22.22.22.1 | 225.0.90.90 | Dense-Mode | L | eth1-02 | none | eth2-01 | Forwarding |
+---------+---------+---------+---------+-----------+-----------+-----------+---------+

Example: Neighbors option

```plaintext
> asg_pim neighbors
+---------+---------+---------+---------+---------+---------+
| PIM Neighbors (All SGMs)                  |
+---------+---------+---------+---------+---------+---------+
| Verification:                              |
| Neighbors Verification: Passed - Neighbors are identical on all blades |
+---------+---------+---------+---------+---------+---------+
| bond1   | 105     | 11:36:45-11:37:59 |
+---------+---------+---------+---------+---------+---------+
```

Showing IGMP Information (asg_igmp)

Use this command to show IGMP information in a tabular format. You can filter the output for specified interfaces and SGMs. If no blade is specified, the command runs a verification to make sure that IGMP data is the same on all SGMs:

- **Group verification** - Makes sure that the groups exist on all SGMs. If a group is missing on some SGMs, a message shows which group is missing on which blade.
- **Global properties** - Makes sure that the flags, address and other information are identical on all SGMs.
- **Interfaces** - Makes sure that all blade have the same interfaces and that they are in the same state (Up or Down). If inconsistencies are detected, a warning message shows.
Syntax

```
asg_igmp -h
asg_igmp [-i <interface>] [-b <sgm_ids>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show command syntax.</td>
</tr>
<tr>
<td>-i</td>
<td>Source interface name.</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;.</td>
</tr>
<tr>
<td></td>
<td>The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>- No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>- One SGM</td>
</tr>
<tr>
<td></td>
<td>- A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>- A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>- One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>- The active Chassis (chassis_active)</td>
</tr>
</tbody>
</table>

Example: Show IGMP information for all interfaces and SGMs

This example shows IGMP information and multicast routes for all interfaces and SGMs. In this example, the verification detected an interface inconsistency.

```
> asg_igmp
Collecting IGMP information, may take few seconds...
+---------------------------------------------------------------+|
| IGMP (All SGMs)                                                ||
+---------------------------------------------------------------+|
| Interface: eth1-01                                            ||
| Verification:                                                 ||
| Group Verification: Passed - Information is identical on all blades||
| Global Properties Verification: Passed - Information is identical on all blades ||
+---------------------------------------------------------------+|
<table>
<thead>
<tr>
<th>Group</th>
<th>Age</th>
<th>Expire</th>
</tr>
</thead>
<tbody>
<tr>
<td>225.0.90.91</td>
<td>2m</td>
<td>4m</td>
</tr>
</tbody>
</table>
+---------------------------------------------------------------+|
| Flags | IGMP Ver | Query Interval | Query Response Interval | protocol | Advertise Address |
+---------------------------------------------------------------+|
| Querier | 2 | 125 | 10 | PIM | 12.12.12.10 |
+---------------------------------------------------------------+|
| Interface: eth1-02                                           ||
| Verification:                                                 ||
| Group Verification: Failed - Found inconsistency between blades ||
| - Group 225.0.90.92: missing in blades 1_02                   ||
| Global Properties Verification: Passed - Information is identical on all blades ||
+---------------------------------------------------------------+|
<table>
<thead>
<tr>
<th>Group</th>
<th>Age</th>
<th>Expire</th>
</tr>
</thead>
<tbody>
<tr>
<td>225.0.90.92</td>
<td>2m</td>
<td>3m</td>
</tr>
</tbody>
</table>
+---------------------------------------------------------------+|
| Flags | IGMP Ver | Query Interval | Query Response Interval | protocol | Advertise Address |
+---------------------------------------------------------------+|
| Querier | 2 | 125 | 10 | PIM | 22.22.22.10 |
+---------------------------------------------------------------+|

NOTE: Inconsistency found in interfaces configuration between blades
Inconsistent interfaces: eth1-02
Example: Show IGMP Information for a specified interface.

```bash
> asg_igmp -i bond1.3
Collecting IGMP information, may take few seconds...
+--------------------------------------------------------------------------------------------------+
<table>
<thead>
<tr>
<th>IGMP (All SGMs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface: bond1.3</td>
</tr>
</tbody>
</table>
+--------------------------------------------------------------------------------------------------+
|Verification                                                                                          |
+--------------------------------------------------------------------------------------------------+
|Group Verification: Passed - Information is identical on all blades                                  |
|Global Properties Verification: Passed - Information is identical on all blades                      |
+--------------------------------------------------------------------------------------------------+
|Group                                                           |Age|Expire |
+--------------------------------------------------------------------------------------------------+
|225.0.90.90                                                           |46m|3m     |
+--------------------------------------------------------------------------------------------------+
|Flags |IGMP Ver |Query Interval |Query Response Interval |protocol |Advertise Address |
+--------------------------------------------------------------------------------------------------+
|Querier | 2 |125          | 10                     |PIM      |12.12.12.11      |
```

**VPN Packet Tracking (bcstats)**

You can run these commands to monitor the IPSEC packet flow.

<table>
<thead>
<tr>
<th>To see:</th>
<th>Run:</th>
</tr>
</thead>
</table>
| Source and destination IP addresses | g_tcpdump for ip proto 50  
(For Site-to-Site VPN) |
| Which SGM encrypted packets are forwarded to | bcstats vpn -v |
| Which SGM holds the outbound SA | g_fw tab -t outbound_SPI -f  
Search for MSPI in the output. MSPI is the Meta SA, and shows which SGM holds the outbound SA. |

**Example - g_fw tab**

```
# fw tab -t outbound_SPI -f
using cptfmt
Formatting table’s data – this might take a while...
local host:  
Date: Nov 14, 2011
12:37:15 172.16.6.171 > : (+)====================================(÷); Table_Name: outbound_sPi; : (+); Attributes: dynamic, id 285, attributes: keep, sync, kbuf 6 7, expires 3600, limit 20400, hashsize 32768; product: VPN-1 & Firewall-1;  
12:37:15 1172.16.6.171 >1 : (+); peer: 172.16.6.189; ,sPi: fs9baec; CPTFMT_sep:   sPi: 1; Ic00MB1: c5364f5e6414aad9; ,cookieR: 95a478b10f9544a6; Expires: 3540/3610; product: VPN-1 & Firewall-1;  
```

The output can include Security Associations (SAs) with an MSPI of 0. These are dummy SAs and can safely be ignored.

**Showing SSM Traffic Statistics (asg_traffic_stats)**

Use this command to show traffic statistics, in terms of throughput (Bits per second) and Packet rate (packets per second), for SSM ports during a specified time period.

Packet rate statistics are divided to four categories:
- Unicast
- Multicast
- Broadcast
- Total packets per second
### Syntax

`asg_traffic_stats <ssm_id> | <if_name> [delay]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ssm_id&gt;</code></td>
<td>SSM name (1-4)</td>
</tr>
<tr>
<td></td>
<td>Shows the traffic statistics for the specified SSM</td>
</tr>
<tr>
<td><code>&lt;if_name&gt;</code></td>
<td>The interface name: eth1-04 or eth1-Sync</td>
</tr>
<tr>
<td></td>
<td>Shows the total traffic statistics for a specified SSM</td>
</tr>
<tr>
<td>delay</td>
<td>Length of time, in seconds, that traffic statistics are collected (Default = 5 seconds).</td>
</tr>
</tbody>
</table>

### Example - Traffic over one interface

```bash
# asg_traffic_stats eth1-04
Processing traffic statistics for 5 seconds...

eth1-04 statistics
---------------------
Incoming traffic:
---------------------
Throughput: 164.9 Kbps
Packet rate: [Total: 252 pps], [Unicast: 14 pps], [Multicast: 161 pps], [Broadcast: 76 pps]

Outgoing traffic:
---------------------
Throughput: 4.0 Kbps
Packet rate: [Total: 2 pps], [Unicast: 2 pps], [Multicast: 0 pps], [Broadcast: 0 pps]
```

### Example - Traffic over one SSM

```bash
# asg_traffic_stats 1
Processing traffic statistics for 5 seconds...

Summary on SSM1
---------------
Incoming traffic:
---------------
Throughput: 319.1 Kbps
Packet rate: [Total: 409 pps], [Unicast: 167 pps], [Multicast: 166 pps], [Broadcast: 75 pps]

Outgoing traffic:
---------------
Throughput: 408.2 Kbps
Packet rate: [Total: 156 pps], [Unicast: 156 pps], [Multicast: 0 pps], [Broadcast: 0 pps]
```

### Showing SGM Forwarding Statistics (asg_blade_stats)

Use this command to show detailed packet forwarding statistics.

### Syntax

`asg_blade_stats [-6] corr [[-p [-v]] [-a] [-reset]]
asg_blade_stats [-6] corr_online
asg_blade_stats [-6] iterator
asg_blade_stats [-6] smo
asg_blade_stats [-6] vpn [-v]
asg_blade_stats [-6] 6in4 [-v]
asg_blade_stats [-6] gre [-v]
asg_blade_stats [-6] icmp_error [-v] |
asg_blade_stats [-6] all
asg_blade_stats [-6] -h | Help`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-6</td>
<td>Show only IPv6 traffic</td>
</tr>
<tr>
<td>-p</td>
<td></td>
</tr>
<tr>
<td>-v</td>
<td>Show detailed statistics (verbose)</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>-a</td>
<td>Show aggregate statistics</td>
</tr>
<tr>
<td>-reset</td>
<td>Reset correction layer statistics (works only with the corr parameter)</td>
</tr>
<tr>
<td>corr</td>
<td>Show correction layer statistics per service (for predefined services) for each SGM.</td>
</tr>
<tr>
<td>iterator</td>
<td>Show information about the last iterator process</td>
</tr>
<tr>
<td>smo</td>
<td>Show statistics for SMO task and logs for each SGM</td>
</tr>
<tr>
<td>vpn</td>
<td>Show statistics for VPN forwarded packets</td>
</tr>
<tr>
<td>6in4</td>
<td>Show statistics for 6in4 tunnel forwarded packets</td>
</tr>
<tr>
<td>gre</td>
<td>Show statistics for GRE forwarded packets</td>
</tr>
<tr>
<td>icmp_error</td>
<td>Show statistics for ICMP ERROR forwarded packets</td>
</tr>
<tr>
<td>vs</td>
<td>Show Virtual System stateless correction layer statistics. (VSX Mode only)</td>
</tr>
<tr>
<td>all</td>
<td>Show all correction layer statistics mentioned above</td>
</tr>
<tr>
<td>help</td>
<td>Show help information</td>
</tr>
</tbody>
</table>

**Multi-blade capture (tcpdump –mcap -view)**

Use this command to see TCP/IP and other packets sent and received by the 61000/41000 Security System. This release includes these 61000/41000 Security System-specific enhancements to the standard tcpdump utility:

- tcpdump -mcap - Gets packets from specified SGMs and saves them to a capture file.
- tcpdump -view - Shows packets in the specified capture file, including the SGM ID from the packet captured packet.

**Syntax**

tcpdump [-b <sgm_ids>] -mcap -w <capture_path> [tcpdump_ops]
tcpdump -view -r <capture_path> [tcpdump_ops]

**Note** - To stop the capture and save the data to the capture file, enter ctl-c at the prompt.

**Parameter**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| -b <sgm_ids> | Works with SGMs and/or Chassis as specified by <sgm_ids>.  The <sgm_ids> can be:  
  - No <sgm_ids> specified or all shows all SGMs and Chassis  
  - One SGM  
  - A comma-separated list of SGMs (1_1,1_4)  
  - A range of SGMs (1_1-1_4)  
  - One Chassis (Chassis1 or Chassis2)  
  - The active Chassis (chassis_active) |
| -w <capture_path> | Saved file full path.  In addition to the merged capture file, per SGM capture files are created in the same directory, suffixed by their SGM ID. |
| -r <capture_path> | Read file full path. Regular tcpdump output, prefixed by SGM ID of the processing SGM ID. |
Example - Capture all SGMs
> tcpdump -mcap -w /tmp/capture
Capturing packets...
Write "stop" and press enter to stop the packets capture process.
1_01:
tcpdump: listening on eth1-Mgmt4, link-type EN10MB (Ethernet), capture size 96 bytes
Received user request to stop the packets capture process.

Example - Capture packets from specified SGMs and interfaces
> tcpdump -b 1_1,1_3,2_1 -mcap -w /tmp/capture -nnni eth1-Mgmt4

Example - Show captured packets from file
> tcpdump -view -r /tmp/capture
Reading from file /tmp/capture, link-type EN10MB (Ethernet)
[1_3] 14:11:57.971587 IP 0.0.0.0.cp-cluster > 172.16.6.0.cp-cluster: UDP, length 45
[2_3] 14:12:07.625171 IP 0.0.0.0.cp-cluster > 172.16.6.0.cp-cluster: UDP, length 45
[2_3] 14:12:09.974195 IP 0.0.0.0.cp-cluster > 172.16.6.0.cp-cluster: UDP, length 45
[2_1] 14:12:10.022995 IP 0.0.0.0.cp-cluster > 172.23.9.0.cp-cluster: UDP, length 32

Traceroute (asg_tracert)
Use this enhanced command to show correct traceret results on the 61000/41000 Security System. The native tracert cannot handle tracert pings correctly because of the stickiness mechanism used in the 61000/41000 Security System firewall. All native tracert command options and parameters are supported by asg_tracert.

Syntax
asg_tracert <ip> <tracert_options>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip&gt;</td>
<td>IP address</td>
</tr>
<tr>
<td>&lt;tracert_options&gt;</td>
<td>Native tracert command options.</td>
</tr>
</tbody>
</table>

Example
> asg_tracert <ip_address> <tracert_options>
traceroute to 100.100.100.99 (100.100.100.99), 30 hops max, 40 byte packets
1  (20.20.20.20)  0.722 ms  0.286 ms  0.231 ms
2  (100.100.100.99)  1.441 ms  0.428 ms  0.395 ms

Hardware Monitoring and Control

Showing Chassis and Component State (asg stat)
Use this command to show the Chassis and hardware component state for single and dual Chassis configurations. The command shows system:

- Up-time
- CPU load: average and current
- Concurrent connections
- Health

Use Verbose mode to show SGM state, process and policy

Syntax
asg stat
asg stat [-v] [-vs <vs_ids>] [-l]

Note - If you run this command in a VSX context, the output is for the applicable Virtual System.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Show detailed Chassis status (verbose mode).</td>
</tr>
</tbody>
</table>
| -vs <vs_ids> | Shows the Chassis status of Virtual Systems. The <vs_ids> can be:  
  - No <vs_ids> (default) - Shows the current Virtual System context.  
  - One Virtual System.  
  - A comma-separated list of Virtual Systems (1,2,4,5).  
  - A range of Virtual Systems (VS 3-5).  
  - all - Shows all Virtual Systems.  
**Note:** This parameter is only relevant in a VSX environment.  
For a Chassis with more than 3 SGMs, the output uses abbreviations to make the output more compact. |
| -l        | Show the meaning of the abbreviations in the output for a Chassis with more than 3 SGMs. |

### Chassis Status Summary

```
> asg stat
```

<table>
<thead>
<tr>
<th>VSX System Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up time</td>
</tr>
<tr>
<td>Current CPUs load average</td>
</tr>
<tr>
<td>Concurrent connections</td>
</tr>
<tr>
<td>Health</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Chassis 1</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Chassis 2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

**Notes**

The output shows that:

- Chassis 1 is in the STANDBY state
- Only three SGMs in Chassis 1 are UP, out of the 4 that are required
- 1 SGM and 2 Power Supplies in Chassis 1 are not running
Chassis Status Details

> asg stat -v

Output (Top Section)

<table>
<thead>
<tr>
<th>VSX System Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>VS ID: 0</td>
</tr>
<tr>
<td>VS Name: Athens</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>STANDBY</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SGM ID</th>
<th>State</th>
<th>Process</th>
<th>Policy Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (local)</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
<tr>
<td>2</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
<tr>
<td>3</td>
<td>DOWN</td>
<td>Inactive</td>
<td>NA</td>
</tr>
<tr>
<td>4</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACTIVE</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SGM ID</th>
<th>State</th>
<th>Process</th>
<th>Policy Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
<tr>
<td>2</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
<tr>
<td>3</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
<tr>
<td>4</td>
<td>UP</td>
<td>Enforcing Security</td>
<td>09Jan14 11:30</td>
</tr>
</tbody>
</table>

Notes

This output shows that:

- **Chassis 1** is STANDBY with 3 SGMs up.
- **Chassis 2** is in ACTIVE state with 4 SGMs up.
- **SGM ID** is the Identifier of the SGM. (local) is the SGM on which you ran the command.
- **State** is the state of the SGM. Can be:
  - **Up** - The SGM is processing traffic
  - **Down** - The SGM is not processing traffic
  - **Detached** - No SGM has been detected in a slot.
- **Note** - To manually change the state of an SGM, use the asg sgm admin command. This command administratively changes the state to up or down. An SGM that is down because of a software or hardware problem cannot be changed to UP using this command.
- **Process** is the state of the SGM security enforcement:
  - **Enforcing Security** - UP and working properly.
  - **Inactive** - DOWN, and is experiencing some problem. It is not handling any traffic.
  - **Initial policy** - The SGM is UP but the policy is not installed on the SGM.
Output (Bottom Section)

<table>
<thead>
<tr>
<th>Chassis Parameters</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit</strong></td>
<td><strong>Chassis 1</strong></td>
</tr>
<tr>
<td>SGMs</td>
<td>4 / 4</td>
</tr>
<tr>
<td>Ports</td>
<td>0 / 0</td>
</tr>
<tr>
<td>Standard</td>
<td>2 / 2</td>
</tr>
<tr>
<td>Bond</td>
<td>0 / 0</td>
</tr>
<tr>
<td>Other</td>
<td>6</td>
</tr>
<tr>
<td>Sensors</td>
<td>6 / 6</td>
</tr>
<tr>
<td>Fans</td>
<td>2 / 2</td>
</tr>
<tr>
<td>SSMs</td>
<td>2 / 2</td>
</tr>
<tr>
<td>CMMs</td>
<td>3 / 3</td>
</tr>
<tr>
<td>Power Supplies</td>
<td>134 / 134</td>
</tr>
<tr>
<td>Chassis Grade</td>
<td>11</td>
</tr>
<tr>
<td>Minimum grade gap for chassis failover:</td>
<td>11</td>
</tr>
<tr>
<td><strong>Synchronization</strong></td>
<td></td>
</tr>
<tr>
<td>Within chassis:</td>
<td>Enabled (Default)</td>
</tr>
<tr>
<td>Between chassis:</td>
<td>Enabled (Default)</td>
</tr>
<tr>
<td>Exception Rules:</td>
<td>(Default)</td>
</tr>
<tr>
<td>Chassis HA mode:</td>
<td>Active Up</td>
</tr>
<tr>
<td>Chassis HA in Freeze</td>
<td>(5 seconds left)</td>
</tr>
</tbody>
</table>

Notes

- The X/X notation shows the number of components that are up and the components must be up. For example, on the SGMs line, 4/4 means that 4 SGMs are up must be up.

- **Chassis grade** is the sum of the grades of all components. In a dual-chassis deployment, the chassis with a higher grade (by at least the Minimum grade gap) becomes ACTIVE. The grade of each component = Unit Weight x the number of components that are UP. The Unit Weight of each component can be configured to reflect the importance of the component in the system. To configure the Unit Weight run:

  ```
  set chassis high-availability factors <sensor name>
  ```

  For example if you wish to change the weight of the SGM from 6 to 12, run:

  ```
  set chassis high-availability factors sgm 12
  ```

  If you run `asg stat -v`, the output shows a higher unit weight and Chassis Grade:

- **Minimum threshold for traffic processing** - The minimum grade required for the chassis to become ACTIVE.

- **Minimum grade gap for chassis failover** - Chassis failover occurs to the chassis with the higher grade only if its grade is greater than the other chassis by more than the minimum gap.

- **Synchronization** - The status of synchronization:
  - **Within chassis** - between SGMs located in the same chassis.
  - **Between chassis** - between SGMs located in different chassis.
  - **Exception Rules** - user configured exception rules. To configure, use the command `g_sync_exception`.

- **Distribution Control blade** - Shows if this option is enabled. When enabled, the SMO handles only management traffic. You always have immediate access to the system with an SSH connection.
### Compact Output for Selected SGMs

```sh
> asg stat -v -vs 0,1,2
```

<table>
<thead>
<tr>
<th>Chassis 1</th>
<th>STANDBY</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SGM</strong> 1</td>
<td><strong>ES</strong> 2</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td><strong>UP</strong></td>
</tr>
<tr>
<td><strong>VS ID</strong></td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 2</th>
<th>ACTIVE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SGM</strong> 1 (1)</td>
<td><strong>ES</strong> 2</td>
</tr>
<tr>
<td><strong>State</strong></td>
<td><strong>UP</strong></td>
</tr>
<tr>
<td><strong>VS ID</strong></td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Chassis 1</th>
<th>Chassis 2</th>
<th>Unit Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGMs</td>
<td>3 / 4 (!)</td>
<td>4 / 4</td>
<td>6</td>
</tr>
<tr>
<td>Ports</td>
<td>Standard</td>
<td>0 / 0</td>
<td>0 / 0</td>
</tr>
<tr>
<td>Sensors</td>
<td>Other</td>
<td>0 / 0</td>
<td>0 / 0</td>
</tr>
<tr>
<td>Fans</td>
<td>6 / 6</td>
<td>6 / 6</td>
<td>5</td>
</tr>
<tr>
<td>SSMs</td>
<td>2 / 2</td>
<td>2 / 2</td>
<td>11</td>
</tr>
<tr>
<td>CMMs</td>
<td>2 / 2</td>
<td>2 / 2</td>
<td>6</td>
</tr>
<tr>
<td>Power Supplies</td>
<td>6 / 6</td>
<td>6 / 6</td>
<td>6</td>
</tr>
<tr>
<td>Chassis Grade</td>
<td>118 / 124</td>
<td>124 / 124</td>
<td>-</td>
</tr>
<tr>
<td>Minimum grade gap for chassis failover:</td>
<td>11</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Synchronization</td>
<td>Within chassis: Enabled (Default)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Between chassis: Enabled (Default)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Exception Rules: (Default)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Distribution</td>
<td>Control Blade: Disabled (Default)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Chassis HA mode:</td>
<td>Active Up</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Notes:**
- `(-)` indicates a disabled state.
- `(!)` indicates an active state.
- `UP` indicates an active and operational state.
- `DOWN` indicates an inactive or failed state.
Output Appreciations Legend

asg stat -l
Legend:

SGM States:

ACT - ACTIVE                  DTC - DETACHED
DWN - DOWN                     NSG - NOT IN SECURITY GROUP

VS States:

ES - Enforcing Security       FSC - FullSync Client
FSS - FullSync Server         IAC - Inactive
IF - Iteration Finished       IFO - Initial Policy
IS - Iteration Started        NPO - No Policy
FC - Policy Completed         FRF - Policy Ready2Finish
PS - Policy Started

Monitoring Chassis and Component Status (asg monitor)

Use this command to continuously monitor Chassis and component status. This command shows the same information as asg stat, but the information stays on the screen and refreshes at user-specified intervals (default = 1 second). To end the monitor session, press Ctrl-c.

Note - If you run this command in a Virtual System context, you will see only the output for that Virtual System. You can also specify the Virtual System as a command parameter.

Syntax

asg monitor  [-v <interval>]
asg monitor [-amw] <interval>
asg monitor [-amw] -vs <vs_ids> <interval>
asg monitor -l
asg monitor -h

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show the command syntax and help information.</td>
</tr>
<tr>
<td>-amw</td>
<td>Shows the Anti-Malware policy date instead of the Firewall policy date.</td>
</tr>
<tr>
<td>-v</td>
<td>Shows only Chassis component status.</td>
</tr>
<tr>
<td>-all</td>
<td>Shows both SGM and Chassis component status.</td>
</tr>
<tr>
<td>&lt;interval&gt;</td>
<td>Sets the data refresh interval (in seconds) for this session.</td>
</tr>
<tr>
<td>-vs &lt;vs_ids&gt;</td>
<td>Shows the component status for one or more Virtual Systems. The &lt;vs_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>* No &lt;vs_ids&gt; (default) - Shows the current Virtual System context.</td>
</tr>
<tr>
<td></td>
<td>* One Virtual System.</td>
</tr>
<tr>
<td></td>
<td>* A comma-separated list of Virtual Systems (1,2,4,5).</td>
</tr>
<tr>
<td></td>
<td>* A range of Virtual Systems (VS 3-5).</td>
</tr>
<tr>
<td></td>
<td>* all - Shows all Virtual Systems.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: This parameter is only relevant in a VSX environment.</td>
</tr>
</tbody>
</table>

For a Chassis with more than 3 SGMs, the output has abbreviations to make the output more compact.

- l        | Shows legend of column title abbreviations. |
- h        | Shows the command syntax and help information. |

Note: asg monitor with no parameters shows the SGM status.
Examples

This example shows the SGM status with the Anti-Malware policy date.

```
> asg monitor -amw
```

```
<table>
<thead>
<tr>
<th>Chassis 1</th>
<th>ACTIVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGM ID</td>
<td>State</td>
</tr>
<tr>
<td>1</td>
<td>UP</td>
</tr>
<tr>
<td>2 (local)</td>
<td>UP</td>
</tr>
<tr>
<td>3</td>
<td>UP</td>
</tr>
<tr>
<td>4</td>
<td>UP</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 2</th>
<th>STANDBY</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGM ID</td>
<td>State</td>
</tr>
<tr>
<td>1</td>
<td>UP</td>
</tr>
<tr>
<td>2</td>
<td>UP</td>
</tr>
<tr>
<td>3</td>
<td>UP</td>
</tr>
<tr>
<td>4</td>
<td>UP</td>
</tr>
</tbody>
</table>

This example shows the Chassis component status.

```
> asg monitor -v
```

```
<table>
<thead>
<tr>
<th>Chassis Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
</tr>
<tr>
<td>--------------------</td>
</tr>
<tr>
<td>SGMs</td>
</tr>
<tr>
<td>Ports</td>
</tr>
<tr>
<td>Standard</td>
</tr>
<tr>
<td>Bond</td>
</tr>
<tr>
<td>Mgmt</td>
</tr>
<tr>
<td>Other</td>
</tr>
<tr>
<td>Sensors</td>
</tr>
<tr>
<td>Fans</td>
</tr>
<tr>
<td>SSMs</td>
</tr>
<tr>
<td>CMMs</td>
</tr>
<tr>
<td>Power Supplies</td>
</tr>
<tr>
<td>Chassis Grade</td>
</tr>
<tr>
<td>Minimum grade gap for chassis failover:</td>
</tr>
<tr>
<td>Synchronization</td>
</tr>
<tr>
<td>Within chassis:</td>
</tr>
<tr>
<td>Between chassis:</td>
</tr>
<tr>
<td>Exception Rules:</td>
</tr>
<tr>
<td>Chassis HA mode:</td>
</tr>
</tbody>
</table>

This example shows the status of the SGMS and Virtual System 3.
**Monitoring Performance (asg perf)**

Use this command to continuously monitor key performance indicators and load statistics. There are different commands for IPv4 and IPv6. You can show the performance statistics for IPv4 traffic, IPv6 traffic or for all traffic.

When you run `asg perf`, the statistics display shows on the screen. The display is automatically updated after a predefined interval (default = 10 seconds). To stop `asg perf` and return to the command line, press `e`.

**Syntax**

```plaintext
asg perf -h
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-h</code></td>
<td>Shows command syntax with help</td>
</tr>
</tbody>
</table>
| `-b <sgm_ids>` | Works with SGMs and/or Chassis as specified by `<sgm_ids>`.
|           | The `<sgm_ids>` can be: |
|           | - No `<sgm_ids>` specified or all shows all SGMs and Chassis |
|           | - One SGM |
|           | - A comma-separated list of SGMs (1_1,1_4) |
|           | - A range of SGMs (1_1-1_4) |
|           | - One Chassis (Chassis1 or Chassis2) |
|           | - The active Chassis (chassis_active) |
| `-vs <vs_ids>` | For VSX Gateways only. Shows performance for Virtual Systems as specified by `<vs_ids>`.
<p>|           | The <code>&lt;vs_ids&gt;</code> can be: |
|           | - No <code>&lt;vs_ids&gt;</code> (default) - Shows the current Virtual System context. |
|           | - One Virtual System. |
|           | - A comma-separated list of Virtual Systems (1,2,4,5). |
|           | - A range of Virtual Systems (VS 3-5). |
|           | - all - Shows all Virtual Systems. |
| <strong>Note:</strong> This parameter is only relevant in a VSX environment. |
| <code>-v</code>      | Shows statistics per SGM. |
| <code>-vv</code>     | Shows statistics per Virtual System. |
| <code>-p</code>      | Show detailed statistics and traffic distribution between these paths on the Active Chassis: |
|           | - Acceleration path (Performance Pack). |
|           | - Medium path (PXL). |
|           | - Slow path (Firewall). |
| <code>-4|-6</code>    | <code>-4</code> shows IPv4 information only. |
|          | <code>-6</code> shows IPv6 information only. |
|          | If no value is specified, the combined performance information for both IPv4 and IPv6 shows. |
| <code>-c</code>      | Show percentages instead of absolute values. |
| <code>-k</code>      | Show peak (maximum) system performance values. |</p>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--peak_hist</td>
<td>Creates an exportable text file that contains all data saved in the peak performance files. You must use this parameter together with -k.</td>
</tr>
<tr>
<td>--perf_hist</td>
<td>Creates exportable text files that contains all performance data saved in the history files. You must use this parameter together with -k.</td>
</tr>
<tr>
<td>-e</td>
<td>Reset peak values and delete all peaks files and system history files.</td>
</tr>
<tr>
<td>--delay &lt;seconds&gt;</td>
<td>Temporarily changes the update interval for the current asg perf session. Enter a delay value in seconds. Default = 10 seconds</td>
</tr>
</tbody>
</table>

Notes:

- The -b <sgm_ids> and -vs <vs_vs_ids> parameters must written be at the beginning of the command string. If both parameters are used, -b <sgm_ids> must be written first.
- When you run asg perf, it continues to show performance information, which is automatically updated after a predefined period of time (default = 10 seconds). The command line is not available while asg perf is running.
- If your 61000/41000 Security System is not configured for VSX, the VSX related commands are not available. They do not show when you run asg perf -h.

Summary without Parameters

> asg perf
Sun Oct 20 11:09:07 IST 2013
Aggregated statistics (IPv4 and IPv6) of SGMs: chassis_active Virtual Systems: 0
SXL is disabled on: [ipv4:2_02],[ipv4:2_03],[ipv4:2_04]

<table>
<thead>
<tr>
<th>Performance Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
</tr>
<tr>
<td>------</td>
</tr>
<tr>
<td>Throughput</td>
</tr>
<tr>
<td>Packet rate</td>
</tr>
<tr>
<td>Connection rate</td>
</tr>
<tr>
<td>Concurrent connections</td>
</tr>
<tr>
<td>Load average</td>
</tr>
<tr>
<td>Acceleration load (avg/min/max)</td>
</tr>
<tr>
<td>Instances load (avg/min/max)</td>
</tr>
<tr>
<td>Memory usage</td>
</tr>
</tbody>
</table>

Notes:

- By default, absolute values are shown
- Unless otherwise specified, the combined statistics for IPv4 and IPv6 are shown
- When no SGMs are specified, performance statistics are shown for the active SGM only
Output with Performance Summary

The `-v` parameter adds a performance summary for each SGM.

```shell
> asg perf -v
Tue Oct 22 07:23:37 IST 2013
Aggregated statistics (IPv4 and IPv6) of SGMs: chassis_active Virtual Systems: 0
+-------------------------------+-------------------+-------------------+
|Performance Summary            |                   |
+-------------------------------+-------------------+-------------------+
|Name                           |Value              |IPv4%              |
+-------------------------------+-------------------+-------------------+
|Throughput                     |10.2 K             |100%               |
|Packet rate                    |11                 |100%               |
|Connection rate                |0                  |N/A                |
|Concurrent connections         |22                 |100%               |
|Load average                   |7%                 |
|Acceleration load (avg/min/max)|6%/6%/6%           |
|Instances load (avg/min/max)   |5%/4%/9%           |
|Memory usage                   |55%                |
+-------------------------------+-------------------+-------------------+
```

Notes

- By default, absolute values are shown
- Unless otherwise specified, the combined statistics for IPv4 and IPv6 are shown
- When no SGMs are specified, performance statistics are shown for the active SGM only

Per Path Statistics

This example shows detailed performance information per SGM and traffic distribution between different paths. It also shows VPN throughput and connections.

```shell
> asg perf -p -v
Tue Oct 22 07:31:31 IST 2013
Aggregated statistics (IPv4 and IPv6) of SGMs: chassis_active Virtual Systems: 0
+-------------------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+
|Per SGM Distribution Summary |                   |                   |                   |                   |                   |                   |                   |                   |
|SGM  |Throughput |Packet rate |Conn. rate |Conc. Conn. |Accel. Rate |Instances Rate |Mem. Usage |
|ID   |           |Rate       |Rate       |Conn        |Cores%      |Cores%         |Usage%     |
+-------------------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+
|_01  |10.2 K     |11          |0          |22          |6/6/6       |5/4/9          |55%        |
+-------------------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+
|Total|10.2 K     |11          |0          |22          |6/6/6       |5/4/9          |55%        |
+-------------------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+-------------------+
```
history files are created based on a predefined interval (Default = every 4 hours). New peak value files are created whenever a new peak value is detected. These files are located at \(/var/log/asgstats\).
The system saves these files until a predefined maximum number of files is reached, after which files are deleted on an oldest first basis. You can also delete all history and peak value files manually.

System performance data includes these parameters:

- Throughput
- Packet rate
- Connection rate
- Concurrent connections
- Acceleration load
- Firewall load
- Memory consumption

You can collect the data contained in the historical peak value files and save them into two comma-separated-value text files. There is one combined file for historical system performance data and another for peak values. You can export these files and analyze them in a spreadsheet or statistical analysis application. The combined files are saved at $FWDIR/conf/asgpeaks.conf.

**To create the combined text files, run:**

```bash
> asg perf -k --last  
> asg perf -k --hist
```

**To delete the history and peak value files, run:**

```bash
> asg perf -k -e
```

---

**Monitoring SGM Resources (asg resource)**

Use this command to show SGM resource usage and thresholds for the whole 61000/41000 Security System.

**Syntax**

```bash
asg resource <-b sgm_string>  
asg resource -h
```

**Parameter** | **Description**
--- | ---
-b sgm | Works with SGMs and/or Chassis as specified by <sgm_ids>.  
The <sgm_ids> can be:
- No <sgm_ids> specified or all shows all SGMs and Chassis
- One SGM
- A comma-separated list of SGMs (1_1,1_4)
- A range of SGMs (1_1-1_4)
- One Chassis (Chassis1 or Chassis2)
- The active Chassis (chassis_active)

-h | Shows usage and exits

**Example**

```bash
> asg resource  
|Resource Table  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SGM ID</td>
<td>Resource Name</td>
<td>Usage</td>
<td>Threshold</td>
</tr>
<tr>
<td>1_01</td>
<td>Memory</td>
<td>31%</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>HD: /</td>
<td>30%</td>
<td>80%</td>
</tr>
<tr>
<td></td>
<td>HD: /var/log</td>
<td>3%</td>
<td>80%</td>
</tr>
<tr>
<td></td>
<td>HD: /boot</td>
<td>19%</td>
<td>80%</td>
</tr>
<tr>
<td>1_02</td>
<td>Memory</td>
<td>31%</td>
<td>50%</td>
</tr>
<tr>
<td></td>
<td>HD: /</td>
<td>30%</td>
<td>80%</td>
</tr>
<tr>
<td></td>
<td>HD: /var/log</td>
<td>2%</td>
<td>80%</td>
</tr>
<tr>
<td></td>
<td>HD: /boot</td>
<td>19%</td>
<td>80%</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>SGM</th>
<th>Memory</th>
<th>Usage</th>
<th>Threshold</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_03</td>
<td>31%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>80%</td>
<td>19.4G</td>
<td>288.6M</td>
</tr>
<tr>
<td></td>
<td>2%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
<tr>
<td>1_04</td>
<td>30%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>29%</td>
<td>80%</td>
<td>19.4G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
<tr>
<td>2_01</td>
<td>31%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>80%</td>
<td>19.4G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
<tr>
<td>2_02</td>
<td>31%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>80%</td>
<td>19.4G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
<tr>
<td>2_03</td>
<td>31%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>80%</td>
<td>19.4G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
<tr>
<td>2_04</td>
<td>31%</td>
<td>50%</td>
<td>31.3G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30%</td>
<td>80%</td>
<td>19.4G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1%</td>
<td>80%</td>
<td>58.1G</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19%</td>
<td>80%</td>
<td>288.6M</td>
<td></td>
</tr>
</tbody>
</table>

**Notes**

- The **SGM** column shows the SGM ID.
- The **Resource** column identifies the resource. There are four types of resources:
  - Memory
  - HD – hard drive space (/)
  - HD: /var/log – space on hard drive committed to log files
  - HD: /boot - location of the kernel
- The **Usage** column shows the percentage of the resource in use.
- The **Threshold** gives an indication of the health and functionality of the component. When the value of the resource is greater than the threshold, an alert is sent. The threshold can be modified in gclish.
- The **Total** column is the total absolute value in units

For example, the first row shows that SGM1 on Chassis 1 has 31.3 Gb of memory, 31% of which is used. An alert will be sent if the usage is greater than 50%.

**Searching for a Connection (asg search)**

You can use this command to:

- Search for a connection or a filtered list of connections.
- See which SGM handles the connection (actively or as backup), and on which Chassis.

You can run this command directly from the command line or in the interactive mode, which lets you enter the parameters in the correct order. The asg search command also runs a consistency test between SGMs. This command supports both IPv6 and IPv4 connections.

**Searching with the Command Line**

**Syntax**

```
asg search [-v] [-v <vs_ids>] <source_ip> <dest_ip> <dest_port> <protocol>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-help</td>
<td>Show the command syntax and help text.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Without connection parameters</td>
<td>Run in the interactive mode.</td>
</tr>
<tr>
<td>-vs &lt;vs_ids&gt;</td>
<td>Shows connections for the specified Virtual System. The &lt;vs_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>• No &lt;vs_ids&gt; (default) - Shows the current Virtual System context.</td>
</tr>
<tr>
<td></td>
<td>• One Virtual System.</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of Virtual Systems (1,2,4,5).</td>
</tr>
<tr>
<td></td>
<td>• all - Shows all Virtual Systems.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> This parameter is only relevant in a VSX environment.</td>
</tr>
<tr>
<td>&lt;source_ip&gt;</td>
<td>Source IPv4 or IPv6 address.</td>
</tr>
<tr>
<td>&lt;dest_ip&gt;</td>
<td>Destination IPv4 or IPv6 address</td>
</tr>
<tr>
<td>&lt;dest_port&gt;</td>
<td>Destination port number.</td>
</tr>
<tr>
<td>&lt;protocol&gt;</td>
<td>IP Protocol.</td>
</tr>
<tr>
<td>&lt;source_port&gt;</td>
<td>Source port number.</td>
</tr>
<tr>
<td>-v</td>
<td>Shows connection indicators for</td>
</tr>
<tr>
<td></td>
<td>• F - Firewall connection table</td>
</tr>
<tr>
<td></td>
<td>• S - SecureXL connection table</td>
</tr>
<tr>
<td></td>
<td>• C - Correction Layer table</td>
</tr>
<tr>
<td></td>
<td>This in addition to the indicators for Active and Backup SGM.</td>
</tr>
</tbody>
</table>

**Notes:**
- You must enter the all parameters in the order shown in the above syntax.
- You can enter the \\" character as a parameter to show all values for that parameter.
- The -vs parameter is only available for a 61000/41000 Security System running VSX.

**Command Line Examples**

**One IPv4 source and destination for the TCP protocol**

```bash
> asg search -v 192.0.2.4 192.0.2.15 * tcp
Lookup for conn: 192.0.2.4, 192.0.2.15, *, tcp>, may take few seconds...

<192.0.2.4, 1130, 192.0.2.15, 49829, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36323, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49851, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36308, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36299, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49835, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49856, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36331, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49857, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49841, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36315, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 1130, 192.0.2.15, 49859, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36300, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]
<192.0.2.4, 36301, 192.0.2.15, 1130, tcp> -> [2_01 A, 1_04 A]

Legend:
A - Active SGM
B - Backup SGM
C - Correction Layer table
F - Firewall connection table
S - SecureXL connection table
One IPv6 source, all destinations, source port 8080, and the TCP protocol

> asg search 2620:0:2a03:16:2:33:0:1 \* 8080 tcp

<2620:0:2a03:16:2:33:0:1, 52117, 951::69cb:e42d:eac0:652f, 8080, tcp> --> [1_01 A, 2_01 B]
<2620:0:2a03:16:2:33:0:1, 62775, 951::69cb:e42d:eac0:652f, 8080, tcp> --> [1_01 A, 2_01 B]
<2620:0:2a03:16:2:33:0:1, 54378, 951::69cb:e42d:eac0:652f, 8080, tcp> --> [1_01 A, 2_01 B]

Legend:
A - Active SGM
B - Backup SGM

All sources, destinations, ports and protocols for VS0

> asg search -vs 0 \* \* \* \* \* \* .

Lookup for conn: <*, *, *, *, *>, may take few seconds...

<172.23.9.130, 18192, 172.23.9.138, 43563, tcp> --> [1_01 A]
<172.23.9.130, 32888, 172.23.9.138, 257, tcp> --> [1_01 A]
<172.23.9.130, 22, 194.29.47.14, 52120, tcp> --> [1_01 A]
<172.23.9.138, 257, 172.23.9.130, 32963, tcp> --> [1_01 A]
<172.23.9.130, 22, 194.29.47.14, 52104, tcp> --> [1_01 A]
<255.255.255.255, 67, 0.0.0.0, 68, udp> --> [1_01 A]
<172.23.9.138, 257, 172.23.9.130, 32864, tcp> --> [1_01 A]
<172.23.9.138, 257, 172.23.9.130, 32888, tcp> --> [1_01 A]
<172.23.9.138, 257, 172.23.9.130, 33465, tcp> --> [1_01 A]
<172.23.9.130, 22, 194.29.40.23, 65515, tcp> --> [1_01 A]
<172.23.9.130, 22, 194.29.47.14, 52493, tcp> --> [1_01 A]
<172.23.9.130, 18192, 172.23.9.138, 49059, tcp> --> [1_01 A]
<172.23.9.130, 18192, 172.23.9.137, 33356, tcp> --> [1_01 A]
<172.23.9.138, 33356, 172.23.9.130, 18192, tcp> --> [1_01 A]
<172.23.9.130, 43563, 172.23.9.130, 18192, tcp> --> [1_01 A]
<172.23.9.130, 32864, 172.23.9.130, 257, tcp> --> [1_01 A]
<0.0.0.0, 68, 255.255.255.255, 67, udp> --> [1_01 A]
<172.23.9.130, 32963, 172.23.9.138, 257, tcp> --> [1_01 A]
<172.23.9.130, 33465, 172.23.9.138, 257, tcp> --> [1_01 A]
<194.29.47.14, 52120, 172.23.9.130, 22, tcp> --> [1_01 A]
<194.29.47.14, 52104, 172.23.9.130, 22, tcp> --> [1_01 A]
<fe80::d840:5de7:8dbe:2345, 546, ff02::1:12, 547, udp> --> [1_01 A]
<194.29.47.14, 52493, 172.23.9.138, 22, tcp> --> [1_01 A]
<172.23.9.138, 49059, 172.23.9.130, 18192, tcp> --> [1_01 A]
<194.29.40.23, 65515, 172.23.9.130, 22, tcp> --> [1_01 A]

Legend:
A - Active SGM
B - Backup SGM

Searching with the Interactive Mode

The interactive mode lets you enter connection search parameters interactively in the required sequence as an alternative to the command line syntax.

To run asg search in the interactive mode:

1. Run:
   > asg search [-vs <vs_ids>] [-v]

2. Enter these parameters in order.
   - Source IPv4 or IPv6 address
   - Destination IPv4 or IPv6 address
   - Destination port number
   - IP protocol
   - Source port number

You can enter the '*' character to show all values for any parameter.
**Interactive Mode Examples**

**Example 1 - One IPv4 source and destination with -v**

```bash
> asg search -v
```

Please enter conn's 5 tuple:
```
-----------------------
Enter source IP (press enter for wildcard):
>192.0.2.4
Enter destination IP (press enter for wildcard):
>192.0.2.15
Enter destination port (press enter for wildcard):
>
Enter IP protocol ('tcp', 'udp', 'icmp' or enter for wildcard):
>tcp
Enter source port (press enter for wildcard):
>
Lookup for conn: <192.0.2.4, *, 192.0.2.15, *, tcp>, may take few seconds...
```

```
<192.0.2.4, 37408, 192.0.2.15, 1130, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 1130, 192.0.2.15, 37408, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 1130, 192.0.2.15, 49653, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 37406, 192.0.2.15, 1130, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 1130, 192.0.2.15, 49663, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 1130, 192.0.2.15, 49658, tcp> - > [2_01 AF, 1_04 AF]
<192.0.2.4, 37407, 192.0.2.15, 1130, tcp> - > [2_01 AF, 1_04 AF]
```

Legend:
A - Active SGM
B - Backup SGM
C - Correction Layer table
F - Firewall connection table
S - SecureXL connection table

**Example 2 - One IPv6 source with any Destination on port 8080 and TCP**

```bash
> asg search 2620::0:2a03:16:2::3:0:1 */* 8080 tcp
```

Enter source IP (press enter for wildcard):
```
> 2620:0:2a03:16:2:0:1
```

Enter destination IP (press enter for wildcard):
```
>
```

Enter destination port (press enter for wildcard):
```
>8080
```

Enter IP protocol ('tcp', 'udp', 'icmp' or enter for wildcard):
```
>tcp
```

Lookup for conn: <2620:0:2a03:16:2:33:0:1, *, *, 8080, tcp>, may take few seconds...
```

```
<2620:0:2a03:16:2:33:0:1, 52117, 951::69cb:e42d:eac0:652f, 8080, tcp> - > [1_01 A, 2_01 B]
<2620:0:2a03:16:2:33:0:1, 62775, 951::69cb:e42d:eac0:652f, 8080, tcp> - > [1_01 A, 2_01 B]
<2620:0:2a03:16:2:33:0:1, 54378, 951::69cb:e42d:eac0:652f, 8080, tcp> - > [1_01 A, 2_01 B]
```

Legend:
A - Active SGM
B - Backup SGM

**Configuring Alerts for SGM and Chassis Events (asg alert)**

The `asg alert` utility is an interactive wizard used to configure alerts for SGM and Chassis events. Event types can include hardware failure, recovery, and performance related events. You can also create events for other, general events.

An alert is sent when an event occurs. For example, an alert is generated when the value of a hardware resource is greater than the threshold. The alert message includes the Chassis ID, SGM ID and/or unit ID, as applicable.
The wizard includes these options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full Configuration Wizard</td>
<td>Create a new alert</td>
</tr>
<tr>
<td>Edit Configuration</td>
<td>Change an existing alert</td>
</tr>
<tr>
<td>Show Configuration</td>
<td>Show existing alert configurations</td>
</tr>
<tr>
<td>Run Test</td>
<td>Run a test simulation to make sure that the alert works correctly</td>
</tr>
</tbody>
</table>

To create or change an alert:

1. Run:
   > asg alert
2. Select and configure these parameters as prompted by the wizard:
   - Alert type and related parameters
   - Event types
   - Alert mode

These sections include details about the alert parameters that you configure with the wizard.

SMS alert parameters
- **SMS Provider URL** - Fully qualified URL to your SMS provider based on this syntax.
- **HTTP proxy and port** (Optional) – Necessary only if your Security Gateway requires a proxy server to reach the SMS provider.
- **SMS rate limit** - Maximum number of SMS messages sent per hour. When there are too many messages, the others are sent together as one message.
- **SMS user text** - Custom prefix for SMS messages

Email alert configuration:
- **SMTP server IP** - Configure one or more SMTP servers to which the email alerts will be sent.
- **Email recipient addresses** - Configure one or more recipient email addresses for each SMTP server.
- **Periodic connectivity checks** - Run a periodic test to make sure that there is connectivity with the SNMP servers. If there is no connectivity, alert messages are saved and sent in one email when connectivity is restored.
- **Interval** - Define the interval, in minutes, between connectivity tests.
- **Sender email address** - Configure a sender email address for email alerts.
- **Subject** - Subject header text for the email alert.
- **Body text** - Enter user-defined text for the alert message.

SNMP alert parameters
Define one or more SNMP managers to get SNMP traps sent from the Security Gateway. For each manager, configure these parameters as prompted:

**Note:** Some parameters do not show, based on your settings.
- **SNMP manager name** - Configure a name for your SNMP manager (unique)
- **SNMP manager IP** - Configure the manager IP address (trap receiver)
- **SNMP version** - Select the SNMP version to use (v2cv3)
- **SNMP v3 user name** - If using SNMP v3 authentication, you must configure this.
- **SNMP v3 engine ID** - Unique SNMP v3 engine ID used by your system. Default = [0x80000000010203EA].
- **SNMP v3 authentication protocol** - MD5 or SHA.
- **SNMP v3 authentication password** - Enter a privacy password.
- **SNMP v3 privacy protocol** - DES or AES.
- **SNMP v3 privacy password** - Enter a privacy password.
- **SNMP user text** - Custom text for the SNMP trap messages.
- **SNMP community string** - Configure the community string for the SNMP manager.

**Log alert parameters**

There are no configurable parameters for log alerts

**Event types**

You can select one or more event types:

- One event type
- A comma-delimited list of more than one event type
- **all** for all event types.

```
<table>
<thead>
<tr>
<th></th>
<th>SGM State</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Chassis State</td>
</tr>
<tr>
<td>3</td>
<td>Port State</td>
</tr>
<tr>
<td>4</td>
<td>Pingable Hosts State</td>
</tr>
<tr>
<td>5</td>
<td>System Monitor Daemon</td>
</tr>
<tr>
<td>6</td>
<td>Route State</td>
</tr>
<tr>
<td>7</td>
<td>Diagnostics</td>
</tr>
</tbody>
</table>

Hardware Monitor events:

<table>
<thead>
<tr>
<th></th>
<th>Fans</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>SSM</td>
</tr>
<tr>
<td>10</td>
<td>CMM</td>
</tr>
<tr>
<td>11</td>
<td>Power Supplies</td>
</tr>
<tr>
<td>12</td>
<td>CPU Temperature</td>
</tr>
</tbody>
</table>

Performance events:

<table>
<thead>
<tr>
<th></th>
<th>Concurrent Connections</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Connection Rate</td>
</tr>
<tr>
<td>15</td>
<td>Packet Rate</td>
</tr>
<tr>
<td>16</td>
<td>Throughput</td>
</tr>
<tr>
<td>17</td>
<td>CPU Load</td>
</tr>
<tr>
<td>18</td>
<td>Hard Drive Utilization</td>
</tr>
<tr>
<td>19</td>
<td>Memory Utilization</td>
</tr>
</tbody>
</table>
```

**Alert Modes**

- **Enabled** - An alert is sent for the selected events
- **Disabled** - No alert is sent for the selected events
- **Monitor** - A log entry is generated instead of an alert

**Diagnostic Events**

We recommend that you run the `asg diag verify` diagnostic tests periodically. Alerts are sent if there are failed tests. The alerts continue with the Message of the Day (MOTD) until the issues are resolved. You can optionally disable the MOTD.

When the issues that caused failed tests are resolved, a "Clear Alert" message is automatically sent the next time that the test runs. You can also run `asg diag verify` manually to make sure that the issue is resolved.

By default, the test runs daily at 01:00. You can change the default time as necessary.

**To change the default time:**

1. Open `/var/opt/CPsuite-R76/fw1/conf/asgsnmp.conf` in a text editor.
2. Change the `asg_diag_alert_wrapper=` parameter as necessary.
3. Run `asg_cp2blades <file_path_name>` to copy this file to all other SGMs.

**To disable the MOTD:**

1. Open `/var/opt/CPsuite-R76/fw1/conf/asg_diag_config` in a text editor.
2. Add this line to the file:

   ```
motd=off
   ```
3. Run `asg_cp2blades <file_path_name>` to copy this file to all other SGMs.

**Collecting System Diagnostics (asg diag)**

Use this command to collect and show diagnostic information. This command runs a list of predefined diagnostic tests. The output shows the result of each test (Passed or Failed) and the location of the output log file.

**Syntax**

- `asg diag list [Test1][,Test2,...]`
- `asg diag verify [Test1][,Test2,...]`
- `asg diag print [Test1][,Test2,...]`
- `asg diag purge [Number_of_logs]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>list</td>
<td>Show the list of tests.</td>
</tr>
<tr>
<td>verify</td>
<td>Run tests and show a summary of the results.</td>
</tr>
<tr>
<td>print</td>
<td>Run tests and show the full output and also summary of the results.</td>
</tr>
<tr>
<td><code>&lt;Test1&gt;[,Test2,...]</code></td>
<td>Comma separated list of test IDs. To see a list of test IDs, run:</td>
</tr>
<tr>
<td></td>
<td>&gt; asg diag list.</td>
</tr>
<tr>
<td>purge</td>
<td>Delete the <code>asg diag</code> logs except for the newest.</td>
</tr>
<tr>
<td><code>&lt;Number_of_logs&gt;</code></td>
<td>The number of most recent logs to keep when <code>asg diag</code> log files. Default = 5.</td>
</tr>
</tbody>
</table>
### Showing the Tests

This example shows the complete list of diagnostic tests. The list shows the test ID, test name and the command that `asg diag` runs to show the specified test results.

```bash
> asg diag list
```

<table>
<thead>
<tr>
<th>ID</th>
<th>Title</th>
<th>Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System Health</td>
<td><code>asg stat -d</code></td>
</tr>
<tr>
<td>2</td>
<td>Hardware</td>
<td><code>asg hw_monitor -q</code></td>
</tr>
<tr>
<td>3</td>
<td>Resources</td>
<td><code>asg resource -q</code></td>
</tr>
<tr>
<td>4</td>
<td>Software Versions</td>
<td><code>asg_version verify -v</code></td>
</tr>
<tr>
<td>5</td>
<td>Software Provision</td>
<td><code>asg_provision -diag</code></td>
</tr>
<tr>
<td>6</td>
<td>CPU Type</td>
<td><code>cpu_socket_verifier -v</code></td>
</tr>
<tr>
<td>7</td>
<td>Media Details</td>
<td><code>transceiver_verifier -v</code></td>
</tr>
<tr>
<td>8</td>
<td>Chassis ID</td>
<td><code>verify_chassis_id</code></td>
</tr>
<tr>
<td>9</td>
<td>Distribution Mode</td>
<td><code>distutil verify -d</code></td>
</tr>
<tr>
<td>10</td>
<td>Policy</td>
<td><code>asg policy verify -a</code></td>
</tr>
<tr>
<td>11</td>
<td>AMW Policy</td>
<td><code>asg policy verify_amw -a</code></td>
</tr>
<tr>
<td>12</td>
<td>VSX Configuration</td>
<td><code>asg vsx_verify -v -a -c -i</code></td>
</tr>
<tr>
<td>13</td>
<td>Installation</td>
<td><code>installation_verify</code></td>
</tr>
<tr>
<td>14</td>
<td>Security Group</td>
<td><code>asg security_group diag</code></td>
</tr>
<tr>
<td>15</td>
<td>Cores Distribution</td>
<td><code>cores_verifier</code></td>
</tr>
<tr>
<td>16</td>
<td>SPI Affinity</td>
<td><code>spi_affinity_verifier -v</code></td>
</tr>
<tr>
<td>17</td>
<td>Clock</td>
<td><code>clock_verifier -v</code></td>
</tr>
<tr>
<td>18</td>
<td>Mgmt Monitor</td>
<td><code>mgmt_monitor snmp_verify -diag</code></td>
</tr>
<tr>
<td>19</td>
<td>Licenses</td>
<td><code>asg_license_verifier</code></td>
</tr>
<tr>
<td>20</td>
<td>Hide NAT range</td>
<td><code>asg_hide_behind_range -v</code></td>
</tr>
<tr>
<td>21</td>
<td>LTE</td>
<td><code>lte_verifier -v</code></td>
</tr>
<tr>
<td>22</td>
<td>IPS Enhancement</td>
<td><code>asg_ips_enhance status</code></td>
</tr>
<tr>
<td>23</td>
<td>MAC Setting</td>
<td><code>mac_verifier -v</code></td>
</tr>
<tr>
<td>24</td>
<td>ARP Consistency</td>
<td><code>asg_arp -v -q</code></td>
</tr>
<tr>
<td>25</td>
<td>Interfaces</td>
<td><code>interface_verifier -q</code></td>
</tr>
<tr>
<td>26</td>
<td>Bond</td>
<td><code>asg_bond -v -q</code></td>
</tr>
<tr>
<td>27</td>
<td>Bridge</td>
<td><code>asg_br_verifier -v</code></td>
</tr>
<tr>
<td>28</td>
<td>IPv4 Route</td>
<td><code>asg_route -q</code></td>
</tr>
<tr>
<td>29</td>
<td>IPv6 Route</td>
<td><code>asg_route -6 -q</code></td>
</tr>
<tr>
<td>30</td>
<td>Dynamic Routing</td>
<td><code>asg_dr_verifier</code></td>
</tr>
<tr>
<td>31</td>
<td>Local ARP</td>
<td><code>asg_local_arp_verifier -v</code></td>
</tr>
<tr>
<td>32</td>
<td>Port Speed</td>
<td><code>asg_port_speed verify</code></td>
</tr>
<tr>
<td>33</td>
<td>Core Dumps</td>
<td><code>core_dump_verifier -v</code></td>
</tr>
<tr>
<td>34</td>
<td>Syslog</td>
<td><code>asg_syslog verify</code></td>
</tr>
<tr>
<td>35</td>
<td>Processes</td>
<td><code>asg_process_verifier -v</code></td>
</tr>
</tbody>
</table>
### Running all Diagnostic Tests

This example shows the summary output for all diagnostic tests. When a test fails, the reasons for failure show in the Reason column.

```
> asg diag verify
Duration of tests vary and may take few seconds to complete
```

<table>
<thead>
<tr>
<th>Tests Status</th>
<th>ID</th>
<th>Title</th>
<th>Result</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>System Health</td>
<td>Failed</td>
<td>(1)Chassis 1 error</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>Hardware</td>
<td>Failed</td>
<td>(1)Chassis fan is down</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(2)Chassis fan exceeds threshold</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>Resources</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>Software Versions</td>
<td>Failed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>Software Provision</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>CPU Type</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>Media Details</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>Chassis ID</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>Distribution Mode</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Policy</td>
<td>Failed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11</td>
<td>AMW Policy</td>
<td>Failed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>VSX Configuration</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>13</td>
<td>Installation</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>Security Group</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>Cores Distribution</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>SPI Affinity</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>17</td>
<td>Clock</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>Mgmt Monitor</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>19</td>
<td>Licenses</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>Hide NAT range</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>21</td>
<td>LTE</td>
<td>Passed</td>
<td>(1)Not configured</td>
</tr>
<tr>
<td></td>
<td>22</td>
<td>IPS Enhancement</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>MAC Setting</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>ARP Consistency</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>Interfaces</td>
<td>Failed</td>
<td>(1)RX drop</td>
</tr>
<tr>
<td></td>
<td>26</td>
<td>Bond</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>27</td>
<td>Bridge</td>
<td>Passed</td>
<td>(1)Not configured</td>
</tr>
<tr>
<td></td>
<td>28</td>
<td>IPv4 Route</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>29</td>
<td>IPv6 Route</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>Dynamic Routing</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>31</td>
<td>Local ARP</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>Port Speed</td>
<td>Failed</td>
<td>(1)Inconsistent chassis configuration</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(2)Inconsistency between chassis and conf file</td>
</tr>
<tr>
<td></td>
<td>Misc</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Tests Summary

Passed: 26/35 tests
Run: "asg diag list 1,2,4,10,11,25,32,33,35" to view a complete list of failed tests

---

#### Showing Specified Diagnostic Tests

This example collects diagnostic information for specified tests.

```bash
> asg diag verify 1,2,3,4,5,30
```

### Tests Status

<table>
<thead>
<tr>
<th>ID</th>
<th>Title</th>
<th>Result</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System Health</td>
<td>Failed</td>
<td>(1)Chassis 1 error</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(2)Chassis 2 error</td>
</tr>
<tr>
<td>2</td>
<td>Hardware</td>
<td>Failed</td>
<td>(1)Chassis fan is down</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(2)Chassis fan exceeds threshold</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(3)CPU exceeds threshold</td>
</tr>
<tr>
<td>3</td>
<td>Resources</td>
<td>Passed</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Software Versions</td>
<td>Failed</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Software Provision</td>
<td>Passed</td>
<td></td>
</tr>
</tbody>
</table>

#### Networking

| 30  | Dynamic Routing           | Passed |                                             |

### Tests Summary

Passed: 3/6 tests
Run: "asg diag list 1,2,4" to view a complete list of failed tests
Output file: /var/log/verifier_sum.1-5.30.2014-02-17_10-56-05.txt
Troubleshooting Failures with asg diag

This example shows how to use the asg diag command for troubleshooting a failed diagnostic test. In this case, the test shows that two fans are down and the CPU temperature exceeds its threshold. The output identifies the failed components.

```plaintext
> asg diag verify 2

<table>
<thead>
<tr>
<th>Tests Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>-----</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tests Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passed: 0/1 test</td>
</tr>
<tr>
<td>Run: &quot;asg diag list 2&quot; to view a complete list of failed tests</td>
</tr>
</tbody>
</table>

> asg diag print 2

<table>
<thead>
<tr>
<th>Hardware Monitor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor</td>
</tr>
<tr>
<td>--------</td>
</tr>
<tr>
<td>Chassis 1</td>
</tr>
<tr>
<td>CMM bay 2</td>
</tr>
<tr>
<td>CPUtemp blade 1, CPU0</td>
</tr>
<tr>
<td>CPUtemp blade 1, CPU1</td>
</tr>
<tr>
<td>CPUtemp blade 2, CPU0</td>
</tr>
<tr>
<td>CPUtemp blade 2, CPU1</td>
</tr>
<tr>
<td>CPUtemp blade 3, CPU0</td>
</tr>
<tr>
<td>CPUtemp blade 3, CPU1</td>
</tr>
<tr>
<td>CPUtemp blade 4, CPU0</td>
</tr>
<tr>
<td>CPUtemp blade 4, CPU1</td>
</tr>
<tr>
<td>CPUtemp blade 5, CPU0</td>
</tr>
<tr>
<td>CPUtemp blade 5, CPU1</td>
</tr>
</tbody>
</table>

| Fan | bay 1, fan 1 | 0 | 11 | Speed Level | 0 |
| Fan | bay 1, fan 2 | 0 | 11 | Speed Level | 0 |

<table>
<thead>
<tr>
<th>PowerConsumption</th>
<th>Bay</th>
<th>Value</th>
<th>Threshold</th>
<th>Units</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>PowerUnit (AC) bay 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit (AC) bay 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit (AC) bay 3</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit (AC) bay 4</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PowerUnit (AC) bay 5</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 1, fan 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 1, fan 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 2, fan 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 2, fan 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 3, fan 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 3, fan 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 4, fan 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PowerUnit Fan bay 4, fan 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>PowerUnitFan</td>
<td>bay 5, fan 1</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
</tr>
<tr>
<td>PowerUnitFan</td>
<td>bay 5, fan 2</td>
<td>0</td>
<td>0</td>
<td>NA</td>
<td>0</td>
</tr>
<tr>
<td>SSM</td>
<td>bay 1</td>
<td>136</td>
<td>0</td>
<td>Mbps</td>
<td>1</td>
</tr>
<tr>
<td>SSM</td>
<td>bay 2</td>
<td>128</td>
<td>0</td>
<td>Mbps</td>
<td>1</td>
</tr>
<tr>
<td>Chassis 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMM</td>
<td>bay 1</td>
<td>1</td>
<td>0</td>
<td>&lt;S,D&gt;/&lt;A&gt;</td>
<td>1</td>
</tr>
<tr>
<td>CMM</td>
<td>bay 2</td>
<td>0</td>
<td>0</td>
<td>&lt;S,D&gt;/&lt;A&gt;</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 1, CPU0</td>
<td>50</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 1, CPU1</td>
<td>64</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 2, CPU0</td>
<td>48</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 2, CPU1</td>
<td>64</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 3, CPU0</td>
<td>48</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 3, CPU1</td>
<td>64</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 4, CPU0</td>
<td>47</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 4, CPU1</td>
<td>74</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 5, CPU0</td>
<td>84</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>CPUtemp</td>
<td>blade 5, CPU1</td>
<td>71</td>
<td>65</td>
<td>Celsius</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 1, fan 1</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 1, fan 2</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 2, fan 1</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 2, fan 2</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 3, fan 1</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
<tr>
<td>Fan</td>
<td>bay 3, fan 2</td>
<td>4</td>
<td>11</td>
<td>Speed Level</td>
<td>1</td>
</tr>
</tbody>
</table>

**Error Types**

This table includes some of the errors shown by `asg diag verify`.

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Error</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System health</td>
<td>Chassis &lt;X&gt; error</td>
<td>General error indicating that Chassis X grade is not perfect.</td>
</tr>
<tr>
<td>Hardware</td>
<td>&lt;Component&gt; is missing</td>
<td>The component is not found in the Chassis.</td>
</tr>
<tr>
<td></td>
<td>&lt;Component&gt; is down</td>
<td>The component is found in the Chassis but is inactive.</td>
</tr>
<tr>
<td>Resources</td>
<td>&lt;Resource&gt; capacity</td>
<td>The specified resource capacity is not as expected. Expected capacity can be tuned.</td>
</tr>
<tr>
<td></td>
<td>&lt;Resource&gt; exceed threshold</td>
<td>The resource’s usage exceeds the configured threshold.</td>
</tr>
<tr>
<td>CPU type</td>
<td>Non compliant CPU type</td>
<td>At least one SGM CPU type is not configured in the list of compliant CPUs. Compliant CPU types can be configured</td>
</tr>
<tr>
<td>Security group</td>
<td>&lt;Source&gt; error</td>
<td>The information gathered from this source is different between the SGMs.</td>
</tr>
<tr>
<td></td>
<td>&lt;Sources&gt; differ</td>
<td>The information gathered from several sources is different.</td>
</tr>
</tbody>
</table>
Changing Compliance Thresholds

You can change some compliance thresholds that define a healthy working system. To do this, edit the `asg_diag configuration file $FWDIR/conf/asg_diag_config` and change the threshold values.

These are the resources you can control:

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>RAM memory capacity in GB</td>
</tr>
<tr>
<td>HD: /</td>
<td>Disk capacity in GB for <code>&lt;disk&gt;</code>:/ partition.</td>
</tr>
<tr>
<td>HD:/var/log</td>
<td>Disk capacity in GB for the <code>/var/log</code> partition.</td>
</tr>
<tr>
<td>HD: /boot</td>
<td>Disk capacity in GB for the <code>/boot</code> partition.</td>
</tr>
<tr>
<td>Skew</td>
<td>The maximum permissible clock difference between the SGMs and SSMs, in seconds.</td>
</tr>
<tr>
<td>Certified cpu</td>
<td>Each line represents one compliant CPU type.</td>
</tr>
</tbody>
</table>

**Monitoring Hardware Components (asg hw_monitor)**

Use this command to show and monitor hardware information and thresholds for monitored components:

- Security Gateway Module - CPU temperature per socket
- Chassis fan speeds
- Security Switch Module - Throughput rates
- Power consumption per Chassis
- Power Supply Unit: Whether installed or not, and PSU fan speed
- Chassis Management Module - Installed, Active or Standby

**Syntax**

`asg hw_monitor [-v] [-f <filter>]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Show detailed component status report (verbose)</td>
</tr>
<tr>
<td>-f</td>
<td>Show status of one or more specified (filtered) components</td>
</tr>
<tr>
<td>&lt;filter&gt;</td>
<td>One or more of these component types, in a comma separated list: CMM CPUtemp Fan PowerConsumption PowerUnit PowerUnit SSM</td>
</tr>
</tbody>
</table>
Sample Output for the 61000 Security System

> asg hw_monitor -v

------------------------------------------------------------------------------
| Hardware Monitor |
------------------------------------------------------------------------------
| Sensor           | Location | Value | Threshold | Units       | State |
------------------------------------------------------------------------------
| Chassis 1
------------------------------------------------------------------------------
<p>| CMM              | bay 1    | 1     | 0         | &lt;S,D&gt;/&lt;A&gt;   | 1    |
| CMM              | bay 2    | 0     | 0         | &lt;S,D&gt;/&lt;A&gt;   | 1    |
| CPUtemp          | bay 1    | 45    | 65        | Celsius     | 1    |
| CPUtemp          | bay 2    | 39    | 65        | Celsius     | 1    |
| CPUtemp          | bay 3    | 44    | 65        | Celsius     | 1    |
| CPUtemp          | bay 4    | 47    | 65        | Celsius     | 1    |
| CPUtemp          | bay 5    | 0     | 65        | Celsius     | 1    |
| CPUtemp          | bay 6    | 0     | 65        | Celsius     | 1    |
| CPUtemp          | bay 7    | 0     | 65        | Celsius     | 1    |
| CPUtemp          | bay 8    | 0     | 65        | Celsius     | 1    |
| CPUtemp          | bay 9    | 0     | 65        | Celsius     | 1    |
| CPUtemp          | bay 10   | 0     | 65        | Celsius     | 1    |
| Fan              | bay 1    | 3     | 11        | Speed Level | 1    |
| Fan              | bay 2    | 3     | 11        | Speed Level | 1    |
| Fan              | bay 3    | 3     | 11        | Speed Level | 1    |
| Fan              | bay 4    | 3     | 11        | Speed Level | 1    |
| PowerConsumption | N/A      | 2711  | 4050      | Watts       | 1    |
| PowerUnit(AC)    | bay 1    | 0     | 0         | NA          | 1    |
| PowerUnit(AC)    | bay 2    | 0     | 0         | NA          | 1    |
| PowerUnit(AC)    | bay 3    | 0     | 0         | NA          | 1    |
| PowerUnit(AC)    | bay 4    | 0     | 0         | NA          | 1    |
| PowerUnit(AC)    | bay 5    | 0     | 0         | NA          | 1    |
| PowerUnitFan     | bay 1    | 0     | 0         | NA          | 1    |
| PowerUnitFan     | bay 2    | 0     | 0         | NA          | 1    |
| PowerUnitFan     | bay 3    | 0     | 0         | NA          | 1    |
| PowerUnitFan     | bay 4    | 0     | 0         | NA          | 1    |
| PowerUnitFan     | bay 5    | 0     | 0         | NA          | 1    |
| SSM              | bay 1    | 0     | 0         | Mbps        | 1    |
| SSM              | bay 2    | 0     | 0         | Mbps        | 1    |</p>
<table>
<thead>
<tr>
<th>Chassis 2</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CMM bay 1</td>
<td></td>
</tr>
<tr>
<td>CMM bay 2</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 1, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 2, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 2, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 3, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 3, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 4, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 4, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 5, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 5, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 6, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 6, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 7, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 7, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 8, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 8, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 9, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 9, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 10, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 10, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 11, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 11, CPU1</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 12, CPU0</td>
<td></td>
</tr>
<tr>
<td>CPUtemp blade 12, CPU1</td>
<td></td>
</tr>
<tr>
<td>Fan bay 1, fan 1</td>
<td></td>
</tr>
<tr>
<td>Fan bay 2, fan 1</td>
<td></td>
</tr>
<tr>
<td>PowerConsumption</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Sample Output for 41000 Security System

<table>
<thead>
<tr>
<th>Hardware Monitor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>Chassis 1</td>
</tr>
<tr>
<td>-------------------</td>
</tr>
<tr>
<td>CMM bay 1</td>
</tr>
<tr>
<td>CMM bay 2</td>
</tr>
<tr>
<td>CPUtemp blade 1, CPU0</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>PowerConsumption</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>SSM</td>
</tr>
<tr>
<td>SSM</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>Chassis 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMM</td>
</tr>
<tr>
<td>CMM</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>CPUtemp</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>Fan</td>
</tr>
<tr>
<td>PowerConsumption</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnit(AC)</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>PowerUnitFan</td>
</tr>
<tr>
<td>SSM</td>
</tr>
<tr>
<td>SSM</td>
</tr>
</tbody>
</table>

Notes

<table>
<thead>
<tr>
<th>Column</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>To identify the location, see the 61000/41000 Security System Front Panel.</td>
</tr>
<tr>
<td>Value Threshold</td>
<td>Most components have a defined threshold value. The threshold gives an indication of the health and functionality of the component. When the value of the resource is greater than the threshold, an alert is sent (&quot;Configuring Alerts for SGM and Chassis Events (asg alert)&quot; on page 53).</td>
</tr>
</tbody>
</table>
| State           | 0 = Component not installed  
1 = Component is installed |

Chassis Control (asg_chassis_ctrl)

The Chassis Control utility lets you monitor and configure SSMs and CMMs with many different command options and parameters. Chassis Control is based on SNMP communications between the different Chassis and components.

Note: You can configure SGMs using this utility, it is recommended to use the more comprehensive asg dxl command.

Syntax

asg_chassis_ctrl <option> <parameters>

Options and Parameters

<table>
<thead>
<tr>
<th>Options and Parameters</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>active_sgms dfsdf</td>
<td>Shows all installed SGMs.</td>
</tr>
<tr>
<td>active_ssm</td>
<td>Shows active SSMs. An SSM that is not installed or is down does not show as ACTIVE.</td>
</tr>
<tr>
<td>get_fans_status</td>
<td>Shows the health status of the Chassis fans.</td>
</tr>
<tr>
<td>get_lb_dist &lt;ssm_id&gt;</td>
<td>Shows the current distribution matrix from the specified SSM. The matrix is a table containing SGM IDs, and used to determine to which other SGMs a packet should be forwarded.</td>
</tr>
<tr>
<td>get_ssm_firmware &lt;ssm_id&gt;</td>
<td>Shows the firmware version of the specified SSM.</td>
</tr>
<tr>
<td>get_ssm_config &lt;ssm_id&gt;</td>
<td>Shows the configuration name of the specified SSM.</td>
</tr>
<tr>
<td>get_ssm_type &lt;ssm_id&gt;</td>
<td>Shows the model of the specified SSM.</td>
</tr>
<tr>
<td>Options and Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>get_psu_status</td>
<td>Shows the current status of the PSUs.</td>
</tr>
<tr>
<td>get_pems_status</td>
<td>Shows the current status of the Chassis PEMs.</td>
</tr>
<tr>
<td>get_cmm_status</td>
<td>Shows the current status of the CMMs.</td>
</tr>
<tr>
<td>get_cpus_temp &lt;sgm_id&gt;</td>
<td>Shows temperatures of the specified SGM CPUs.</td>
</tr>
<tr>
<td>get_dist_md5sum</td>
<td>Shows the md5sum of the distribution matrix for the given SSM. Comparing this checksum against the checksum on other SSM verifies that they are synchronized.</td>
</tr>
<tr>
<td>get_ports_stat &lt;ssm_id&gt;</td>
<td>Prints the port status for the specified SSM.</td>
</tr>
<tr>
<td>get_dist_mode &lt;ssm_id&gt;</td>
<td>Shows the port distribution mode for the specified SSM.</td>
</tr>
<tr>
<td>get_dist_mask &lt;ssm_id&gt;</td>
<td>Shows a summary of the distribution masks in the different modes.</td>
</tr>
<tr>
<td>get_matrix_size &lt;ssm_id&gt;</td>
<td>Shows the size, in bytes, of the SSM distribution matrix.</td>
</tr>
<tr>
<td>get_sel_info &lt;cmm_id&gt;</td>
<td>Shows data from the specified CMM event. This information is useful for troubleshooting and system forensics.</td>
</tr>
<tr>
<td>restart_ssm &lt;ssm_id&gt;</td>
<td>Restarts the specified SSM.</td>
</tr>
<tr>
<td>restart_cmm &lt;cmm_id&gt;</td>
<td>Restart the specified CMM.</td>
</tr>
<tr>
<td>start_ssm &lt;ssm_id&gt;</td>
<td>Starts the specified SSM.</td>
</tr>
<tr>
<td>shutdown_ssm &lt;cmm_id&gt;</td>
<td>Shuts down the specified SSM.</td>
</tr>
<tr>
<td>mib2_stats &lt;ssm_id&gt; &lt;port_id&gt; [&lt;err&gt;]</td>
<td>Shows MIB2 statistics for the specified SSM and port. &lt;err&gt; = Error type.</td>
</tr>
<tr>
<td>get_bmac &lt;ssm_id&gt;</td>
<td>Shows SGM MAC addresses from the SSM.</td>
</tr>
<tr>
<td>get_power_type</td>
<td>Shows the Chassis input power type (AC or DC).</td>
</tr>
<tr>
<td>get_ac_power_type</td>
<td>Shows the AC power type.</td>
</tr>
<tr>
<td>jumbo_frames enable</td>
<td>disable</td>
</tr>
<tr>
<td>set_port_mtu &lt;ssm_id&gt; &lt;port_id&gt; &lt;mtu_size&gt;</td>
<td>Sets the port MTU size for the specified SSM and Port. &lt;ssm_id&gt; - SSM identifier (1-4 or all) &lt;port_id&gt; - Port number &lt;mtu_size&gt; - This MTU size can be one of these values:</td>
</tr>
</tbody>
</table>
|                                             | • Integer value up to 12,288  
|                                             | • max - Maximum supported MTU size  
|                                             | • default - System default MTU size (typically 1544)                                                                                      |
| get_port_mtu <ssm_id> <port_id>             | Shows the MTU for the specified SSM and port.                                                                                            |
| get_port_media_details <ssm_id>             | Shows port information.                                                                                                                     |
Options and Parameters | Description
---|---
get.pem.cb.status | Shows PEM status.
help [-v] | Shows help messages in [-v] verbose mode

Notes
To see the syntax for an option, run the command and option without any parameters.

To make sure that the Chassis Control commands work correctly, run this command on both Chassis modules:

```
> asg_chassis_ctrl get_cmm_status
```

Getting CMM(s) status
CMM #1 -> Health: 1,  Active: 1
CMM #2 -> Health: 1,  Active: 0
Active CMM firmware version: 2.83

Security Monitoring

**SYN Defender (sim synatk, sim6 synatk, asg synatk)**

A SYN flood attack occurs when a host, typically with a forged address, sends a flood of TCP/SYN packets. Each of these packets is handled as a connection request, which causes the server to create a "half-open connection". This occurs because the gateway sends a TCP/SYN-ACK (Acknowledge) packet, and waits for a response packet, which never arrives. These half-open connections eventually exceed the maximum available connections, which causes a denial of service condition. SYN defender protects the gateway by dropping excessive half-open connections.

You can use these commands to:

- Configure a defense against an IPv4 SYN Flood attack. *(sim synatk)*
- Configure a defense against an IPv6 SYN Flood attack. *(sim6 synatk)*
- Monitor the system during attacks and normal system operation. *(asg synatk)*

This protection works with Performance Pack. SYN Defender disables templates, but does not turn off Performance Pack. This action can degrade Firewall performance.

**Syntax**

```
sim syntak [ -e ] [ -d ] [ -m ] [ -g ] [ -t <threshold> ] [ -a ] [ monitor ] [ monitor -v ]
sim6 syntak [ -e ] [ -d ] [ -m ] [ -g ] [ -t <threshold> ] [ -a ] [ monitor ] [ monitor -v ]
asg synatk [ -b <sgm_ids> ] [ -4 | -6 ]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-e</td>
<td>Enable SYN defender. This make the system engage when it recognizes an attack on an external interface. External interfaces are defined in SmartDashboard. Internal interfaces are always in monitor mode.</td>
</tr>
<tr>
<td>-d</td>
<td>Disable SYN Defender.</td>
</tr>
<tr>
<td>-mSYN</td>
<td>Set monitor mode. SYN defender only sends a log when it recognizes an attack.</td>
</tr>
<tr>
<td>-g</td>
<td>Enforce on all interfaces.</td>
</tr>
<tr>
<td>-t &lt;threshold&gt;</td>
<td>Set the SYN Defender threshold number of half-opened connections.</td>
</tr>
<tr>
<td>-a</td>
<td>Use configuration from $PPKDIR/conf/synatk.conf</td>
</tr>
<tr>
<td>monitor</td>
<td>Show the attack monitoring tool.</td>
</tr>
<tr>
<td>monitor -v</td>
<td>Show the attack monitoring tool with extra (verbose) information.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Show the status for specified SGMs and Chassis. Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>- No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>- One SGM</td>
</tr>
<tr>
<td></td>
<td>- A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>- A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>- One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>- The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-6</td>
<td>Shows the IPv6 status only.</td>
</tr>
<tr>
<td>-4</td>
<td>Shows the IPv4 status only.</td>
</tr>
</tbody>
</table>

---

### Monitoring a Syn Attack - Standard Output

This example shows that there are two interfaces under attack. Interface eth2-03 was attacked 3 seconds ago and eth2-04 is recovering from an attack that ended 24 seconds ago.

```
> asg synatk -b all -4
```

```
<table>
<thead>
<tr>
<th>SYN Defender status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration</td>
</tr>
<tr>
<td>----------------------</td>
</tr>
<tr>
<td>Configuration</td>
</tr>
<tr>
<td>IF</td>
</tr>
<tr>
<td>----------------------</td>
</tr>
<tr>
<td>eth1-Mgmt4</td>
</tr>
<tr>
<td>eth1-01</td>
</tr>
<tr>
<td>eth2-01</td>
</tr>
<tr>
<td>eth2-02</td>
</tr>
<tr>
<td>eth2-03</td>
</tr>
<tr>
<td>eth2-04</td>
</tr>
</tbody>
</table>

---

**Output information**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF</td>
<td>Interface name.</td>
</tr>
<tr>
<td>Topology</td>
<td>Topology as defined in SmartDashboard.</td>
</tr>
<tr>
<td>Enforce</td>
<td>Action taken by SYN Defender:</td>
</tr>
<tr>
<td></td>
<td><strong>Prevent</strong> - Detects attacks and enforces protection.</td>
</tr>
<tr>
<td></td>
<td><strong>Detect</strong> - Detects attacks, but only generates log entries. Does not enforce protection.</td>
</tr>
<tr>
<td></td>
<td><strong>Disabled</strong> - Protection is disabled.</td>
</tr>
<tr>
<td>Column</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>State</td>
<td>Current Syn Defender state:</td>
</tr>
<tr>
<td></td>
<td><strong>Disabled</strong> - Syn Defender is disabled for this interface.</td>
</tr>
<tr>
<td></td>
<td><strong>Monitor</strong> - The gateway is not under attack and Syn Defender monitors</td>
</tr>
<tr>
<td></td>
<td>connections.</td>
</tr>
<tr>
<td></td>
<td><strong>Active</strong> - The gateway is under attack and Syn Defender enforces</td>
</tr>
<tr>
<td></td>
<td>protections.</td>
</tr>
<tr>
<td></td>
<td><strong>Grace</strong> - The gateway An attack has ended and the normal service is</td>
</tr>
<tr>
<td></td>
<td>restored.</td>
</tr>
<tr>
<td>non-established conns</td>
<td><strong>Peak</strong> - The highest number of half-opened connections for this interface.</td>
</tr>
<tr>
<td></td>
<td>This can help you to configure the correct threshold.</td>
</tr>
<tr>
<td></td>
<td><strong>Current</strong> - The number of half-opened connections at this time.</td>
</tr>
</tbody>
</table>

### Monitoring a SYN Attack - Verbose Output

This example shows the verbose output.

```plaintext
> sim synatk monitor -v
+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+
| SYN Defender statistics                  | Status: Under Attack (!)                | Spoofed SYN/sec                          | 534000                                    |
+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+
| IF | Topology | Defend (sec) | SYN cookie rate                          |
| eth2-01 | External | 28 | 345345 | 40 | 95 % |
| eth2-02 | External | 12 | 150 | 50 | 33 % |
+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+
| Sum | 345495 | 90 | 93 % |
+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+-------------------------------------------+
```

### Output Description

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IF</td>
<td>The interface name</td>
</tr>
<tr>
<td>Topology</td>
<td>The interface topology as defined in SmartDashboard.</td>
</tr>
<tr>
<td>Defend</td>
<td>The attack duration in seconds.</td>
</tr>
<tr>
<td>Sent SYN cookie rate</td>
<td>Number of SYN packets received per second.</td>
</tr>
<tr>
<td>BAU</td>
<td>Business as usual. The number of legitimate connections handled per second.</td>
</tr>
<tr>
<td>Spoofed</td>
<td>The percentage of spoofed SYN packets out of all traffic.</td>
</tr>
</tbody>
</table>
Showing Syn Defender Status

This example shows the status of SYN Flood attack protection for all SGMs. It shows that blade 1-01 is under attack, and there are 3 half-open connections.

```
> asg synatk
+---------------------------------------------------------------+
| SYN Defender status                                          |
+---------------------------------------------------------------+
| Blade  | IP   | Config   | Threshold | Status       | Non est. conns |
+---------------------------------------------------------------+
| 1_01 (!) | IPv4 | Enforcing | 5000      | Under Attack | 3              |
| 1_01     | IPv6 | Enforcing | 5000      | Normal       | 0              |
| 1_02     | IPv4 | Enforcing | 5000      | Normal       | 0              |
| 1_02     | IPv6 | Enforcing | 5000      | Normal       | 0              |
+---------------------------------------------------------------+
```

F2F Quota (asg f2fq, fwaccel f2fg stats)

Use these commands to show details of an F2F (Forward to Firewall) DDoS flood attack, and how the protection works to mitigate it. F2F detects traffic floods and intelligently prevents performance degradation on the 61000/41000 Security System. It assigns a high priority to known, important packets from Performance Pack and drops those suspected of being part of a DDoS attack.

Two examples of known F2F flood attacks are UDP floods and fragmentation attacks. These attacks cause excessive resource allocation when they try to put the packet fragments together.

**Syntax**

```
fwaccel f2fg stats [-v]
asg f2fq [-b <sgm_ids>] [-6 | -4]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Shows detailed (verbose) statistics.</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>One SGM</td>
</tr>
<tr>
<td></td>
<td>A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-6</td>
<td>Shows the IPv6 status only</td>
</tr>
<tr>
<td>-4</td>
<td>Shows the IPv4 status only</td>
</tr>
</tbody>
</table>
Example - fwaccel f25
This example shows details of activity for all Firewall instances.

```bash
> fwaccel f2fq stats -v
```

<table>
<thead>
<tr>
<th>DDOS Mitigation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mode:</td>
</tr>
<tr>
<td>Status</td>
</tr>
<tr>
<td>Last 10 seconds drops</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Instance</th>
<th>Reason</th>
<th>Drops / Hits</th>
</tr>
</thead>
<tbody>
<tr>
<td>FW 0</td>
<td>CONN_MISS_TCP_SYN</td>
<td>103365 / 104629</td>
</tr>
<tr>
<td>FW 1</td>
<td>FRAG</td>
<td>6232 / 13816</td>
</tr>
<tr>
<td></td>
<td>CONN_MISS_TCP_SYN</td>
<td>101096 / 102203</td>
</tr>
<tr>
<td></td>
<td>CONN_MISS_TCP_OTHER</td>
<td>13146 / 14359</td>
</tr>
<tr>
<td>FW 2</td>
<td>FRAG</td>
<td>1339 / 1339</td>
</tr>
<tr>
<td></td>
<td>CONN_MISS_TCP_SYN</td>
<td>101087 / 102143</td>
</tr>
<tr>
<td>All</td>
<td>FRAG</td>
<td>7571 / 15155</td>
</tr>
<tr>
<td></td>
<td>CONN_MISS_TCP_SYN</td>
<td>305548 / 308975</td>
</tr>
<tr>
<td></td>
<td>CONN_MISS_TCP_OTHER</td>
<td>13146 / 14359</td>
</tr>
</tbody>
</table>

The output shows this information:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Last 10 seconds drops</td>
<td>The number of dropped packets during the last 10 seconds.</td>
</tr>
<tr>
<td>Instance</td>
<td>The verbose output shows a historical aggregate of the results, for each Firewall instance.</td>
</tr>
<tr>
<td>Drops / Hits</td>
<td>The number of dropped packets out of the total number of packets, grouped by the attack type.</td>
</tr>
</tbody>
</table>

Example - asg f2fq
This output shows how the protection mitigates the DDoS attack, per SGM.

```bash
> asg f2fq
```

<table>
<thead>
<tr>
<th>Blade</th>
<th>Protocol</th>
<th>Config</th>
<th>Status</th>
<th>Last 10 sec drops</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_01</td>
<td>IPv4</td>
<td>Enforcing</td>
<td>Under Attack</td>
<td>151130</td>
</tr>
<tr>
<td>1_01</td>
<td>IPv6</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_02</td>
<td>IPv4</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_02</td>
<td>IPv6</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_03</td>
<td>IPv4</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_03</td>
<td>IPv6</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_04</td>
<td>IPv4</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
<tr>
<td>1_04</td>
<td>IPv6</td>
<td>Enforcing</td>
<td>Normal</td>
<td>0</td>
</tr>
</tbody>
</table>
Showing the Number of Firewall and SecureXL Connections (asg_conns)

Use this command to show the number of firewall and SecureXL connections on each SGM.

Syntax

asg_conns [-b <sgm_ids>]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| <sgm_ids> | Works with SGMs and/or Chassis as specified by <sgm_ids>. The <sgm_ids> can be:  
|           | No <sgm_ids> specified or all shows all SGMs and Chassis  
|           | One SGM  
|           | A comma-separated list of SGMs (1_1,1_4)  
|           | A range of SGMs (1_1-1_4)  
|           | One Chassis (Chassis1 or Chassis2)  
|           | The active Chassis (chassis_active) |
| -6        | Show only IPv6 connections |
| -h        | Show syntax and help information |
Example

```bash
> asg_conns
_01:
   #VALS     #PEAK   #SLINKS
      246      1143       246
_02:
   #VALS     #PEAK   #SLINKS
       45       172        45
_03:
   #VALS     #PEAK   #SLINKS
       45      212        45
_04:
   #VALS     #PEAK   #SLINKS
      223       624       223
_05:
   #VALS     #PEAK   #SLINKS
       45       246        45
```

Total (fwl connections table): 604 connections

_01:
There are 60 conn entries in SecureXL connections table
Total conn entries @ DB 0:  4
Total conn entries @ DB 3:  2
.
.
Total conn entries @ DB 26:  4
Total conn entries @ DB 30:  2
_02:
There are 16 conn entries in SecureXL connections table
Total conn entries @ DB 0:  2
Total conn entries @ DB 1:  2
.
.
Total conn entries @ DB 26:  2
_03:
There are 16 conn entries in SecureXL connections table
Total conn entries @ DB 0:  2
Total conn entries @ DB 5:  2
.
.
Total conn entries @ DB 30:  2
_04:
There are 260 conn entries in SecureXL connections table
Total conn entries @ DB 0: 10
Total conn entries @ DB 1:  6
.
.
Total conn entries @ DB 31:  94
_05:
There are 16 conn entries in SecureXL connections table
Total conn entries @ DB 2:  2
.
.
Total conn entries @ DB 26:  2
```

Total (SecureXL connections table): 368 connections
**Packet drop monitoring (HLINK_1)**

Use this command in the Expert mode to monitor dropped packets in real time. Drop statistics are taken from these modules:

- NICs
- Operating system
- CoreXL
- PSL
- Performance Pack

This command opens a monitor session and shows aggregated data from SGMs and, optionally, SSMs. To stop an open session, press Ctrl-c.

**Syntax**

`asg_drop_monitor [-r] [-ssm[-t timeout]] [-6]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-r</code></td>
<td>Reset statistics to 0</td>
</tr>
<tr>
<td><code>-ssm</code></td>
<td>Include dropped packets from SSMs</td>
</tr>
<tr>
<td><code>-t</code></td>
<td>Change the default</td>
</tr>
<tr>
<td><code>-6</code></td>
<td>Show only IPv6 results</td>
</tr>
<tr>
<td><code>-h</code></td>
<td>Show command syntax and help information</td>
</tr>
</tbody>
</table>

**Output**

NICs drops (Rx):
0
IP Stack qdisc drops (Tx):
0
CoreXL queue drops (F2F):
0
CoreXL queue drops (PXL F2P)
0
PSL drops (total):
0
PSL drops (udp):
0
PSL rejects:
0
Ppak drops:

Displaying aggregated data from blades: all

<table>
<thead>
<tr>
<th>Reason</th>
<th>Value</th>
<th>Reason</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>general reason</td>
<td>0</td>
<td>PXL decision</td>
<td>0</td>
</tr>
<tr>
<td>fragment error</td>
<td>0</td>
<td>hl - spoof viol</td>
<td>0</td>
</tr>
<tr>
<td>F2F not allowed</td>
<td>0</td>
<td>hl - TCP viol</td>
<td>0</td>
</tr>
<tr>
<td>corrupted packet</td>
<td>0</td>
<td>hl - new conn</td>
<td>0</td>
</tr>
<tr>
<td>clr pkt on vpn</td>
<td>0</td>
<td>partial conn</td>
<td>0</td>
</tr>
<tr>
<td>encrypt failed</td>
<td>0</td>
<td>drop template</td>
<td>0</td>
</tr>
<tr>
<td>decrypt failed</td>
<td>0</td>
<td>outb - no conn</td>
<td>9</td>
</tr>
<tr>
<td>interface down</td>
<td>0</td>
<td>cluster error</td>
<td>0</td>
</tr>
<tr>
<td>XMT error</td>
<td>0</td>
<td>template quota</td>
<td>0</td>
</tr>
<tr>
<td>anti spoofing</td>
<td>0</td>
<td>Attack mitigation</td>
<td>0</td>
</tr>
<tr>
<td>local spoofing</td>
<td>0</td>
<td>sanity error</td>
<td>0</td>
</tr>
<tr>
<td>monitored spoofed</td>
<td>0</td>
<td>Conns limit. Exceed</td>
<td>0</td>
</tr>
<tr>
<td>Conns limit. Add fail</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Other Monitoring Commands

**Showing System Serial Numbers**

These commands show and save serial numbers for 61000/41000 Security System hardware components:

- `asg_sgm_serial` - Shows SGM serial numbers only.
- `asg_serial_info` - Shows CMM, SSM and Chassis serial numbers.

The information is saved in the `gasginfo` archive file.

Run these commands in the Expert mode. This command shows serial numbers from SGMs in UP state that belong to the security group.

**Syntax**

```
asg_sgm_serial [-a]
asg_serial_info [-a]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-a</td>
<td>Apply command on all SGMs in the security group</td>
</tr>
</tbody>
</table>

**Examples**

```
# asg_sgm_serial
1_01:
  Board Serial          : AKO0769153
1_02:
  Board Serial          : AKO0585533
2_01:
  Board Serial          : AKO0462069
2_02:
  Board Serial          : AKO0447878

# asg_serial_info
  chassis 1 CMM1 serial: 1163978/005
  chassis 1 CMM2 serial: 1157482/001
  chassis 1 SSM1 serial: 0011140011
  chassis 1 SSM2 serial: 0011140012
  chassis 1 serial: 1159584/016
  chassis 2 CMM1 serial: 1163090/041
  chassis 2 CMM2 serial: 1155519/014
  chassis 2 SSM1 serial: 0311310621
  chassis 2 SSM2 serial: 0311310626
  chassis 2 serial: 0831232/001
```

**Notes**

To show CMM, SSM and Chassis serial numbers, one of the SGMs on each Chassis must be up and running. For example, if no UP SGM is found on Chassis-2, the serial numbers for components for all components in the Chassis will not show or be saved.

**Showing the 61000/41000 Security System Version (ver)**


To see which version is installed on a 61000/41000 Security System, run:
For 61000 Security System:
> ver
1_01:
Product version Check Point 61000 R76
OS build 106
OS kernel version 2.6.18-92cpx86_64
OS edition 64-bit

for 41000 Security System:
> ver
1_04:
Product version Check Point Gaia 41000 R76
OS build 105
OS kernel version 2.6.18-92cpx86_64
OS edition 64-bit

Looking a Log Files (asg log)
Use this command to see the contents of the specified log file.

Syntax
asg log [-b <sgm_ids>] <log_name> [-tail [<n>]] [-f <filter>]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>• No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>• One SGM</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>• A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>• One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>• The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>&lt;log_name&gt;</td>
<td>Enter the log file to show:</td>
</tr>
<tr>
<td></td>
<td>• audit</td>
</tr>
<tr>
<td></td>
<td>Shows the audit logs in /var/log</td>
</tr>
<tr>
<td></td>
<td>For example: /var/log/asgaudit.log.1</td>
</tr>
<tr>
<td></td>
<td>• smd</td>
</tr>
<tr>
<td></td>
<td>Shows the System Monitor Daemon logs in /var/log</td>
</tr>
<tr>
<td></td>
<td>For example: /var/log/sdm.log.2</td>
</tr>
<tr>
<td></td>
<td>• ports</td>
</tr>
<tr>
<td></td>
<td>Shows the ports logs in /var/log</td>
</tr>
<tr>
<td></td>
<td>For example: /var/log/ports</td>
</tr>
<tr>
<td></td>
<td>• dist_mode</td>
</tr>
<tr>
<td></td>
<td>Shows the logs for distribution mode activity.</td>
</tr>
<tr>
<td>-tail [&lt;n&gt;]</td>
<td>Show only last n lines of the log file for each SGM. For example, -tail 3 shows only the last three lines of the specified log file. Default = 10 lines.</td>
</tr>
<tr>
<td>-f filter</td>
<td>Word or phrase use as a filter. For example, -f debug</td>
</tr>
</tbody>
</table>

Example - Audit logs
> asg log audit
Feb 02 17:36:12 1_01 WARNING: Blade admin up on blades:
1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:16:17 1_01 WARNING: Blade admin down on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:17:40 1_01 WARNING: Blade admin up on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:19:53 1_01 WARNING: Blade admin down on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:22:31 1_01 WARNING: Blade admin up on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:23:30 1_01 WARNING: Reboot on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 08:38:16 1_01 WARNING: Reboot on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 09:21:09 1_01 WARNING: Reboot on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 11:07:08 1_01 WARNING: Reboot on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y
Feb 03 11:33:10 1_01 WARNING: Reset sic on blades: all, User: y, Reason: y
Feb 03 11:50:08 1_01 WARNING: Reset sic on blades: all, User: y, Reason: y
Feb 03 13:32:32 1_01 WARNING: Reset sic on blades: all, User: y, Reason: y
Feb 03 14:30:26 1_01 WARNING: Reset sic on blades: all, User: johndoe, Reason: test
Feb 03 14:48:03 1_01 WARNING: Reset sic on blades: all, User: johndoe, Reason: test
Feb 03 15:34:11 1_01 WARNING: Reset sic on blades: all, User: y, Reason: y
Feb 03 17:55:23 1_01 WARNING: Reboot on blades: 1_02,1_03,1_04,1_05,2_01,2_02,2_03,2_04,2_05, User: y, Reason: y

Example - Port logs (last 12 lines)
> asg log ports-tail 12
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-09 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-10 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-11 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-12 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-13 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-14 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-15 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-16 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-Mgmt1 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-Mgmt2 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-Mgmt3 link is down
Feb 3 18:01:40 2_05 Athens-ch02-05 cmd: Chassis 2 eth2-Mgmt4 link is down

Example - Using a filter
> asg log -b 1_01,1_04 dist_mode -f bridge
Feb 2 18:10:30 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-bridges = 4
Feb 2 18:10:30 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-vsbridges = 4
Feb 2 18:12:31 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-bridges = 4
Feb 2 18:12:31 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-vsbridges = 4
Feb 2 18:14:14 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-bridges = 4
Feb 2 18:14:14 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-vsbridges = 4
Feb 2 18:14:30 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-bridges = 4
Feb 2 18:14:30 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-vsbridges = 4
Feb 2 18:16:19 1_01 Athens-ch01-01 distutil:0: initialize_environment: vs-ids-bridges = 4

Looking at the Auditlog File (asg_auditlog)

Use the asg_auditlog command to see the contents of the auditlog file. This log file contains an entry for each change made to the SGM configuration database with gclish or other commands. The auditlog file for each SGM is located in the /var/log directory.

The asg_auditlog command collects and summarizes records from the SGMs. The output shows actions that occur on different SGMs within n seconds (default = 5) on one line. These are considered to be global actions applicable to all SGMS. You can change the number of seconds for this purpose.

The log contains two types of activities:

Permanent - The action permanently changes the configuration database on the SGM hard disk.

Transient - The action changes the configuration database in SGM memory, which does not survive reboot.
Syntax

`auditlog [-b <sgm_ids>] [-d <n>] [-tail [number]] [-f filter]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-b &lt;sgm_ids&gt;</code></td>
<td>Works with SGMs and/or Chassis as specified by <code>&lt;sgm_ids&gt;</code>.</td>
</tr>
<tr>
<td></td>
<td>The <code>&lt;sgm_ids&gt;</code> can be:</td>
</tr>
<tr>
<td></td>
<td>- No <code>&lt;sgm_ids&gt;</code> specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>- One SGM</td>
</tr>
<tr>
<td></td>
<td>- A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>- A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>- One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>- The active Chassis (chassis_active)</td>
</tr>
<tr>
<td><code>-d &lt;n&gt;</code></td>
<td>Number of seconds between the same actions that occur on different SGMs, which show on one output line. Default = 5 seconds.</td>
</tr>
<tr>
<td><code>-tail &lt;n&gt;</code></td>
<td>Show only last <code>n</code> lines of the log file for each SGM. For example, <code>-tail 3</code> shows only the last three lines of the specified log file. Default = 10 lines.</td>
</tr>
<tr>
<td><code>-f &lt;filter&gt;</code></td>
<td>Word or phrase to use as an output filter. For example, <code>-f t</code> shows only transient changes.</td>
</tr>
</tbody>
</table>

Example - Show last lines

This example shows the last five activities, in this case, cpstop actions.

```
> asg_auditlog -tail 5
```

```
Feb 3 05:30:49 admin localhost p -command:cpstop t [1 Blades: 1_03]
Feb 3 05:30:49 admin localhost p -command:cpstop:description Stop\ Check\ Point\ products\ installed [1 Blades: 1_03]
Feb 3 05:30:49 admin localhost p +command:cpstop:description Global\ extension\ for\ cpstop 1 Blades: 1_03]
Feb 3 05:30:49 admin localhost p -command:cpstop:description Global\ extension\ for\ cpstop 1 Blades: 1_03]
Feb 3 05:30:49 admin localhost p +command:cpstop:path /bin/cpstop_start [1 Blades: 1_03]
```

Notes:

`p +` = Permanent action that added or changed an item in the configuration database.

`p -` = Permanent action that deleted an item in the configuration database

`t +` = Transient action that added or changed an item in the configuration database in memory only.

`t -` = Transient action that deleted an item in the configuration database in memory only.

Example - filter

This example shows only permanent configuration save actions.

```
> asg_auditlog -f p +configurationSave
```

```
Feb 3 15:21:31 admin localhost p +configurationSave t [2 Blades: 1_01,1_02]
Feb 3 15:21:31 admin localhost p +configurationSave t [2 Blades: 1_03,1_04]
Feb 3 15:22:03 admin localhost p +configurationSave t [3 Blades: 1_01,1_02,2_02]
Feb 3 15:22:08 admin localhost p +configurationSave t [4 Blades: 2_01,2_03,2_04,2_05]
Feb 3 15:24:23 admin localhost p +configurationSave t [2 Blades: 1_03,1_04]
Feb 3 15:24:24 admin localhost p +configurationSave t [2 Blades: 1_03,1_04]
Feb 3 15:24:29 admin localhost p +configurationSave t [5 Blades: 1_03,1_04,2_03,2_04,
Feb 3 15:24:30 admin localhost p +configurationSave t [4 Blades: 2_01,2_03,2_04,2_05]
Feb 3 15:24:35 admin localhost p +configurationSave t [2 Blades: 2_01,2_02]
Feb 3 15:24:36 admin localhost p +configurationSave t [1 Blades: 2_02]
Feb 3 15:24:44 admin localhost p +configurationSave t [2 Blades: 2_01,2_03]
Feb 3 15:24:51 admin localhost p +configurationSave t [2 Blades: 2_02,2_04]
Feb 3 15:24:56 admin localhost p +configurationSave t [1 Blades: 2_05]
```
Working with the firewall Database Configuration (asg config)

Use this command to show the current firewall database configuration. You can also save the current configuration to a file. The output and saved file include configuration information for all SGMs. The asg config command is useful to:

- Copy the firewall configuration to a different system. For example, if you deploy a new 61000/41000 Security System, you can use the saved configuration from an existing 61000/41000 Security System to quickly get up and running.
- Quickly re-configure a system that was reverted to factory defaults. Before reverting to the factory default image, save the existing configuration then use it to override the factory settings.

Syntax

```
asg config show|save [-t] [<path>] <file>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>Show the existing database configuration</td>
</tr>
<tr>
<td>save</td>
<td>Saves the current configuration to a file</td>
</tr>
<tr>
<td>-t</td>
<td>Add a timestamp to the file name. (save only)</td>
</tr>
<tr>
<td>&lt;file&gt;</td>
<td>Name and path of the saved configuration file. If you do not enter a path, the configuration is saved to /home/admin.</td>
</tr>
</tbody>
</table>

Example

```
$ asg config save -t myconfig
```

This example saves the current configuration to /home/admin/myconfig.

Showing Software and Firmware versions (asg_version)

Description

You can use the asg_version command to:

- Retrieve system configuration
- Retrieve software versions:
  - Check Point software (Firewall and Performance Pack versions)
  - Firmware versions for SGMs, SSMs, and CMMs
- Make sure that system hardware components are running approved software and firmware versions

Syntax

```
asg_version -h
asg_version verify
asg_version [-v] [-i] [-b <sgm_ids>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Show complete command syntax</td>
</tr>
<tr>
<td>verify</td>
<td>Makes sure that system hardware components run approved software and firmware versions</td>
</tr>
<tr>
<td>-i</td>
<td>Show active and standby SGMs</td>
</tr>
</tbody>
</table>
-b <sgm_ids>  Works with SGMs and/or Chassis as specified by <sgm_ids>.

The <sgm_ids> can be:

- No <sgm_ids> specified or all shows all SGMs and Chassis
- One SGM
- A comma-separated list of SGMs (1_1,1_4)
- A range of SGMs (1_1-1_4)
- One Chassis (Chassis1 or Chassis2)
- The active Chassis (chassis_active)

**Showing a List of Two SGMs**

```
> asg_version 1_01,1_03
SGMs =========

--- 2 SGMs: 1_01 1_03 ---
OS build 42, OS kernel version 2.6.18-92cpx86_64, OS edition 64-bit

Hardware
-------
**-- 1 blade: 1_01 --**
BIOS: 1.30 BL: 1.52 IPMC: 1.52 FPGA: 2.40 FPGARE: 2.40
**-- 1 blade: 1_03 --**
BIOS: 0.54 BL: 1.42 IPMC: 1.42 FPGA: 2.38 FPGARE: 2.38

OS version
-------
BIOS: 0.54 BL: 1.42 IPMC: 1.42 FPGA: 2.38 FPGARE: 2.
```
Showing Verbose Mode

> asg_version -v

+------------------------------------------------------------------------+
| Hardware Versions                                                     |
+------------------------------------------------------------------------+
| Component | Type     | Configuration | Firmware |
+-----------+-----------+--------------+----------+
| Chassis 2 |           |              |          |
+-----------+-----------+--------------+----------+
| SSM1      | SSM160    | N/A          | 2.4.C7   |
| SSM2      | N/A       | N/A          | N/A      |
| CMM       | N/A       | N/A          | 2.83     |
+-----------+-----------+--------------+----------+

SGMs
-----
Type
-----
**-- 2 blades: 2_02 2_03 --**
SGM220

OS version
---------
**-- 2 blades: 2_02 2_03 --**
OS build 80, OS kernel version 2.6.18-92cpx86_64, OS edition 64-bit

FireWall-1 version
------------------
**-- 2 blades: 2_02 2_03 --**
This is Check Point VPN-1(TM) & FireWall-1(R) 61000_R76 - Build 083
kernel: 61000_R76 - Build 083

Performance Pack version
-------------------------
**-- 2 blades: 2_02 2_03 --**
This is Check Point Performance Pack version: 61000_R76 - Build 083
Kernel version: 61000_R76 - Build 083

Hardware
--------
**-- 1 blade: 2_02 --**
BIOS: 1.30 BL: 1.42 IPMC: 1.52 FPGA: 2.40 FPGARE: 2.40
**-- 1 blade: 2_03 --**
BIOS: 1.30 BL: 1.52 IPMC: 1.54 FPGA: 2.40 FPGARE: 2.40

SSD
---
-- 1 blade: 2_02 --
Firmware Version: 2CV102M3
-- 1 blade: 2_03 --
Firmware Version: 4PC10362

Number of cores
----------------
-- 1 blade: 2_02 --
8
-- 1 blade: 2_03 --
12

Number of CoreXL instances
--------------------------
-- 2 blades: 2_02 2_03 --
4
CPUs frequency
-------------
- 1 blade: 2_02 -*
  2.13GHz
- 1 blade: 2_03 -*
  2.4GHz

**Showing System Messages (asg_varlog)**

Use this command to show system messages written to message files stored in the `/var/log` directory on SGMs. The output shows in chronological order. Each line shows the SGM that created the log entry.

**Syntax**

```
asg_varlog [-b <sgm-ids>] [-tail <number>] [-f <filter>]
```

```
asg_varlog -h
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-b &lt;sgm_ids&gt;</code></td>
<td>The SGMs from which to collect <code>/var/log/messages</code>. Works with SGMs and/or Chassis as specified by <code>&lt;sgm_ids&gt;</code>. The <code>&lt;sgm_ids&gt;</code> can be:</td>
</tr>
<tr>
<td></td>
<td>• No <code>&lt;sgm_ids&gt;</code> specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>• One SGM</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>• A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>• One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>• The active Chassis (chassis_active)</td>
</tr>
<tr>
<td><code>-tail &lt;number&gt;</code></td>
<td>Show only last n lines of the log file for each SGM. For example, <code>-tail 3</code> shows only the last three lines of the specified log file. Default = 10 lines.</td>
</tr>
<tr>
<td><code>-f &lt;filter&gt;</code></td>
<td>Word or phrase to use as an output filter. For example, <code>-f ospf</code> shows only OSPF messages.</td>
</tr>
<tr>
<td><code>-h</code></td>
<td>Shows command syntax and help information.</td>
</tr>
</tbody>
</table>

**Example**

This example shows messages on Chassis1 containing the word ‘restarted’.

```
> asg_varlog -b chassis1 -f Restarted
Feb 5 12:40:07 1_03 Athens-ch01-03 pm[8465]: Restarted /bin/routed[8489], count=1
Feb 5 12:40:09 1_04 Athens-ch01-04 pm[8449]: Restarted /bin/routed[9995], count=1
Feb 5 12:40:09 1_04 Athens-ch01-04 pm[8449]: Restarted /opt/CPsuite-R76/fw1/bin/cmd[11291], count=1
Feb 5 12:40:09 1_04 Athens-ch01-04 pm[8449]: Restarted /opt/CPsuite-R76/fw1/bin/cmd[11291], count=1
Feb 5 12:40:11 1_04 Athens-ch01-04 pm[8449]: Restarted /usr/libexec/gexecd[11328], count=2
Feb 5 12:40:11 1_04 Athens-ch01-04 pm[8449]: Restarted /usr/libexec/gexecd[11328], count=2
Feb 5 12:40:11 1_04 Athens-ch01-04 pm[8449]: Restarted /usr/libexec/gexecd[11328], count=2
Feb 5 12:40:11 1_04 Athens-ch01-04 pm[8449]: Restarted /usr/libexec/gexecd[11328], count=2
```

**Monitoring the System with SNMP**

You can use SNMP to monitor various aspects of the 61000/41000 Security System, including:

- Software versions
- Hardware status
- Key performance indicators
- Chassis high availability status
To monitor the system using SNMP

1. Upload the MIB to your third-party SNMP monitoring software.
   The SNMP MIB is located on each SGM under: $CPDIR/lib/snmp/chkpnt.mib
   For monitoring the 61000/41000 Security System, the only supported OIDs are under
   iso.org.dod.internet.private.enterprise.checkpoint.products.asg (OID 1.3.6.1.4.1.2620.1.48)
2. Enable the SNMP agent on the 61000/41000 Security System.
   In gclish, run:
   > set snmp agent on

SNMP Traps

The 61000/41000 Security System supports this SNMP trap only:
iso.org.dod.internet.private.enterprise.checkpoint.products.asgTrap
(OID 1.3.6.1.4.1.2620.1.2001)

The SNMP traps MIB is located on each SGM under: $CPDIR/lib/snmp/chkpnt-trap.mib

Note - The set snmp traps command is not supported. You must use the asg alert
configuration wizard for this purpose.

To learn more about SNMP, see Configuring asg alerts ("Configuring Alerts for SGM and Chassis Events
(asg alert)" on page 53).

SNMP in a VSX Gateway

There are two SNMP modes for a 61000/41000 Security System configured as a VSX Gateway:

Default Mode - Monitor global SNMP data from the 61000/41000 Security System. Data is
accumulated from all SGMs for all Virtual System.

Virtual Systems Mode  Monitor each Virtual System separately.

Note - SNMP traps are supported for VS0 only.

Supported SNMP Versions

The SNMP Virtual Systems mode uses SNMP version 3 to query the Virtual Systems. You can run remote
SNMP queries on each Virtual System in the VSX Gateway.

For systems that only support SNMP versions 1 and 2:

- You cannot run remote SNMP queries for each Virtual System. You can only run a remote SNMP query
  on VS0.
- You can use gclish to change the Virtual System context and then run a local SNMP query on it.

Enabling the SNMP Virtual System Mode

To use SNMP Per Virtual Systems:

1. Run this command to configure an SNMP V3 user:
   > add snmp usm user jon security-level authNoPriv authpass-phrase VALUE
2. Run one of these commands to set the SNMP mode:
   > set snmp mode vs
   or
   > set snmp mode default
3. To start SNMP agent, run:
   > set snmp agent on
To see Virtual System throughput from a Linux host:

```
# snmpwalk -m /CPDIR/lib/snmp/chkpnt.mib -n ctxname_vsid1 -v 3 -l authNoPriv -u jon -A mypassword 192.0.2.72 asgThroughput
```

To query Virtual System throughput, from its context:

1. Go to the expert mode.
2. To change to the applicable Virtual System, run:
   ```
   > vsenv <vs_ids>
   ```
3. Run:
   ```
   # snmpwalk -m /CPDIR/lib/snmp/chkpnt.mib -v 2c -c public localhost asgThroughput
   ```

Common SNMP MIBs

This table shows common SNMP MIBs that are applicable to the 61000/41000 Security System.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>OID</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Throughput</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.1&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Connection Rate (cps)</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.2&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Packet Rate (pps)</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.3&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Concurrent conn.</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.4&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Accelerated cps</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.6&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System non-accelerated cps</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.7&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Accelerated Concurrent conn.</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.8&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Non-accelerated Concurrent conn.</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.9&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System CPU load AVG.</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.10&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System Acceleration CPU load AVG</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.11&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System FW instances load AVG</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.14&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>System VPN Throughput</td>
<td>String</td>
<td>1.3.6.1.4.1.2620.1.48.17&lt;IPver_index&gt;</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>OID</td>
<td>Comments</td>
</tr>
<tr>
<td>-------------------------------------------</td>
<td>------</td>
<td>-----------------------------------------------</td>
<td>-----------------------------------------------</td>
</tr>
<tr>
<td>System Path distribution (fast, medium, slow, drops).</td>
<td>Table</td>
<td>1.3.6.1.4.1.2620.1.48.&lt;IPver_index&gt;.24</td>
<td>Path Distribution of: Throughput PPS CPS Concurrent conn.</td>
</tr>
<tr>
<td>Per SGM counters</td>
<td>Table</td>
<td>1.3.6.1.4.1.2620.1.48.&lt;IPver_index&gt;.25</td>
<td>Counters of: Throughput cps pps concurrent conn sxl CPU usage (avg / min/max) fw CPU usage (avg/min/max)</td>
</tr>
<tr>
<td>Performance peaks</td>
<td>Table</td>
<td>1.3.6.1.4.1.2620.1.48.&lt;IPver_index&gt;.26</td>
<td></td>
</tr>
<tr>
<td>Sensors Per Chassis</td>
<td>Table</td>
<td>1.3.6.1.4.1.2620.1.48.22.1.1</td>
<td>Status Details of: Fans SSMs CPU temp CMM PSUs PSUs Fans</td>
</tr>
<tr>
<td>Resources Per SGM</td>
<td>Table</td>
<td>1.3.6.1.4.1.2620.1.48.23</td>
<td>Memory and HD utilization.</td>
</tr>
</tbody>
</table>

**Note:**

<IPver_index>= 20 for IPv4 or 21 for IPv6.
Monitoring Virtual Systems (cpha_vsx_util monitor)

Use this command to stop or start Virtual System (VS) monitoring.

The state of an SGM is not affected by unmonitored Virtual Systems. For example, an unmonitored Virtual Systems in problem state (pnote) is ignored, and the SGM state does change to Down.

Not monitoring a Virtual Systems is useful if you want an SGM to be UP even if a specific Virtual Systems is Down or does not have a Policy (for example, after running unload local).

Syntax

cpha_vsx_util monitor start|stop <vs_ids>
cpha_vsx_util monitor show

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>Show all unmonitored Virtual Systems</td>
</tr>
<tr>
<td>stop</td>
<td>Stop monitoring the Virtual Systems</td>
</tr>
<tr>
<td>start</td>
<td>Start monitoring the Virtual Systems.</td>
</tr>
<tr>
<td>&lt;vs_ids&gt;</td>
<td>One or more Virtual Systems in one of the following formats &quot;1&quot; or &quot;1,3,4-6&quot; or &quot;1-3&quot;</td>
</tr>
</tbody>
</table>

Note

When you stop Virtual System monitoring, you must run Virtual Systems cpha_vsx_util monitor start to start it again. Monitoring does not start automatically after reboot.
Chapter 2

System Configuration

In This Section:

- Administration .......................................................................................................................... 88
- Synchronize SGM Time (asg_ntp_sync_config) ......................................................................... 100
- Configuring SGMs (asg_blade_config) .................................................................................... 101
- Backing Up and Restoring an SGM (backup_system) ............................................................... 102
- Configuring SGM state (asg_sgm_admin) .................................................................................. 104
- Image Management .................................................................................................................. 105
- High Availability ..................................................................................................................... 107
- Monitoring, Logs and Auditing ................................................................................................ 115
- Port Mirroring (SPAN Port) ...................................................................................................... 123
- Security ..................................................................................................................................... 126
- VSX Provisioning ..................................................................................................................... 130

Administration

Working with Global Commands

The 61000/41000 Security System operating system includes a set of global commands that apply to all or specified SGMS in a system.

gclish commands apply globally to all SGMs by default. Some gclish commands are applicable to the 61000/41000 Security System and its components.

gclish commands do not apply to SGMs in the DOWN state. If you run a set command while a SGM is down, the command will not update that SGM. The SGM synchronizes its database during the startup process and the changes are applied after reboot.

clish commands are documented in Gaia Admin Guide. Most of these commands are also available in the 61000/41000 Security System.

Notes

- Documentation for the Chassis feature is in the Hardware Monitoring and Chassis High Availability ("Chassis High Availability Active/Standby Mode" on page 107) sections.
- auditlog is enabled by default. All commands are recorded in the log and can be retrieved with asg_auditlog (documented separately).
- config-lock is the command that protects gclish database. The lock can be held by single SGM per system. When user attempts to perform gclish set operations from specific SGM, he should make sure that this SGM holds the config-lock. In order to acquire config-lock, the command set config-lock on override should be executed.
- gclish traffic runs on Sync interface, port 1129/TCP.
- gclish can run extended commands. Run show commands extended to see the list of extended commands, which can run from gclish.
- To run command on specified SGMs, use the blade-range specification. When you use blade-range, all gclish embedded commands will run only on this subset of SGMs. Since all SGMs must have identical configuration, the use of blade-range is not recommended.
**Check Point global commands**

**Description**

The global commands are scripts that run commands on more than one SGM. This section includes Check Point product-related commands, such as `fw`, `sim`, `fwaccel`, and `cpconfig`.

- The general global command syntax is shown in "OS global commands" document
- The list of available commands is: `sim`, `sim6`, `fwaccel`, `fwaccel6`, `fw`, `fw6`, `cpconfig`
- Those commands are available in gclish and in the Expert mod if you add the "g_" prefix.
- Other relevant documents may include "OS global commands" and "General commands".

**fwaccel, fwaccel6**

These commands let you dynamically enable or disable acceleration for IPv4 traffic while the 61000/41000 Security System is in operation. `fwaccel6` has the same functionality as `fwaccel`, but for IPv6 traffic. This setting goes back to the default value after reboot.

When you run these commands from gclish, `fwaccel/fwaccel6` are, for most parameters, comparison global commands that show combined information from all SGMs. "fwaccel stats" and "fwaccel notifstats" commands show aggregated statistics from all SGMs

**Syntax**

```
fwaccel {on|off|stat|stats [-s] [-d] |conns [-s] -m <max_entries> |templates [-s] -m <max_entries>}
fwaccel {on|off|stat|stats [-s] [-d] |conns [-s] -m <max_entries> |templates [-s] -m <max_entries>}
```

**Parameter** | **Description**
--- | ---
-b | Works with SGMs and/or Chassis as specified by `<sgm_ids>`. The `<sgm_ids>` can be:
| No `<sgm_ids>` specified or all shows all SGMs and Chassis
| One SGM
| A comma-separated list of SGMs (1_1,1_4)
| A range of SGMs (1_1-1_4)
| One Chassis (Chassis1 or Chassis2)
| The active Chassis (chassisActive)

**Note**: You can only select SGMs from one Chassis with this option.

on | Starts acceleration
off | Stops acceleration
stat | Shows the acceleration device status and the status of the Connection Templates on the local Security Gateway.
stats | Shows acceleration statistics.
stats -s | Shows more summarized statistics.
stats -d | Shows dropped packet statistics.
conns | Shows all connections.
conns -s | Shows the number of connections defined in the accelerator.
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>conns -m</code></td>
<td>Limits the number of connections displayed by the <code>conns</code> command to the number entered in the variable <code>max_entries</code>.</td>
</tr>
<tr>
<td><code>templates</code></td>
<td>Shows all connection templates.</td>
</tr>
<tr>
<td><code>templates -m</code></td>
<td>Limits the number of templates displayed by the <code>templates</code> command to the number entered in the variable <code>max_entries</code>.</td>
</tr>
<tr>
<td><code>templates -s</code></td>
<td>Shows the number of templates currently defined in the accelerator.</td>
</tr>
</tbody>
</table>

**Example**

```bash
> fwaccel stats
Displaying aggregated data from blades: all
```

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>-----------------------</td>
<td>-------------</td>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Accelerated Path</strong></td>
<td></td>
<td><strong>Accelerated Path</strong></td>
<td></td>
</tr>
<tr>
<td>accel packets</td>
<td>6518</td>
<td>accel bytes</td>
<td>870476</td>
</tr>
<tr>
<td>conns created</td>
<td>38848</td>
<td>conns deleted</td>
<td>38043</td>
</tr>
<tr>
<td>C total conns</td>
<td>801</td>
<td>C templates</td>
<td>0</td>
</tr>
<tr>
<td>C TCP conns</td>
<td>493</td>
<td>C delayed TCP conns</td>
<td>0</td>
</tr>
<tr>
<td>C non TCP conns</td>
<td>308</td>
<td>C delayed nonTCP con</td>
<td>0</td>
</tr>
<tr>
<td>conns from templates</td>
<td>0</td>
<td>temporary conns</td>
<td>0</td>
</tr>
<tr>
<td>nat conns</td>
<td>0</td>
<td>C nat conns</td>
<td>0</td>
</tr>
<tr>
<td>dropped packets</td>
<td>0</td>
<td>dropped bytes</td>
<td>0</td>
</tr>
<tr>
<td>nat templates</td>
<td>0</td>
<td>port alloc templates</td>
<td>0</td>
</tr>
<tr>
<td>conns from nat tmpl</td>
<td>0</td>
<td>port alloc conns</td>
<td>0</td>
</tr>
<tr>
<td>Policy deleted tmpl</td>
<td>0</td>
<td>C Policy deleted tmp</td>
<td>0</td>
</tr>
<tr>
<td><strong>Accelerated VPN Path</strong></td>
<td></td>
<td><strong>Accelerated VPN Path</strong></td>
<td></td>
</tr>
<tr>
<td>C crypt conns</td>
<td>0</td>
<td>enc bytes</td>
<td>0</td>
</tr>
<tr>
<td>dec bytes</td>
<td>0</td>
<td>ESP enc pkts</td>
<td>0</td>
</tr>
<tr>
<td>ESP enc err</td>
<td>0</td>
<td>ESP dec pkts</td>
<td>0</td>
</tr>
<tr>
<td>ESP dec err</td>
<td>0</td>
<td>ESP other err</td>
<td>0</td>
</tr>
<tr>
<td>AH enc pkts</td>
<td>0</td>
<td>AH enc err</td>
<td>0</td>
</tr>
<tr>
<td>AH dec pkts</td>
<td>0</td>
<td>AH dec err</td>
<td>0</td>
</tr>
<tr>
<td>AH other err</td>
<td>0</td>
<td>espudp enc pkts</td>
<td>0</td>
</tr>
<tr>
<td>espudp enc err</td>
<td>0</td>
<td>espudp dec pkts</td>
<td>0</td>
</tr>
<tr>
<td>espudp dec err</td>
<td>0</td>
<td>espudp other err</td>
<td>0</td>
</tr>
<tr>
<td><strong>Medium Path</strong></td>
<td></td>
<td><strong>Medium Path</strong></td>
<td></td>
</tr>
<tr>
<td>PXL packets</td>
<td>0</td>
<td>PXL async packets</td>
<td>0</td>
</tr>
<tr>
<td>PXL bytes</td>
<td>0</td>
<td>PXL conns</td>
<td>0</td>
</tr>
<tr>
<td>C PXL conns</td>
<td>0</td>
<td>C PXL templates</td>
<td>0</td>
</tr>
<tr>
<td><strong>Firewall Path</strong></td>
<td></td>
<td><strong>Firewall Path</strong></td>
<td></td>
</tr>
<tr>
<td>F2F packets</td>
<td>10077862</td>
<td>F2F bytes</td>
<td>1185051123</td>
</tr>
<tr>
<td>F2F conns</td>
<td>38839</td>
<td>C F2F conns</td>
<td>800</td>
</tr>
<tr>
<td>TCP violations</td>
<td>0</td>
<td>C partial conns</td>
<td>0</td>
</tr>
<tr>
<td>C anticipated conns</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>General</strong></td>
<td></td>
<td><strong>General</strong></td>
<td></td>
</tr>
<tr>
<td>memory used</td>
<td>0</td>
<td>free memory</td>
<td>0</td>
</tr>
</tbody>
</table>

(*) Statistics marked with C refer to current value, others refer to total value
**Monitor Mode**

fwaccel_m continuously monitors fwaccel output in real, which is useful to show acceleration statistics in real time. When you run this command, the screen goes into the monitor mode and shows changes in parameters as highlighted text. You cannot run commands or do other operations while in the Monitor mode.

To close the Monitor mode and continue working with the command line, press Ctrl-c.

**Example**

> fwaccel_m stats -p

Displaying aggregated data from blades: all

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>accel packets</td>
<td>0</td>
<td>accel bytes</td>
<td>0</td>
</tr>
<tr>
<td>conns created</td>
<td>25799</td>
<td>conns deleted</td>
<td>24687</td>
</tr>
<tr>
<td>C total conns</td>
<td>11900</td>
<td>C templates</td>
<td>0</td>
</tr>
<tr>
<td>C TCP conns</td>
<td>838</td>
<td>C delayed TCP conns</td>
<td>0</td>
</tr>
<tr>
<td>C non TCP conns</td>
<td>248</td>
<td>C delayed nonTCP con</td>
<td>0</td>
</tr>
<tr>
<td>conns from templates</td>
<td>0</td>
<td>temporary conns</td>
<td>0</td>
</tr>
<tr>
<td>nat conns</td>
<td>0</td>
<td>C nat conns</td>
<td>0</td>
</tr>
<tr>
<td>dropped packets</td>
<td>0</td>
<td>dropped bytes</td>
<td>0</td>
</tr>
<tr>
<td>nat templates</td>
<td>0</td>
<td>port alloc templates</td>
<td>0</td>
</tr>
<tr>
<td>conns from nat tmpl</td>
<td>0</td>
<td>port alloc conns</td>
<td>0</td>
</tr>
<tr>
<td>Policy deleted tmpl</td>
<td>0</td>
<td>C Policy deleted tmp</td>
<td>0</td>
</tr>
<tr>
<td>conns auto expired</td>
<td>0</td>
<td>conns reused</td>
<td>0</td>
</tr>
</tbody>
</table>

**Accelerated VPN Path**

| C crypt conns     | 0         | enc bytes         | 0         |
| dec bytes         | 0         | ESP enc pkts      | 0         |
| ESP enc err       | 0         | ESP dec pkts      | 0         |
| ESP dec err       | 0         | ESP other err     | 0         |
| AH enc pkts       | 0         | AH enc err        | 0         |
| AH dec pkts       | 0         | AH dec err        | 0         |
| AH other err      | 0         | espudp enc pkts   | 0         |
| espudp enc err    | 0         | espudp dec pkts   | 0         |
| espudp dec err    | 0         | espudp other err  | 0         |

**Medium Path**

| FXL packets       | 0         | FXL async packets | 0         |
| FXL bytes         | 0         | FXL conns         | 0         |
| C FXL conns       | 0         | C FXL templates   | 0         |

**Firewall Path**

| F2F packets       | 13963723  | F2F bytes         | 15620824161 |
| F2F conns         | 0         | C F2F conns       | 1103        |
| TCP violations     | 0         | C partial conns   | 0           |
| port alloc f2f     | 0         |                   | 0           |
fw, fw6

When run fw/fw6 commands are global scripts that run the fw/fw6 command on each SGM.

Example 1
fw ctl

Output
> fw ctl
--* 6 blades: 1_01 1_02 1_03 2_01 2_02 2_03 --*

Usage:
fw ctl command args...
Commands: install, uninstall, pstat, iflist, arp, debug, kdebug, bench, chain, conn

Example 2
fw ctl iflist

Output
gdual7-t43-ch02-02 > fw ctl iflist
--* 6 blades: 1_01 1_02 1_03 2_01 2_02 2_03 --*
0 : BPEth0
1 : BPEth1
2 : eth1-Mgmt4
3 : eth2-Mgmt4
4 : eth1-01
5 : eth1-CIN
6 : eth2-CIN
8 : eth2-01
16 : Sync
17 : eth1-Mgmt1
18 : eth2-Mgmt1

fw dbgfile

Use this command for debugging of the system.

fw dbgfile collect collects firewall debugging information (fw ctl debug).
User needs to stop its collection manually - by writing stop.
fw dbgfile view shows the collected debugging information

Syntax
fw [gexec-flags] dbgfile [collect|view] [fw ctl debug options]

Example 1
> fw dbgfile collect -f /home/admin/temp.dbg -buf 2300 -m kiss + pmdump -m fw + xlate

Notes
Debug collection: fw dbgfile collect [-buf BUF_SIZE] -f FILE [FLAGS]
FILE - file to collect the debug information to, full path should be provided
FLAGS - debug flags

Example 2
> fw dbgfile view /home/admin/temp.dbg

Notes
Debug viewing: fw dbgfile view FILE
FILE - file containing debug information collected by the collect option, full path should be provided.
Global Operating System Commands

Global operating system commands are standard Linux commands that run on all or specified SGMs. When you run a global command in the gclish shell, the operating system runs a global script, which the standard Linux command on the SGMs. When you run a command in the Expert mode, it works as a standard Linux command. To use the global command in the Expert mode, run the global command script version as shown in this table:

<table>
<thead>
<tr>
<th>gclish Command</th>
<th>Global Command - Expert Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>arp</td>
<td>g_arp</td>
</tr>
<tr>
<td>cat</td>
<td>g_cat</td>
</tr>
<tr>
<td>cp</td>
<td>g_cp</td>
</tr>
<tr>
<td>dmesg</td>
<td>g_dmesg</td>
</tr>
<tr>
<td>ethtool</td>
<td>g_ethtool</td>
</tr>
<tr>
<td>ls</td>
<td>g_ls</td>
</tr>
<tr>
<td>md5sum</td>
<td>g_md5sum</td>
</tr>
<tr>
<td>Mv</td>
<td>g_mv</td>
</tr>
<tr>
<td>Netstat</td>
<td>g_netstat</td>
</tr>
<tr>
<td>Reboot</td>
<td>g_reboot</td>
</tr>
<tr>
<td>tail</td>
<td>g_tail</td>
</tr>
<tr>
<td>tcpdump</td>
<td>g_tcpdump</td>
</tr>
<tr>
<td>ifconfig</td>
<td>g_ifconfig</td>
</tr>
<tr>
<td>top</td>
<td>g_top</td>
</tr>
</tbody>
</table>

The parameters and options for the standard Linux command are available for the global command. In addition, you can use the `-b` parameter to select some or all SGMs for the global command.

Syntax

```
{<gclish_command> | <global_command>} [-b <sgm_ids>] <command_options>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| `-b <sgm_ids>` | Works with SGMs and/or Chassis as specified by `<sgm_ids>`. The `<sgm_ids>` can be:  
  • No `<sgm_ids>` specified or all shows all SGMs on the Active Chassis  
  • One SGM  
  • A comma-separated list of SGMs (1_1,1_4)  
  • A range of SGMs (1_1-1_4)  
  • One Chassis (Chassis1 or Chassis2)  
  • The active Chassis (chassis_active)  

  Note: You can only select SGMs from one Chassis with this option.  

<table>
<thead>
<tr>
<th><code>&lt;gclish_command&gt;</code></th>
<th>In the gclish shell, enter the standard command</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;global_command&gt;</code></td>
<td>In the Expert mode, enter the global command as shown in the table</td>
</tr>
<tr>
<td><code>&lt;command_options&gt;</code></td>
<td>Enter the standard command options for the specified command.</td>
</tr>
</tbody>
</table>

One or more flags may be specified, however the `-l` and `-r` flags should not be specified together.
Global arp

This example shows the interfaces on all SGMs

```plaintext
> arp
1_01:
Address   HWtype  HWaddress           Flags Mask   Iface
192.0.2.2  ether   00:1C:7F:02:04:FE   C            Sync
172.23.9.28 ether 00:14:22:09:D2:22   C            eth1-Mgmt4
192.0.2.3  ether   00:1C:7F:03:04:FE   C            Sync
1_02:
Address   HWtype  HWaddress           Flags Mask   Iface
192.0.2.3  ether   00:1C:7F:03:04:FE   C            Sync
172.23.9.28 ether 00:14:22:09:D2:22   C            eth1-Mgmt4
192.0.2.1  ether   00:1C:7F:01:04:FE   C            Sync
1_03:
Address   HWtype  HWaddress           Flags Mask   Iface
192.0.2.1  ether   00:1C:7F:01:04:FE   C            Sync
172.23.9.28 ether 00:14:22:09:D2:22   C            eth1-Mgmt4
192.0.2.2  ether   00:1C:7F:02:04:FE   C            Sync
```

Global ls

This example runs the ls command from the Expert mode on SGMs 1_1, 1_2, and 1_3. The output shows the combined results for these SGMs.

```plaintext
# g_ls ls
-- 4 blades: 1_01 1_02 1_03
CPbackup  ace  crash  lib  log  opt  run  suroot
CPsnapshot cache  empty  lock  mail  preserve  spool  tmp
```

Global top

The global top command shows SGM processor activity in real time. The default output also shows a list of the most processor-intensive processes. In addition to the standard functionality of the Linux top command, global top adds these features for the 61000/41000 Security System:

- The global top relies on the user configuration for the local top utility; The global command will use the local SGM configuration file for configuring the output on the remote SGMs.

```plaintext
> top [local] [-f [ -o filename] [-n niter] | -s <filename> | -h] [global command-flags] [top cmd line args]
```

How to manage g_top display

Top uses a configuration file to manage output display; top by default will copy and use this configuration file from the local blade (usually located under ~/.toprc). This file will be copied to all SGMs and will be used when calling top.

To manage g_top display:
1. Run local top (from shell) and set the desired display view
2. Save configuration (shift+w)
3. Run global top

local mode

It is also possible for each blade to display output using its own local configuration file simply run "top local"

How to send output to a file

At times, it is more convenient to send g_top output to a file, for example, when there are more SGMs then the screen can handle. To enable the file mode use the -f flag.

Output file

In file mode the output top will be sent to a file (default: /var/log/gtop.<time>). Use --o flag to specify a different file to save in.
Number of iterations
By default top will perform one iteration in file mode, use --n to specify a different number

Showing output file
Use top --s <filename> to show the content of file <filename>.

Global Commands Generated by CMM

Description
The CMM monitors and controls Chassis components. It can turn on and off SGMs and SSMs.
Users can turn on and turn SGMs in serious circumstances, such as when a SGM is not accessible with the
Sync interface. In this case, the reboot command does not work.

These are the commands that control SGM power from CMM:
- asg_reboot <global command-flags> – Restart SGMs
- asg_hard_shutdown <global command-flags> – Turn off SGMs
- asg_hard_start <global command-flags> -- Turn on SGMs

To learn more about <global commands-flags>, see the OS Global commands section. You can run global
commands from gclish and the expert shells.

Example
> asg_reboot -b 1_03,2_05
You are about to perform hard reboot on SGMs: 1_03,2_05
It might cause performance hit for a period of time
Are you sure? (Y - yes, any other key - no) Y

Hard reboot requires auditing
Enter your full name: User1
Enter reason for hard reboot [Maintenance]:
WARNING: Hard reboot on SGMs: 1_03,2_05, User: User1, Reason: Maintenance

Rebooting SGMs: 1_03,2_05

Notes
- To run these commands for SGMs on a remote Chassis, at least one SGM must be UP and running on
  the remote Chassis.
- To learn how to restart an SSM from the CMM, see the asg_chassis_ctrl section.
General global commands

Description

The global commands are utilities that run certain commands on more than one SGM. This document is dealing with general purpose utilities.

The global commands syntax is shown in "OS global commands" document

The list of available commands is: update_conf_file, global, asg_cp2blades, asg_clear_table, asg_clear_messages, asg_blade_stats

Those commands are available in the gclish in addition they are available in bash:

<table>
<thead>
<tr>
<th>Gclish Name</th>
<th>Bash Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>update_conf_file</td>
<td>g_update_conf_file</td>
</tr>
<tr>
<td>global</td>
<td>global_help</td>
</tr>
<tr>
<td>asg_cp2blades</td>
<td>asg_cp2blades</td>
</tr>
<tr>
<td>asg_clear_table</td>
<td>asg_clear_table</td>
</tr>
<tr>
<td>asg_clear_messages</td>
<td>asg_clear_messages</td>
</tr>
<tr>
<td>asg_blade_stats</td>
<td>asg_blade_stats</td>
</tr>
</tbody>
</table>

Other relevant documents may include "OS global commands" and "CP global commands".

update_conf_file

Usage: update_conf_file <file_name> <var>=<value>

Description: update_conf_file is a utility to add, update and remove variables from configuration files (configuration file format is specified below)

Input parameters:

file-name - Name/Path of .conf file to update. In case of known conf files full path is not required known conf files are: fwkern.conf, simkern.conf

var - Variable name

value - New value. An empty value will remove the variable from the .conf file (yet "=" sign must be specified)

Example

> cat /home/admin/MyConfFile.txt

```
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3 blades: 2_01 2_02 2_03</td>
<td></td>
</tr>
</tbody>
</table>
```

> cat: /home/admin/MyConfFile.txt: No such file or directory

> update_conf_file /home/admin/MyConfFile.txt var1=hello

> cat /home/admin/MyConfFile.txt

```
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3 blades: 2_01 2_02 2_03</td>
<td></td>
</tr>
</tbody>
</table>
```

> var1=hello

> update_conf_file /home/admin/MyConfFile.txt var2=24h
> cat /home/admin/MyConfFile.txt
-^ 3 blades: 2_01 2_02 2_03 -^-
var2=24h
var1=hello

> update_conf_file /home/admin/MyConfFile.txt var1=goodbye
> cat /home/admin/MyConfFile.txt
-^ 3 blades: 2_01 2_02 2_03 -^-
var2=24h
var1=goodbye

> update_conf_file /home/admin/MyConfFile.txt var2=
> cat /home/admin/MyConfFile.txt
-^ 3 blades: 2_01 2_02 2_03 -^-
var1=goodbye

Configuration file required format:
The configuration file is composed of lines of variable initialization where each line defines one variable
Line format is: <variable>=<value>
Variable name must not include "=" sign
Note: fwkern.conf and simkern.conf are aligned with this definition

global help
Usage: global help
Description: shows the list of global commands accessible through gclish and their general usage
Example:
> global help
Usage: <command_name> [-b SGMs] [-a -l -r --] <native command arguments>
Executes the specified command on specified blades.

Optional Arguments:
-b blades: in one of the following formats
1_1,1_4 or 1_1-1_4 or 1_01,1_03-1_08,1_10
all (default)
chassis1
chassis2
chassis_active
-a : Force execution on all SGMs (incl. down SGMs).
-l : Execute only on local blade.
-r : Execute only on remote SGMs.

Command list:
arp cat cp cpconfig cplic cpstart cpstop dmesg ethtool fw fw6 fwaccel fwaccel6
fwaccel6_m fwaccel_m ls md5sum mv netstat reboot sim sim6 snapshot_recover
snapshot_show_current tail tcqpdump top unlock update_conf_file vpn asg
asg_cp2blades
usage: asg_cp2blades [global command-flags] [-s] file-name-full-path [destination-full-path]
Description: this utility copies files from the current SGM to any specified SGMs

Input parameters:
Global command flags – the global flags which specify on which SGMs to be applied on
-s - flag that specify whether to save a local copy of the old file on each of the selected SGMs. The saved copy will reside on the same directory as the original file and will end with .bak.<date>.<time>
file-name-full-path – full path to the file to be copied. If full-path is not specified the file will be searched in current directory.
destination-full-path – full path to a destination location for the file. If destination was not specified, the file will be copied to the source file location

description: 

- example:
gcpmodule-ch02-01 > cat /home/admin/note.txt
- 1 blade: 2_01
hello world
- 2 blades: 2_02 2_03

- example:
gcmodule-ch02-01 > asg_cp2blades /home/admin/note.txt
Operation completed successfully
gcmodule-ch02-01 > cat /home/admin/note.txt
- 3 blades: 2_01 2_02 2_03
hello world

asg_clear_table
usage: asg_clear_table [global command-flags]
Description: clears firewall connection table. This function will delete connections from fw connection table. Its success indication is having less than 50 connections; it will repeat delete process for up to 15 times until meeting this threshold.
Note: if connected to the machine by SSH, this command will delete current connection and user will need to re-establish the connection

- Example:
gcpmodule-ch02-01 > asg_clear_table
This action will erase the messages in /var/log/messages and will be executed on blades: all
Are you sure? (Y - yes, any other key - no) y
Command completed successfully

asg_clear_messages
usage: asg_clear_messages [global command-flags]
Description: clears all messages in /var/log/messages files

- Example:
gcpmodule-ch02-01 > asg_clear_messages
This action will erase the messages in /var/log/messages and will be executed on blades: all
Are you sure? (Y - yes, any other key - no) y
Command completed successfully
> asg varlog
Dec 5 16:33:07 2_01 cpmodule-ch02-01 clish[30185]: cmd by admin: asg varlog
gcpmodule-ch02-01 >

Example
> show interface eth1-01 ipv4-address
1_01:
ipv4-address 4.4.4.10/24

1_02:
ipv4-address 4.4.4.10/24

1_03:
ipv4-address 4.4.4.10/24

1_04:
ipv4-address 4.4.4.10/24

1_05:
Blade 1_05 is down. See "/var/log/messages".

2_01:
ipv4-address 4.4.4.10/24

2_02:
ipv4-address 4.4.4.10/24

2_03:
ipv4-address 4.4.4.10/24

2_04:
ipv4-address 4.4.4.10/24

2_05:
ipv4-address 4.4.4.10/24

Configuring Chassis state (asg chassis_admin -c)
Use this command to put a Chassis in the administrative UP or DOWN state. You must have administrator permissions to do this.

When a Chassis is in the Administrative DOWN state:
- Backup connections for SGMs are lost
- New connections are not synchronized with the Down Chassis.

Syntax
> asg chassis_admin -c <chassis_id> down|up

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;chassis_id&gt;</td>
<td>Chassis identification number (1 or 2)</td>
</tr>
<tr>
<td>down</td>
<td>up</td>
</tr>
</tbody>
</table>
Example

> asg chassis_admin
You are about to perform Chassis_admin down on Chassis: 2
Are you sure? (Y - yes, any other key - no) y
Chassis_admin down requires auditing
Enter your full name: John
Enter reason for chassis_admin down [Maintenance]: test
WARNING: Chassis_admin down on Chassis: 2, User: John, Reason: test
Chassis 2 is going DOWN...
Chassis 2 state is DOWN

Notes

- This command is audited. (asg log audit)
- Run this command to see the Chassis state:
  > asg stat/monitor

Note - In a Dual Chassis environment, a Chassis in the administrative DOWN causes degradation of the system performance.

Synchronize SGM Time (asg_ntp_sync_config)

Description
Use the `asg_ntp_sync_config` command to synchronize the time for all SGMs and the CMM with an NTP server.

Syntax

```
asg_ntp_sync_config set primary|secondary <ntp_ip|hostname> [-v <version>] [-r <timeout>]
asg_ntp_sync_config {disable|enable|delete}
asg_ntp_sync_config show
asg_ntp_sync_config -h
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set</td>
<td>Configure an NTP server</td>
</tr>
<tr>
<td>primary</td>
<td>The system uses this NTP server by default</td>
</tr>
<tr>
<td>secondary</td>
<td>The system uses this if the primary NTP server is not available</td>
</tr>
<tr>
<td>NTP Server &lt;ip</td>
<td>hostname&gt;</td>
</tr>
<tr>
<td>-v &lt;version&gt;</td>
<td>Server version of the NTP Service (default = NTPv4)</td>
</tr>
<tr>
<td>timeout</td>
<td>Timeout in seconds between refreshes (default = 300 seconds).</td>
</tr>
<tr>
<td>show</td>
<td>Show NTP Server configuration</td>
</tr>
<tr>
<td>disable</td>
<td>Disable NTP service</td>
</tr>
<tr>
<td>enable</td>
<td>Enable NTP service</td>
</tr>
<tr>
<td>delete</td>
<td>Delete primary or secondary NTP Service</td>
</tr>
<tr>
<td>-h</td>
<td>Show syntax and help information</td>
</tr>
</tbody>
</table>
Notes:

- This command runs `ntpd -u` on each SGM and the CMM to synchronize to the local time,
- If you define a refresh time that is less than the default (300 seconds), refresh occurs every 300 seconds.
- To allow time synchronization for all SGMs, you must disable the `replies_from_any_port` property for the NTP over UDP service:
  a) In GUIDedit, search for the NTP/UDP service.
  b) Go to the `replies_from_any_port` property.
  c) Change the property to `false`.
  d) Install policy.

Validation
1. Run `show time` on all SGMs and make sure that the time is the same.
2. Run `tcpdump` on port 123/UDP for the applicable interface to make sure that all SGMs initiate NTP connections.

Configuring SGMs (asg_blade_config)

Description
Use the `asg_blade_config` command to manage SGMs:

- Copy the SGM configuration from another SGM
- Change the synchronization start IP address
- Reset the system uptime value
- Get a policy from the Security Management server

Syntax

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`asg_blade_config pull_config [policy</td>
<td>all] [-force] &lt;ip_addr&gt;`</td>
</tr>
<tr>
<td><code>asg_blade_config full_sync &lt;ip&gt;</code></td>
<td>Run a full synchronization from another SGM.</td>
</tr>
<tr>
<td><code>asg_blade_config set_sync_start_ip &lt;ip&gt;</code></td>
<td>Synchronization interface on remote SGM</td>
</tr>
<tr>
<td>`asg_blade_config reset_uptime</td>
<td>reset_uptime_user`</td>
</tr>
<tr>
<td>`asg_blade_config get_smo_ip</td>
<td>is_in_security_group`</td>
</tr>
<tr>
<td>`asg_blade_config is_in_pull_conf_group</td>
<td>config fetch_smc`</td>
</tr>
<tr>
<td><code>asg_blade_config upgrade_start &lt;new_version&gt;</code></td>
<td>Return the Synchronization IP address of the Single Management Object, as defined in SmartDashboard. This address is not shown in SmartDashboard.</td>
</tr>
<tr>
<td>`asg_blade_config upgrade_stop</td>
<td>upgrade_stat`</td>
</tr>
</tbody>
</table>

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>pull_config</td>
<td>Copy the configuration from another SGM.</td>
</tr>
<tr>
<td>full_sync &lt;ip&gt;</td>
<td>Run a full synchronization from another SGM.</td>
</tr>
<tr>
<td>set_sync_start_ip &lt;ip&gt;</td>
<td>Synchronization interface on remote SGM</td>
</tr>
<tr>
<td>reset_uptime</td>
<td>Changes the Synchronization start IP address from the local SGM to the specified IP address.</td>
</tr>
<tr>
<td>reset_uptime_user</td>
<td>Resets the system uptime value on all SGMs to the current time.</td>
</tr>
<tr>
<td>get_smo_ip</td>
<td>An interactive command that resets the uptime for all SGMs to a user configured time.</td>
</tr>
<tr>
<td>is_in_pull_conf_group</td>
<td>Return the Synchronization IP address of the Single Management Object, as defined in SmartDashboard. This address is not shown in SmartDashboard.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>is_in_security_group</td>
<td>Make sure that the local SGM is in the Security Group.</td>
</tr>
<tr>
<td>is_in_pull_conf_group</td>
<td>Make sure that the local SGM is in the Pulling Configuration Group. If not, the SGM cannot copy the configuration and policy.</td>
</tr>
<tr>
<td>config fetch_smc</td>
<td>Get the policy from the Security Management Server, and send it to all SGMs.</td>
</tr>
<tr>
<td>upgrade_start &lt;new_version&gt;</td>
<td>Start upgrade procedure.</td>
</tr>
<tr>
<td>upgrade_stop</td>
<td>Stop the upgrade procedure.</td>
</tr>
<tr>
<td>upgrade_stat</td>
<td>Shows the upgrade procedure and policy status</td>
</tr>
<tr>
<td>upgrade_fc</td>
<td>Use the full connectivity upgrade option</td>
</tr>
</tbody>
</table>

**Troubleshooting asg_blade_config**

To troubleshoot problems associated with the `asg_blade_config` command, examine the logs stored at: `/var/log/blade_config`. For example, if the SGM unexpectedly reboots, you can search the log file for the word reboot to learn why.

**Backing Up and Restoring an SGM (backup_system)**

You use the `backup_system` command to save and restore SGM configuration, including:

- Chassis and operating system configuration
- Network configuration
- Security policy

When you backup an SGM, the backup files are copied to all other SGMs in your system. The data is contained in these `.tgz` files, located in the `/var/CPbackup/asg_backup` (default) directory:

```plaintext
<file_name>.asg_config.gz  
<file_name>.policy.tgz  
<file_name>.tgz
```

You can accept the default file names or enter your own file name at run time. When you use the setup wizard to configure a new 61000/41000 Security System, the backup files are automatically created with the file name `initial`.

The restore option lets you restore the SGM to a user-specified backup file. You can restore the Chassis and network configuration with or without the security policy. When you restore the security policy, SmartDashboard does not show the newly restored policy.

**Command Syntax**

```
backup_system backup|backup <file_name>
backup_system restore|restore <file_path>
backup_system show
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>backup</td>
<td>Backup with the default file name.</td>
</tr>
<tr>
<td>backup &lt;file_name&gt;</td>
<td>Backup with user specified file name.</td>
</tr>
<tr>
<td>restore</td>
<td>Restore from a file selected from an interactive menu, which shows all files in the default directory.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>restore &lt;file_path&gt;</td>
<td>Advanced restore from a file and path specified on the command line. This option is useful when you have backup files saved in a location other than the default location.</td>
</tr>
<tr>
<td>show</td>
<td>Show the saved backup files.</td>
</tr>
</tbody>
</table>

**Backup Procedure**

To backup an SGM:
2. Run:
   ```bash
   # backup_system backup [<file_name>]
   ```
   A confirmation shows when the backup is completed.

We recommend that you copy your backup files to external storage for disaster recovery.

**Restore Procedures**

To Restore an SGM - Interactive file selection:
2. Run:
   ```bash
   # backup_system restore
   ```
3. Select the backup file to restore from the menu.
4. Optional: Press `y` to backup the system.
   Enter a file name when prompted or press `Enter` to accept the default file name.
5. When prompted:
   - Press `y` to restore the system configuration and security policy.
   or
   - Press `n` to restore the system configuration only.
6. When prompted, press `Y` to continue.
7. When prompted, enter your user name.
8. When prompted, enter the reason for this restore operation.
9. When prompted, press `Y` to continue.
10. Run:
    ```bash
        # g_reboot -b all
    ```

To Restore an SGM - Manual file selection:
2. Run:
   ```bash
   # backup_system restore <file_path>
   ```
   Enter the complete path and file name, including the `.tgz`.
3. Optional: Press `y` to backup the system.
   Enter a file name when prompted or press `Enter` to accept the default file name.
4. When prompted:
   - Press `y` to restore the system configuration and security policy.
   or
   - Press `n` to restore the system configuration only.
5. When prompted, press `Y` to continue.
6. When prompted, enter your user name.
7. When prompted, enter the reason for this restore operation.
8. When prompted, press `Y` to continue.
9. Run:
   `# g_reboot -b all`

To restore to a new or different SGM:
1. Do a clean installation on a new SGM.
2. Establish SIC trust with the management server.
3. Copy the saved backup files to `/var/CPbackup/asg_backup/` on the new SGM.
4. Do the Interactive File Selection restore procedure (above).
   a) When prompted, select the saved backup file from the interactive menu.
   b) Select the option to restore both the configuration and security policy.
5. Run:
   `# g_reboot -b all`

Configuring SGM state (asg sgm_admin)

Use this command to manually change the state (Up or Down) for one or more SGMs.

**Syntax**

```
asg sgm_admin -b <sgm_ids> <up|down|down -a> [-p]
asg sgm_admin -h
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by <code>&lt;sgm_ids&gt;</code>. The <code>&lt;sgm_ids&gt;</code> can be:</td>
</tr>
<tr>
<td></td>
<td>- No <code>&lt;sgm_ids&gt;</code> specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>- One SGM</td>
</tr>
<tr>
<td></td>
<td>- A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>- A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>- One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>- The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-p</td>
<td>Persistent. The setting is kept after reboot</td>
</tr>
<tr>
<td>-a</td>
<td>Synchronize accelerated connections to other SGMs</td>
</tr>
<tr>
<td>-h</td>
<td>Show command syntax and help information</td>
</tr>
</tbody>
</table>

**Example**

```
> asg sgm_admin -b 2_03 -p
You are about to perform blade_admin up on blades: 2_03
Are you sure? (Y - yes, any other key - no) y
Blade_admin up requires auditing
Enter your full name: Fred
Enter reason for blade_admin up [Maintenance]: test
WARNING: Blade_admin up on blades: 2_03, User: Fred, Reason: test

Performing blade_admin up on blades: 2_03
[2_03]Setting blade to normal operation ...
[2_03]pulling configuration from: 192.0.2.16 (may take few seconds)
[2_03]Blade current state is ACTIVE
```

**Notes**

- When an SGM is in the Administrative Down state:
  - `gclish` commands do not run on this SGM.
  - Traffic is not sent to this SGM.
  - `asg_stat` shows the SGM as DOWN (admin).
- When an SGM is changed to Administrative Up, it automatically synchronizes the configuration from a different SGM that is in the UP state.
- This command generates log entries. To show the logs, run:
  > asg log audit
- This command is useful for debugging. We do not recommend that you use it in production environments because it causes performance degradation.

Image Management

You can:
- **Revert** to a saved image. This restores the system, including the configuration of the installed products.
- **Delete** an image from the local system.
- **Export** an existing image. This creates a compressed version of the image. You can then download the exported image to another computer and delete the exported image from the Gaia computer, to save disk space. You must not rename the exported image. If you rename a snapshot image, it is not possible to revert to it.
- **Import** uploads an exported image and makes an image of it (a snapshot). You can revert to the image at a later time.
- **See a list of saved images.**

**Global Image Management - (snapshot)**

Use this command to create, import, export, and show snapshots for all SGMs in the 61000/41000 Security System.

**Syntax**

To create a new image:

```plaintext
add snapshot VALUE desc VALUE
```

To delete an image:

```plaintext
delete snapshot VALUE
```

To export or import an image, or to revert to an image:

```plaintext
set snapshot export VALUE path VALUE name VALUE
set snapshot import VALUE path VALUE name VALUE
set snapshot revert VALUE
```

To show image information:

```plaintext
show snapshot VALUE all
show snapshot VALUE date
show snapshot VALUE desc
show snapshot VALUE size
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>snapshot VALUE</td>
<td>Name of the image</td>
</tr>
<tr>
<td>desc VALUE</td>
<td>Description of the image</td>
</tr>
<tr>
<td>snapshot export VALUE</td>
<td>The name of the image to export</td>
</tr>
<tr>
<td>snapshot import VALUE</td>
<td>The name of the image to import</td>
</tr>
<tr>
<td>path VALUE</td>
<td>The storage location for the exported image. For example: /var/log</td>
</tr>
<tr>
<td>name VALUE</td>
<td>The name of the exported image (not the original image).</td>
</tr>
<tr>
<td>all</td>
<td>All image details</td>
</tr>
</tbody>
</table>
Notes

- You must have sufficient free space on the backup partition to create snapshot image for all SGMs. The required free disk space is the actual size of the root partition, multiplied by 1.15.
- The free space required in the export file storage location is the size of the snapshot multiplied by two.
- The minimum size of a snapshot is 2.5G, so the minimum free space you need in the export file storage location is 5G.

**Image Management for Specified SGMs (g_snapshot)**

**Description**

Show and revert snapshots for specified SGMs or Chassis. This is in contrast to the gclish snapshot command, which works for all SGMs together. You must run this command from the Expert mode.

To show saved snapshots, run g_snapshot show. Enter the applicable parameters and do the instructions on the screen.

To restore SGMs or Chassis to a saved snapshot run g_snapshot revert and enter the applicable parameters.

**Syntax**

```plaintext
g_snapshot  [-b <sgm_ids>] show
g_snapshot  [-b <sgm_ids>] revert <snapshot_name>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>show</td>
<td>Shows saved snapshots for the specified SGMs or Chassis.</td>
</tr>
<tr>
<td>revert</td>
<td>Restore specified SGMs or Chassis to the specified snapshot.</td>
</tr>
<tr>
<td>&lt;snapshot_name&gt;</td>
<td>Snapshot file name</td>
</tr>
<tr>
<td>&lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;. The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>• No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>• One SGM</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>• A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>• One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>• The active Chassis (chassis_active)</td>
</tr>
</tbody>
</table>

**Examples**

- > g_snapshot -b 1_1,1_4 revert My_Snapshot
  This example restores SGMs 1_1 and 1_4 to My_Snapshot.
- > g_snapshot -b chassis2 revert My_Snapshot
  This example restores Chassis2 to My_Snapshot.
- > g_snapshot -b Chassis1 show
  This example shows the saved snapshots for all SGMs on Chassis1.
High Availability

**Chassis High Availability Active/Standby Mode**

The Chassis High Availability mechanism is based on two identical Chassis. One Chassis handles traffic (Active state), while the other Chassis is in Standby state. The Standby Chassis is synchronized with the Active Chassis so that traffic continues uninterrupted when there is a Chassis failover.

To make sure that the most reliable Chassis is active, each Chassis is assigned a quality grade based on continuous monitoring of its critical components. See set Chassis high-availability factors ("Setting Chassis Weights (chassis high-availability factors)" on page 109) for a detailed explanation of the grading system.

The Chassis with the highest is automatically selected as the Active Chassis. Whenever the other Chassis grade is greater than the minimum grade gap for failover, failover occurs automatically. See Setting the minimum gap failover (on page 107) for details.

Each Chassis port has its own unique MAC address. The MAC addresses are different for the ports on both Chassis. A Chassis failover event sends GARP packets for each interface. See GARP Chunk Mechanism (on page 214) for details.

You use gclish commands to configure parameters such as:
- Chassis HA grade factors, failover grade difference for failover,
- Failover freeze interval,
- ports factor
- Chassis HA Active Up or Primary Up mode.

**Synchronizing Clusters on a Wide Area Network**

The synchronization network can be spread over remote sites, which makes it easier to deploy geographically distributed clustering. There are two limitations to this capability:

1. The synchronization network must guarantee no more than 100ms latency and no more than 5% packet loss.
2. The synchronization network may only include switches and hubs. No routers are allowed on the synchronization network, because routers drop Cluster Control Protocol packets.

**Setting the Minimum Gap Failover**

Use the `set chassis high-availability failover` command to set the minimum grade gap for Chassis failover.

Syntax:

```
> set chassis high-availability failover <1-1000>
```

**Setting the Freeze Interval**

Use the `set chassis high-availability freeze_interval` command to set a freeze interval. After a failover, the Chassis is prevented or frozen from failing over again until the interval expires. In addition, when the grade of standby Chassis is changed, and with the new grade, this Chassis will be active, there is another freeze before becoming active Chassis. The reason for this freeze, before becoming active Chassis, is to let the Chassis grade stabilize before becoming active Chassis, and avoid grade flapping (for example: fan goes up, down, up, down...).

Syntax:

```
> set chassis high-availability freeze_interval <1-1000>
```

Note: When running `asg stat` after Chassis failover, you will be notified with the freeze time:
Setting Port Priority (for Each Port)

Use the `set chassis high-availability port priority` command to set a port priority (high or standard) for each port.

**Syntax:**

```
set chassis high-availability port <interface> priority <1-2>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Standard priority</td>
</tr>
<tr>
<td>2</td>
<td>Other priority</td>
</tr>
</tbody>
</table>

Use this command together with the `set chassis high-availability factors port` command.

1. First set the port grade as standard or high.
   For example:
   ```shell
   set chassis high-availability factors port standard 50
   ``
   This sets the standard grade at 50.
2. Then decide which ports have the high grade or the standard grade.
   For example:
   ```shell
   set chassis high-availability port eth1-01 priority 2
   ``
   This assigns to `eth1-01` the standard port grade.

Chassis HA - Link Preemption Mechanism

**Description:**

The Link Preemption Mechanism prevents constant Chassis fail-over and failback whenever there is interface link flapping.

When you enable this feature, an interface state that changes from down to up, is only considered in the Chassis grade if the link state is up for X seconds (default is 10 sec).

**Configuration:**

The Link Preemption Mechanism is enabled by default with a preemption time of 10 seconds.

To configure the preemption time, run these commands from gclish:

```sh
> fw ctl set int fwha_ch_if_preempt_time < preemption time >
> update_conf_file fwkern.conf fwha_ch_if_preempt_time=< preemption time >
```

Sample commands that set the preemption time to 20 seconds:

```sh
> fw ctl set int fwha_ch_if_preempt_time 20
> update_conf_file fwkern.conf fwha_ch_if_preempt_time=20
```

**Deactivation:**

To disable Link Preemption Mechanism, run these commands from gclish:

```sh
> fw ctl set int fwha_ch_if_preempt_time 0
> update_conf_file fwkern.conf fwha_ch_if_preempt_time=0
```

**Verification:**

To check the preemption time value, run this command from gclish:

```sh
> fw ctl get int fwha_ch_if_preempt_time
```

Chassis HA – Sync Lost Mechanism

The 61000/41000 Security System uses the Check Point proprietary *Cluster Control Protocol* (CCP) to send UDP control packets between two High Availability Chassis. When a sync interface fails, it is necessary to send a SYNC_LOST message the other Chassis. The SYNC_LOST mechanism handles loss of connectivity between two Chassis on the Sync network.

To prevent the two Chassis from changing their states to Active, a SYNC_LOST CCP is sent over non-sync interface (the Data Ports and Management interfaces) to the other Chassis. This causes the two Chassis to
freeze their current state until connectivity between the two Chassis is restored. During the Sync Loss, the Standby Chassis, does not change its state to Active until it stops receiving SYNC_LOST packets from the other Chassis.

The 61000/41000 Security System sends SYNC_LOST messages in this manner:
- For VSX environments - All interfaces of the VS0 context only
- For non-VSX environments - All Chassis interfaces

**Configuration**:

Synchronize Lost mechanism is enabled by default.

To disable Sync Lost Mechanism, run these commands from gclish:

```bash
> fw ctl set int fwha_ch_sync_lost_mechanism_enabled 0
> update_conf_file fwkern.conf fwha_ch_sync_lost_mechanism_enabled=0
```

To enable Sync Lost Mechanism, run these commands from gclish:

```bash
> fw ctl set int fwha_ch_sync_lost_mechanism_enabled 1
> update_conf_file fwkern.conf fwha_ch_sync_lost_mechanism_enabled=1
```

**Verification**:

To check whether the mechanism is enabled:

```bash
> fw ctl get int fwha_ch_sync_lost_mechanism_enabled
(1-enabled, 0-disabled)
```

---

**Setting Chassis Weights (chassis high-availability factors)**

Each component in a Chassis has a weight factor, which is a numerical value that reflects the component importance level. Ports might be more important than fans and receive a higher value or a greater weight. The Chassis grade is the sum of all these component weights. In a high-availability dual-Chassis deployment, the Chassis with the higher grade becomes active and processes traffic. The grade of each component = (Unit Weight) X (Number of UP components)

To see the weight of each component, run: `asg stat -v`.

Use the `set chassis high-availability factors` command to configure a component's weight.

**Syntax**

```
set chassis high-availability factors [SGM <factor> | port high <factor> | port standard <factor> | sensor cmm <factor> | sensor fans <factor> | sensor power_supplies <factor> | sensor ssm <factor> | pnote pingable_hosts <factor>]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGM</td>
<td>Sets the weight factor for an SGM</td>
</tr>
<tr>
<td></td>
<td>The weight factor must be between 0 and 1000</td>
</tr>
<tr>
<td></td>
<td>Example: <code>set chassis high-availability factors sgm 100</code></td>
</tr>
<tr>
<td>port high</td>
<td>A port has one of two grades: high or standard. This parameter sets a weight factor for the high grade</td>
</tr>
<tr>
<td></td>
<td>The weight factor must be between 0 and 1000</td>
</tr>
<tr>
<td></td>
<td>Example: <code>set chassis high-availability factors Port high 70</code></td>
</tr>
<tr>
<td></td>
<td>This means that ports set to high grade have a weight of 70.</td>
</tr>
<tr>
<td>port standard</td>
<td>A port has one of two grades: high or standard. This parameter sets a weight factor for the standard grade</td>
</tr>
<tr>
<td></td>
<td>The weight factor must be between 0 and 1000</td>
</tr>
<tr>
<td></td>
<td>Example: <code>set chassis high-availability factors Port standard 50</code></td>
</tr>
<tr>
<td></td>
<td>This means that ports set to standard grade have a weight of 50.</td>
</tr>
<tr>
<td>Sensor CMMs</td>
<td>Sets a weight factor for CMMs</td>
</tr>
<tr>
<td></td>
<td>The weight factor must be between 0 and 99</td>
</tr>
<tr>
<td></td>
<td>Example: <code>set chassis high-availability factors sensor cmm 40</code></td>
</tr>
</tbody>
</table>
Chassis High Availability Active/Active Mode

In Active/Active mode the two Chassis in a dual Chassis configuration handle connections. Connections between the two Chassis are synchronized.

This mode is supported only in a Layer 2 (L2) topology.

Configure this mode when:

- An external device or protocol distributes connections to the two Chassis and so determines which Chassis is Active.
- Routing to Chassis is not symmetric. Packets on some connections may be sent to the two Chassis.

Changing the High Availability Mode

When changing from Active/Active to Active/Standby or from Active/Standby to Active/Active:

1. Put the one Chassis in an administrative DOWN state. From gclish run:
   asg chassis_admin -c <Chassis_id> down
2. Put the same Chassis in an administrative UP state. From gclish run:
   asg chassis_admin -c <Chassis_id> up

To change the High Availability Chassis mode:

Run

set chassis high-availability mode [0|1|2|3]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Active/Standby: No primary Chassis. Also known as Active Up mode. In this mode, the Chassis that is UP stays up until the other Chassis gets a higher grade.</td>
</tr>
<tr>
<td>1</td>
<td>Active/Standby: Chassis 1 is Primary Chassis. Also known as Primary Up mode. In this mode, if Chassis 1 has a grade that is high enough to make it Active, it will become Active and will take over from Chassis 2. Chassis 2 then becomes Standby.</td>
</tr>
<tr>
<td>2</td>
<td>Active/Standby: Chassis 2 is Primary Chassis. Also known as Primary Up mode. In this mode, if Chassis 2 has a grade that is high enough to make it Active, it will become Active and will take over from Chassis 1. Chassis 1 then becomes Standby.</td>
</tr>
<tr>
<td>3</td>
<td>Active/Active Mode</td>
</tr>
</tbody>
</table>
Admin Down on First Join (down_on_first_join)

You can configure the 61000/41000 Security System to automatically set a newly installed SGM in a Security Group to the Admin Down state. This lets the administrator make sure that the SGM is configured correctly before it handles traffic.

Syntax

```
set chassis high-availability down_on_first_join [0|1]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1 - Admin Down on First Join is enabled</td>
</tr>
</tbody>
</table>

To add a new SGM to a Security Group with Admin Down:
1. Run:
```
set chassis high-availability down_on_first_join 1
```
2. Install the new SGM and add it to the Security Group.
3. Run this command to set the SGM to the UP state:
```
> asg sgm_admin -b <sgm_ids> up -p
```

Chassis ID Configuration

When installing and configuring Chassis high availability, you must make sure that Chassis ID are different before you start to configure the software. Chassis IDs are configured on the CMM and should be <1> for the first Chassis and <2> for the second Chassis.

**Note:** If the 61000/41000 Security System is up and running, change the Chassis ID on the Standby Chassis, hence you will have to perform Chassis failover.

Procedure for 61000 Security System

1. Remove the upper CMM from the Chassis.
2. Log in to the remaining CMM.
3. Connect the serial cable to the console port on the CMM.
4. Connect to the CMM with a terminal emulation application, such as PuTTY.
5. Make sure that the Speed (baud rate) is set to 9600.
   No IP address is necessary.
6. Log in with user name and password admin/admin.
7. Open `/etc/shmm.cfg` in a text editor.
8. Search for and set `SHMM_CHASSID=` to the correct Chassis ID
   ```
   Chassis ID
   SHMM_CHASSID=<Chassis_id>
   ```
9. Remove the lower CMM, which you just reconfigured, from the Chassis.
10. Insert the upper CMM into the Chassis.
11. Do steps 2 - 8 on the upper CMM.
12. Remove the upper CMM from the Chassis.
13. Insert both CMMs into the Chassis.
14. Attach the correct identification labels to the Chassis and CMMs.
   This step is required if the Chassis has already been configured (After First Time Configuration Wizard)
15. Remove all SGMs from the Chassis and then reinsert them.
   This step causes a hard reboot of the system.

Procedure for 41000 Security System

1. Remove the right CMM from the Chassis
2. Log in to the remaining CMM.
3. Connect the serial cable to the console port on the CMM.
4. Connect to the CMM with a terminal emulation application, such as PuTTY.
5. Make sure that the Speed (baud rate) is set to 9600.
   No IP address is necessary.
6. Log in with user name and password `admin/admin`.
7. Open `/etc/shmm.cfg` in a text editor.
8. Search for and set `SHMM_CHASSID=` to the correct Chassis ID
   
   ```
   Chassis ID
   SHMM_CHASSID=<Chassis_id>
   ```
9. Remove from the left CMM from the chassis.
10. Insert the right CMM into the Chassis.
11. Do steps 2-8 on the right CMM.
12. Remove the right CMM from the Chassis.
13. Insert both CMMs into the Chassis.
14. Attach the correct identification labels to the Chassis and CMMs.
   This step is required if the Chassis has already been configured (After First Time Configuration Wizard)
15. Remove all SGMs from the Chassis and then reinsert them.
   This step causes a hard reboot of the system

### Configuring a Unique IP address per Chassis (UIPC)

**Description**

In dual-Chassis deployment:

- A heavy load on the active Chassis can prevent you from making a network connection to the SMO and implementing management tasks.
- You may also require direct access to the standby Chassis to troubleshoot a problem, such as an SGM that is down. (You cannot use the SMO to connect to the standby Chassis).

These two scenarios can be solved by assigning a unique IP address to each Chassis. Assigning a unique IP address to each chassis adds an extra alias IP to the management interfaces on all SGMs in the chassis.

- If there is a high load on the SMO, connect using the unique IP assigned to the standby chassis. The SGMs on the standby chassis are always UP and available to run `gclish` management commands.
- When you need to connect directly to the standby chassis, use the standby chassis' unique IP.

**Notes**

- Similar to the SMO mechanism, only one SGM owns the UIPC task
- The UIPC feature is disabled by default
- If the 61000/41000 Security System is not managed by a management port, the unique IP can be added to one of the data ports.

**Syntax**

```plaintext
set chassis id <Chassis_id> general unique_ip <ip_addr>
delete chassis id <Chassis_id> general unique_ip
show chassis id <Chassis_id> general unique_ip
```

In `gclish`, run:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;Chassis_id&gt;</code></td>
<td>Valid values: 1/2/all</td>
</tr>
<tr>
<td><code>ip_addr</code></td>
<td>An alias IP address on the same network as one of the SGMs interfaces</td>
</tr>
</tbody>
</table>
Manual configuration

Although the UIPC feature is automatically enabled when you run the configuration commands, you can also manually enable or disable it:

- To manually enable UIPC, run: `g_fw ctl set int fwha_uipc_enabled 1`
- To manually disable UIPC run: `g_fw ctl set int fwha_uipc_enabled 0`

Example 1   set chassis id 1 general unique_ip 172.16.6.186

Output
>set chassis id 1 general unique_ip 172.16.6.186
Adding alias IP: 172.16.6.186 to chassis 1
Alias IP was added successfully

Example 2   delete chassis id 1 general unique_ip

Output
>delete chassis id 1 general unique_ip
Deleting alias IP 172.16.6.186 of chassis 1
Alias IP was deleted successfully

**asg_sync_manager**

Description

The asg_sync_manager enables the user to define its required synchronization level. The synchronization level is a combination of system synchronization settings (e.g. backup connections to standby Chassis) and specific rules (e.g. do not synchronize HTTP connections). Specific rules are referred to as sync exception table. Connections are serially matched against this table.

In addition to the synchronization settings, this utility also controls SecureXL delayed synchronization parameters: when connection is created within SecureXL (from SecureXL template), asg_sync_manager can set the period until it will be synchronized to firewall.

By default, specific sync exception table consists of a single rule, which is not to synchronize DNS traffic.

Key synchronization properties are also displayed in asg stat -v

Usage The utility is interactive. The following options are available:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Print sync exceptions table</td>
<td>This view displays the sync exception table. Each entry in this table consists of:</td>
</tr>
<tr>
<td></td>
<td>1. &lt;5-tuple, including wild cards&gt;</td>
</tr>
<tr>
<td></td>
<td>2. synchronization mode (none, within Chassis only, between Chassis only, both within ,between Chassis and to all SGMs)</td>
</tr>
<tr>
<td></td>
<td>3. SecureXL delayed synchronization value</td>
</tr>
<tr>
<td></td>
<td>In addition, global synchronization values are displayed</td>
</tr>
<tr>
<td>2) Add new sync exceptions rule</td>
<td>Add new rule to the sync exceptions table. The user can hit enter at any stage to apply the default value. Specific rules allow the use of wildcards within 5-tuple. New rule will apply for new connections</td>
</tr>
<tr>
<td>3) Delete old sync exception rule</td>
<td>Delete rule from the sync exceptions table</td>
</tr>
<tr>
<td>4) Set sync between Chassis flag on / off</td>
<td>Global system setting: whether to synchronize connections to backup Chassis</td>
</tr>
<tr>
<td>5) Set sync within local Chassis flag on / off</td>
<td>Global system setting: whether to synchronize connections within active Chassis</td>
</tr>
</tbody>
</table>
Option | Description
--- | ---
6) Configure sync between Chassis SGMs ratio | Minimal SGMs ratio between active and backup Chassis for synchronization to occur. If the number of UP SGMs in standby Chassis is significantly low, compared to active Chassis, synchronization might overload them. Default ratio for synchronization is 70% and it can be re-configured here. After configuration, user can also choose to restore default settings.
7) Set default delay notifications | Default delayed synchronization setting are divided to HTTP related services (30) and all other services (5). User can reconfigure these settings here. Note that when configuring service delayed synchronization in SmartDashboard it overrides these settings.
8) Enable / Disable unicast sync | The user can enable / disable unicast sync (correction layer will be enabled / disabled accordingly) and return to legacy synchronization scheme (synchronize connections to all SGMs). Changing this setting requires reboot of all SGMs.

**Example 1**

```
asg_sync_manager
```

**Output**

Please choose one of the following:

1) Print sync exceptions table
2) Add new sync exceptions rule
3) Delete old sync exception rule
4) Set sync between Chassis flag on / off
5) Set sync within local Chassis on / off
6) Configure sync between Chassis blades ratio
7) Set default delay notifications
8) Enable / Disable unicast sync
e) Exit

**Tip:** you can always press e to return to main menu

**Example 2**

The following example shows how to add rule for all Virtual Systems which limits the synchronization of HTTP traffic, initiated from network 3.3.3.0/24 to network 4.4.4.0/24 to active Chassis only:

```
Enter vs range: [default: 0]
>all
Enter source IP [0.0.0.0]:
>3.3.3.0
Enter source IP mask length [0]:
>24
Enter destination IP [0.0.0.0]:
>4.4.4.0
Enter destination IP mask length [0]:
>24
Enter destination port [0]:
>80
Enter IP protocol number (for example: tcp = 6, udp = 17):
>6
Enter the sync exception rule [3 - sync to all chassis]:
0 = no sync
1 = sync only to local chassis
2 = sync only to other chassis
3 = sync to all chassis
4 = sync to all SGMs
>1
Enter delay notification [30 - http, 5 - other]:
> to insert new exception to vs 0-1,2: <3.3.3.0/24, 4.4.4.0/24, 80, 6> sync rule: 1, delay: 5 ? (y/n)
>y
```
After adding this rule, sync exception table will be displayed as follows:

+-----------------------------------------------+---------+---------+-----+-----------------+-------+-------+-----+-----+-----------+
|      |  | VS     | Source  | Mask | Destination   | Mask | DPort | Ipp | Sync | Delay     |
| Idx  |  |        |         |      |              |      |       |    |     |           |
| 1    |  | 0-1,2  | 0.0.0.0 | 0    | 0.0.0.0      | 0    | 53    | 17  | 0   | 5         |
| 2    |  | 0-1,2  | 3.3.3.0 | 24   | 4.4.4.0      | 24   | 80    | 6   | 1   | 5         |

*Sync: 0=no sync, 1=sync only to local Chassis, 2=sync only to other Chassis, 3 = sync to all Chassis
**Delay: The time it takes for connections created from templates to synchronize

Verifying the High Availability Configuration

Each of the set commands has a corresponding show command. For example:

To verify

set chassis high-availability mode <0-3>

Run

show chassis high-availability mode

Monitoring, Logs and Auditing

Redirecting Alerts and Logs to External syslog server (asg_syslog)

Description

asg_syslog command should be used in order to redirect alert messages and firewall logs to remote syslog servers.

This command allows configuring the following:

- Remote syslog servers either by IPv4 address or by hostname to log all alert messages.
- Remote syslog servers to log FW logs.
- Disable/Enable firewall logs to be sent to the Log Server. (Log Server is configured from SmartDashboard: Right-click gateway object > Edit > Logs and Masters > Log Servers)
- Verify configuration consistency on all SGMs.
- Recover configuration on all SGMs by forcing current SGM configuration on all SGMs.

asg_syslog is available only from Expert shell

Syntax:

asg_syslog <verify|print [-v] | recover>
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;verify&gt;</td>
<td>Verify configuration consistency on all SGMs</td>
</tr>
<tr>
<td>&lt;print&gt; [-v]</td>
<td>Print remote syslog servers configuration</td>
</tr>
<tr>
<td>&lt;recover&gt;</td>
<td>Recover configuration files on all SGMs and restart syslog service</td>
</tr>
</tbody>
</table>

**Example 1**

```bash
asg_syslog verify
```

**Output**

```
<table>
<thead>
<tr>
<th>Service</th>
<th>Path</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPLog</td>
<td>/etc/syslog_servers_list.conf</td>
<td>Passed</td>
</tr>
<tr>
<td>Alert</td>
<td>/etc/syslog.conf</td>
<td>Passed</td>
</tr>
</tbody>
</table>
```

**Notes**

Configuration files on all SGMs are identical

**Example 2 asg_syslog print**

**Output**

```
<table>
<thead>
<tr>
<th>Service</th>
<th>Server IP</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>alert</td>
<td>5.5.5.5</td>
<td>disable</td>
</tr>
<tr>
<td>alert</td>
<td>6.6.6.6</td>
<td>enable</td>
</tr>
</tbody>
</table>
```

* Firewall logging is disabled

**Syntax**

Configure remote syslog servers for alerts:

```
Usage
asg_syslog <disable|enable|set|delete> alert <IP address|hostname>
```

Configure remote syslog server for firewall logs:

```
Usage
asg_syslog <disable|enable|set[-s <status>]|delete> cplog <IP address>
```

*Note: When configuring alert syslog servers, syslog service is being restarted on all SGMs.*

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;set&gt;</td>
<td>Set remote syslog server</td>
</tr>
<tr>
<td>-s &lt;status&gt;</td>
<td>Set connection with status &lt;enable&gt; or &lt;disable&gt;</td>
</tr>
<tr>
<td>&lt;disable&gt;</td>
<td>Disable sending Firewall logs / alerts to a remote syslog server defined by IP address or host name.</td>
</tr>
<tr>
<td></td>
<td>Note: disable operation will not remove the configuration. You can enable it again using the 'enable' parameter</td>
</tr>
<tr>
<td>&lt;enable&gt;</td>
<td>Enable sending Firewall logs / alerts to a remote syslog server defined by IP address or host name.</td>
</tr>
<tr>
<td></td>
<td>This parameter can be used after the remote server has been configure (see 'set' parameter)</td>
</tr>
</tbody>
</table>
### Parameter	Description
---
<delete>  
Delete remote syslog server.

<ip address | hostname>  
IPV4 address or hostname of the remote syslog server. Hostname will be applicable when hostname resolution can be made, either via DNS or by static configuration.

**Examples:**

```bash
# asg_syslog set alert 5.5.5.5  
Writing new configuration  
Updating all SGMs with new configuration  
Restarting syslog service on all SGMs  
syslog alert server 5.5.5.5 configured successfully
```

<table>
<thead>
<tr>
<th>Service</th>
<th>Server IP</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>alert</td>
<td>5.5.5.5</td>
<td>enable</td>
</tr>
</tbody>
</table>

Firewall logging is disabled

```bash
# asg_syslog disable alert 5.5.5.5  
Updating all SGMs with new configuration  
Restarting syslog service on all SGMs  
syslog alert server 5.5.5.5 status changed to disable
```

<table>
<thead>
<tr>
<th>Service</th>
<th>Server IP</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>alert</td>
<td>5.5.5.5</td>
<td>disable</td>
</tr>
</tbody>
</table>

* Firewall logging is disabled

```bash
# asg_syslog set cplog 6.6.6.6
```

Writing new configuration  
Updating all SGMs with new configuration  
syslog cplog server 6.6.6.6 configured successfully

<table>
<thead>
<tr>
<th>Service</th>
<th>Server IP</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>alert</td>
<td>5.5.5.5</td>
<td>disable</td>
</tr>
<tr>
<td>cplog</td>
<td>6.6.6.6</td>
<td>disable</td>
</tr>
</tbody>
</table>

* Firewall logging is disabled

**Syntax:**

To Disable/Enable firewall logs to be sent to Firewall log server (i.e. SmartView Tracker):

```
asg_syslog < disable | enable > log_server
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;disable&gt;</td>
<td>Enable sending firewall logs to the log server. (log server is configured in Smart Dashboard)</td>
</tr>
<tr>
<td>&lt;enable&gt;</td>
<td>Disable sending firewall logs to the log server. (log server is configured in Smart Dashboard)</td>
</tr>
</tbody>
</table>
Example:

# asg_syslog disable log_server

# asg_syslog print -v

<table>
<thead>
<tr>
<th>Service</th>
<th>Server IP</th>
<th>Port</th>
<th>Protocol#</th>
<th>RFC version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Status</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

* Firewall logging is disabled

**Monitoring Management Interfaces Link State**

**Description**

By Default, 61000/41000 Security System monitor link state only on data ports (ethX-YZ). The Management Monitor feature lets SNMP monitor Management ports for the SSM60 and SSM160 components. The link state is sent to all SGMs and is integrated as part of the Chassis High Availability mechanism. Once enabled, management ports show in the asg stat -v output.

Monitored management ports are included in the Chassis grade mechanism, according to pre-defined factors (default = 11). In addition, the asg if command shows the link state of Management interfaces based on the feature mechanism.

**Note** - For the SSM60, it is necessary to pre-configure the Base Switch to enable the SNMP server before you enable the feature itself. See ("SSM60 snmp-server configuration" on page 119) for details. After you configure the SNMP server, run:

```
set chassis high-availability management-monitoring on
```
SSM60 snmp-server configuration

On each Chassis, login to every SSM base switch address using telnet.

Enter ‘enable’ mode
Enter ‘configure terminal’ mode
Execute the following 5 commands:

- snmp-server enable
- snmp-server view myview 1.3 included
- snmp-server group mygroup v3 auth read myview write myview notify myview
- snmp-server system-name BI_cp
- snmp-server user asg1 group mygroup v3 auth md5 asg1asg1

exit ‘configure terminal’ mode
execute ‘write’ to save configuration

Validating snmp configuration

After configuring all SSM60, validate configuration by running the following command from shell:

mgmt_monitor snmp_verify
Output after successful configuration:

Please wait while querying the snmp-servers on all SSMs

Chassis 1:
---------
SSM1:  OK
SSM2:  OK

Chassis 2:
---------
SSM1:  OK
SSM2:  OK

Configuring Non-local RADIUS Users Management port factor

Management Ports are integrated as part of the Chassis HA grade mechanism therefore; setting Management port factors (for all Management ports) are the same as ‘Standard’ or ‘Other’ data ports factors.

Use the `set Chassis high-availability factors` port management command to change management port factors (default = 11)

**Log Server Distribution (asg_log_servers)**

**Description** In SmartDashboard, multiple log servers can be configured per gateway object. In such an environment, the gateway sends its logs to all of its configured log servers. If the gateway object is a 61000/41000 Security System appliance (consisting of many SGMs) each SGM will send its logs to all log servers in the configuration. To reduce the load on the log servers, use the `asg_log_servers` command to enable log distribution (load sharing).

When enabled, each SGM sends its logs to one log server only. The decision as to which Log Server will be assigned to which SGM is done automatically and cannot be defined by the user.

**Syntax** `asg_log_servers`

**Example**

`asg_log_servers`

**Output**

```
> asg log_servers
+---------------------------------+ Log Servers Distribution +---------------------------------+
+---------------------------------+                           +---------------------------------+

Log Servers Distribution Mode: Disabled

Available Log Servers:
  * logServer
  * Giga
  * LogServer?

Logs will be sent to all available servers.

Choose one of the following options:
-----------------------------------------
1) Configure Log Servers Distribution mode
2) Exit
>1

+---------------------------------+ Log Servers Distribution +---------------------------------+
+---------------------------------+                           +---------------------------------+

Log Servers Distribution Mode: Disabled

Choose the desired option:
-----------------------------------------
1) Enable Log Servers Distribution mode
2) Disable Log Servers Distribution mode
3) Back
```
If log server distribution is already enabled, the command shows which log servers are assigned to each SGM:

```
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Log Servers Distribution</td>
</tr>
<tr>
<td>-----------------------------</td>
</tr>
</tbody>
</table>

Log Servers Distribution Mode: Enabled

Available Log Servers:
* LogServer
* Gaia
* LogServer2

Log Servers Distribution:

<table>
<thead>
<tr>
<th>SGM id</th>
<th>Chassis 1</th>
<th>Chassis 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Gaia</td>
<td>Gaia</td>
</tr>
<tr>
<td>2</td>
<td>LogServer2</td>
<td>LogServer2</td>
</tr>
<tr>
<td>3</td>
<td>LogServer</td>
<td>LogServer</td>
</tr>
<tr>
<td>4</td>
<td>Gaia</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>LogServer</td>
<td>-</td>
</tr>
<tr>
<td>7</td>
<td>-</td>
<td>Gaia</td>
</tr>
<tr>
<td>8</td>
<td>-</td>
<td>LogServer2</td>
</tr>
<tr>
<td>9</td>
<td>LogServer</td>
<td>LogServer</td>
</tr>
<tr>
<td>10</td>
<td>Gaia</td>
<td>-</td>
</tr>
<tr>
<td>11</td>
<td>LogServer2</td>
<td>-</td>
</tr>
<tr>
<td>12</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

("-" - SGM is not in Security Group)

Choose one of the following options:

1) Configure Log Servers Distribution mode
2) Exit

Note - You cannot configure an SGM to send its logs to a particular log server. Distribution takes place automatically.

Configuring a Dedicated Logging Port

Description The 61000/41000 Security System logging mechanism lets each SGM forward logs directly to a logging server over the SSM's management ports. However, management ports can experience a high load when a large number of logs are forwarded. Load on the SSM management ports can be significantly reduced by:

- Setting up a dedicated SSM port for logging
- Assigning the dedicated logging port to each SGM

To set up a dedicated logging port:

1. Install a log server and create an object for it in SmartDashboard.
2. Connect the log server directly to a management port on the SSM.
   
   Important - Do not use the same port which connects to the Security Management server.
3. In gclish, run the set interface command to configure the port as a dedicated logging port:

Syntax

```
set interface <interface> ipv4-address <IP> mask-length <length>
```
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface</td>
<td>The interface that connects directly to the log server.</td>
</tr>
<tr>
<td>ipv4-address</td>
<td>IPv4 address of the logging server</td>
</tr>
<tr>
<td>mask-length</td>
<td>mask length</td>
</tr>
<tr>
<td>&lt;length&gt;</td>
<td></td>
</tr>
</tbody>
</table>

**Example**

```
set interface eth1-Mgmt2 ipv4-address 2.2.2.10 mask-length 24
```

**Output**

```
> set interface eth1-Mgmt2 ipv4-address 2.2.2.10 mask-length 24
 1_01: success
 1_02: success
 1_03: success
 2_01: success
 2_02: success
 2_03: success
>
```

**Notes**

- For each SGM, `eth1-Mgmt2` is set as a unique logging port.
- `2.2.2.0/24` is the logging server network or leads to the logs server network.

**Connecting to the logging server:**

1. Open SmartDashboard.
2. Open the Single Management Object (SMO) for the 61000/41000 Security System.
3. On the **Logs and Masters > Log Servers** page, select **Define Log Servers**.
4. Select the dedicated log server.
5. Install policy.

**Note**

- The SMO in SmartDashboard makes sure that return traffic from the logging server, such as ACKS, reaches the correct SGM.
- 61000/41000 Security System can be configured to send logs to more than one log server.

**Command Auditing**

Command auditing is a way of:

- Notifying users about critical actions they are about to take
- Obtaining confirmation for critical actions
- Creating forensic logs

If users confirm the action, they are requested to supply their names and a reason for running the command. If the command affects a critical device or a process (pnote) a second confirmation may be required.
For example, if you use administrative privileges to change the state of a SGM to DOWN the output looks like this:

```
> asg_sgm_admin -b 2_01 down
You are about to perform sgm_admin down on blades: 2_01
Are you sure? (y - yes, any other key - no) y
sgm_admin down requires auditing
Enter your full name: John Smith
Enter reason for sgm_admin down [Maintenance]:
WARNING: sgm_admin down on SGM: 2_01, User: John Smith, Reason: Maintenance
```

To view the audit logs, run `asg log audit`:

```
# asg log audit
Aug 01 08:53:45 1_01 WARNING: sgm_admin down on SGM: 1_02, User: susan, Reason: Maintenance
Aug 02 08:54:21 1_01 WARNING: Reboot on Blades: 1_01, User: susan, Reason: Maintenance
Aug 04 08:55:33 2_01 WARNING: sgm_admin up on SGMs: 1_02, User: susan, Reason: Maintenance
Aug 06 11:48:30 2_01 CRITICAL: Sync turn off between chassis on blades: all, User: ms, Reason: Maintenance
Aug 07 11:49:02 2_01 CRITICAL: Sync turn on between chassis on blades: all, User: Paul, Reason: increase performance
Aug 08 11:49:17 2_01 CRITICAL: Sync turn off within chassis on blades: all, User: Tom, Reason: testing sync
Aug 08 11:49:43 2_01 CRITICAL: Sync turn on within chassis on blades: all, User: Peter, Reason: Maintenance
Aug 09 12:38:24 2_01 CRITICAL: Reboot on blades: all, User: ms, Reason: Maintenance
```

Port Mirroring (SPAN Port)

Port Mirroring lets a gateway listen to traffic on a mirror port or SPAN port on a switch. The mirror port on a Check Point gateway is typically configured to monitor and analyze network traffic with no effect on the physical network. The mirror port duplicates the network traffic and records the activity in logs.

You can use mirror ports:

- As a permanent part of your deployment, to monitor the use of applications in your organization.
- As an evaluation tool to see the capabilities of the Application Control and IPS Software Blades before you decide to purchase them.

The mirror port does not enforce a policy and therefore you can only use it to see the monitoring and detection capabilities of the blades.

Benefits of a mirror port include:

- There is no risk to your production environment.
- It requires minimal set-up configuration.
- It does not require expensive TAP equipment.

### Configuring Port Mirroring on a Security Gateway

1. To configure port mirroring log, run `> add bridging group 0` to create a new bridge group.
2. Run `> add bridging group 0 <if_name>` to add the interface to bridging group br0.
3. In SmartDashboard, add the bridge interface to the 61000/41000 Security System gateway object.
4. Change the bridge interface name to `br0`.
5. Select **Global Properties** from the **Policy** menu.
6. Select **Stateful Inspection** and clear the **Drop out of state packets** options.

![Stateful Inspection Settings](image)

7. Install policy.
8. From the 61000/41000 Security System command line, run:
   ```
   > asg_span_port set
   ```
   This defines the interface as a SPAN port.
9. Reboot all SGMs.

   We recommend that you run `asg if` to make sure that the bridge and its related interface are up and running.

### Configuring Port Mirroring for a VSX Gateway

To configure port mirroring for a VSX Gateway:
1. In SmartDashboard, create new Virtual System in the Bridge mode.
2. Add an interface for the SPAN port that is connected to the physical port of the SSM.
3. Select **Global Properties** from the **Policy** menu.
4. Select **Stateful Inspection** and clear the **Drop out of state packets** options.

![Stateful Inspection Settings](image)

5. Install policy on the Virtual System.
6. Open an SSH connection to the VSX Gateway.
7. From the new Virtual System context, run:
   ```
   > asg_span_port set
   ```
8. Reboot all SGMs.

**Disabling Port Mirroring on a VSX Gateway**

To disable port mirroring on a VSX Gateway:

1. Go to the Bridge Mode Virtual System context.
2. Run:
   ```
   > asg_span_port unset
   ```
3. **Recommended**: Do these steps in SmartDashboard:
   a) **Go to Policy > Global Properties > Stateful Inspection**.
   b) Select both **Drop out of state packets** options.
4. We recommend that you **Undo** step 4 in Configuring Port Mirroring on a VSX Gateway
5. Install policy on the Virtual Systems.
6. Reboot all SGMs.

**Additional Port Mirroring Configuration Steps**

Do these recommended additional steps as necessary for the specified scenarios:

- In Application and URL Filtering policies, change the destination default settings from **internet** to **any**
- For IPS, turn off the Sequence Verifier (Reduces CPU Utilization)
- Disable **Out of State Protections** (Reduces CPU Utilization)
- Set the **Distribution Mode** to **General**:
  a) Run
     ```
     > asg dxl dist_mode set
     ```
  b) Select **General** (option 2)
Security

Generic Routing Encapsulation – GRE (asg_gre)

Description:
Generic Routing Encapsulation (GRE) is a tunneling protocol that can encapsulate a wide variety of network layer protocols inside virtual point-to-point links over an Internet Protocol internetwork.

Syntax:
# asg_gre load | stat | verify

To configure GRE, you will need to edit this configuration file:
$FWDIR/conf/gre_loader.conf

Tunnel configuration:
tunnel=<tunnel interface name> local_tun_addr=<local tunnel ip address>
remote_tun_addr=<remote tunnel ip address> phy_ifname=<physical interface name>
local_addr=<local physical address> remote_addr=<remote physical address>
ttl=<ttl>

Route configuration:
tunnel_route=<tunnel interface name> remote_tun_addr=<remote tunnel ip address> network=<network>

Configuration Example:
Configure tunnel interface with these parameters:
- Tunnel interface name: "GREtun"
- Local tunnel address 10.0.0.3
- Remote tunnel address 10.0.0.4
- Physical interface eth2-01
- Local address 40.40.40.1
- Remote address 40.40.40.2
- ttl 64
1. Use the following line:
tunnel=GREtun local_tun_addr=10.0.0.3 remote_tun_addr=10.0.0.4 phy_ifname=eth2-01 local_addr=40.40.40.1 remote_addr=40.40.40.2 ttl=64

2. To add route for 50.50.50.0/24 to go through the tunnel use the following line:
tunnel_route=GREtun remote_tun_addr=10.0.0.4 network=50.50.50.0/24

Note: All parameters are required
3. After editing the configuration file, use asg_gre to load it:
Output:

# asg_gre load
# asg_gre load
Copying configuration file to all blades... done
1_01:
Clearing existing GRE tunnels...
Loading GRE module... Done
Loading tunnel interface: GREtun
Loading route: 50.50.50.11/32 via 10.0.0.4 (GREtun)
Loading tunnel interface: GREtuA
Loading tunnel interface: GREtuB
Loading tunnel interface: GREtuC
Configuration loaded
1_02:
Clearing existing GRE tunnels...
Loading GRE module... Done
Loading tunnel interface: GREtun
Loading route: 50.50.50.11/32 via 10.0.0.4 (GREtun)
Loading tunnel interface: GREtuA
Loading tunnel interface: GREtuB
Loading tunnel interface: GREtuC
Configuration loaded
1_03:
Clearing existing GRE tunnels...
Loading GRE module... Done
Loading tunnel interface: GREtun
Loading route: 50.50.50.11/32 via 10.0.0.4 (GREtun)
Loading tunnel interface: GREtuA
Loading tunnel interface: GREtuB
Loading tunnel interface: GREtuC
Configuration loaded
1_04:
Clearing existing GRE tunnels...
Loading GRE module... Done
Loading tunnel interface: GREtun
Loading route: 50.50.50.11/32 via 10.0.0.4 (GREtun)
Loading tunnel interface: GREtuA
Loading tunnel interface: GREtuB
Loading tunnel interface: GREtuC
Configuration loaded

**Role Based Administration (RBA)**

**Description:**

The access to gclish features is controlled by Role Based Administration (RBA): each user is assigned with a role. Each role has a set of read-only features and read-write features. The user is not exposed to any features, other than the ones assigned to his role.


**Notes:**

- Extended commands have no read/write notion. When an extended command is added to a role (either as read or write), it can be executed by the users assigned to this role, regardless of its implications
- Each extended command should be separately added to role. Since asg command is the "entrance" to the 61000/41000 Security System, it usually needs to be added to all roles
- In order to allow user to run extended commands, its uid must be zero. This property is enforced when adding new users
- The user account information file located at /etc/passwd should not be edited by the user. RBA configuration should be performed only via gclish.
Example:
> add rba role myRole domain-type System readonly-features Chassis,interface readwrite-features route
> add user myUser uid 0 homedir /home/myUser
> set user myUser password
> add rba user myUser roles myRole
> show rba role myRole

**RADIUS Authentication**

**Description**

RADIUS (Remote Authentication Dial-In User Service) is a client/server authentication system that supports remote-access applications. User profiles are kept in a central database on a RADIUS authentication server. Client computers or applications connect to the RADIUS server to authenticate users.

You can configure the 61000/41000 Security System to work as a RADIUS client. The 61000/41000 Security System does not include RADIUS server functionality. You can configure the 61000/41000 Security System to authenticate users even when they are not defined locally. See Configuring Non-local RADIUS Users.

You can configure your 61000/41000 Security System computer to connect to multiple RADIUS servers. If the first server in the list is unavailable, the next RADIUS server in the priority list connects. You can delete a server at all times.

**To set the 61000/41000 Security System as a Radius client**

Use the `aaa radius-servers` commands to add, configure, and delete Radius authentication servers.

**To configure RADIUS for use in a single authentication profile:**

```
add aaa radius-servers priority VALUE host VALUE [ port VALUE ] prompt-secret secret
add aaa radius-servers priority VALUE host VALUE [ port VALUE ] secret VALUE
```

Example: Adding a new radius server 1.1.1.1 which listens on port 1812

```
add aaa radius-servers priority 1 host 1.1.1.1 port 1812 prompt-secret timeout 3
```

**To delete a RADIUS configuration:**

```
delete aaa radius-servers priority VALUE
```

**To change the configuration of a RADIUS entry:**

```
set aaa radius-servers priority VALUE host VALUE
set aaa radius-servers priority VALUE new-priority VALUE
set aaa radius-servers priority VALUE port VALUE
set aaa radius-servers priority VALUE prompt-secret
set aaa radius-servers priority VALUE secret VALUE
set aaa radius-servers priority VALUE timeout VALUE
```

**Note:** the configuration is done according to the priority and not the server ID or name.

**To view a list of all servers associated with an authentication profile:**

```
show aaa radius-servers list
```

**To view the RADIUS server configuration:**

```
show aaa radius-servers priority VALUE host
show aaa radius-servers priority VALUE port
show aaa radius-servers priority VALUE timeout
```

**Parameters:**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>priority</td>
<td>RADIUS server priority as an integer between 0 and 999 (default=0). When there two or more RADIUS servers, Gaia connects to the server with the highest priority. Low numbers have the higher priority.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>new-priority</td>
<td>New RADIUS server priority as an integer between 0 and 999 (default=0). When there are two or more RADIUS servers, Gaia connects to the server with the highest priority. Low numbers have the higher priority.</td>
</tr>
<tr>
<td>host</td>
<td>RADIUS server IP address in dot-delimited format.</td>
</tr>
<tr>
<td>port</td>
<td>UDP port on the RADIUS server. This value must match the port as configured on the RADIUS server. Typically this 1812 (default) or 1645 (non-standard but a commonly used alternative).</td>
</tr>
<tr>
<td>prompt secret</td>
<td>Shared secret (password) text string. The system prompts you to enter the value.</td>
</tr>
<tr>
<td>timeout</td>
<td>The number of seconds to wait for the server to respond. The default value 3 seconds.</td>
</tr>
<tr>
<td>secret</td>
<td>The shared secret used to authenticate the RADIUS server and the local client. You must define this value on your RADIUS server.</td>
</tr>
</tbody>
</table>

**Note:** After RADIUS client configuration, any authentication request will be forwarded to the RADIUS server. As a result, every account that is configured locally should be configured on the RADIUS server as well.

**Configuring Non-local RADIUS Users**

In order to allow login with non-local user to the 61000/41000 Security System, you need to define a default role for all non-local users that are configured in the Radius server.

The default role can include a combination of administrative (read/write) access to some features, monitoring (read-only) access to other features, and no access to other features.

**Syntax:** to define default role for non-local users

```plaintext
add rba role radius-group-any domain-type System readonly-features <List> readwrite-features <List>
```

- `readonly-features <List>` - Comma separated list of Gaia features that have read only permissions in the specified role.
- `readwrite-features <List>` - Comma separated list of Gaia features that have read/write permissions in the specified role.

**Example:**

```plaintext
add rba role radius-group-any domain-type System readonly-features arp readwrite-features
```

**Verification:**

Authenticate to the 61000/41000 Security System with a non-local user:

```plaintext
MyLaptop > ssh my_radius_user@my_61k_server
```

Upon successful authentication, the user 'my_radius_user' will be assigned the role 'radius-group-any' granted all the privileges defined in the radius-group-any role

**Configuring Local Radius users (with specific role)**

You can configure users to have different roles by creating new users on the 61000/41000 Security System and assigning them the required role.
To create a new user

add user <Name> uid 0 homedir <Path>

**Example:** add a new user named "local"
add user local uid 0 homedir /home/local

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>user</td>
<td>Login name of the user.</td>
</tr>
<tr>
<td>homedir</td>
<td>Full path for the user home directory</td>
</tr>
</tbody>
</table>

Setting user password

It is recommended to leave the local user's password blank.

Setting user role

You can choose a role from any preexisting roles, or to create a new role and to provide it with custom permissions. The "Adding a new role" section that is present inside this document outlines the procedure required for creating a new role.

To assign a user to a role, run

add rba user <User> roles <Role>

**Example:** to assign user "local" to role "radius"

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>The user name to assign a role to.</td>
</tr>
<tr>
<td>Roles</td>
<td>The role to assign to the user.</td>
</tr>
</tbody>
</table>

To add a new role

add rba role <Name> domain-type System
readonly-features <List>
readwrite-features <List>

**Example:**
add rba role radius domain-type System
readonly-features Chassis,configuration
readwrite-features aaa-servers

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role</td>
<td>Determines the role’s name.</td>
</tr>
<tr>
<td>readonly-features</td>
<td>Comma separated list of features to grant read only permissions for.</td>
</tr>
<tr>
<td>readwrite-features</td>
<td>Comma separated list of features to grant read/write permissions for.</td>
</tr>
</tbody>
</table>

**VSX Provisioning**

VSX can be provisioned in 2 of the following methods:

1. Fresh installation (With SmartDashboard)
2. Reconfigure (Via ‘vsx_util reconfigure’ in management CLI)
   - **Note:** Before starting one of above methods, verify that the SMO is the only SGM in the security group.
3. After successful operation, additional SGMs can be added to the security group.

**Clean Installation**

This section shows you how to create a new VSX Gateway using the **VSX Gateway Wizard**. After you complete the VSX Gateway Wizard, you can configure the VSX Gateway definition with SmartDashboard. For example, you can add or delete interfaces, or configure existing interfaces to support VLANs.
Before starting, you must verify that the SMO is the only SGM in the group.

**To start the VSX Gateway wizard:**
1. Open SmartDashboard. If you are using Multi-Domain Server, open SmartDashboard from the Domain Management Server of the VSX Gateway.
2. From the **Network Objects** tree, right-click on **Check Point** and select **VSX > Gateway**. The **General Properties** page of the **VSX Gateway Wizard** opens.

**Configuring VSX Gateway General Properties**
The **General Properties** page contains basic identification properties for VSX Gateways.

- **VSX Gateway Name**: Unique, alphanumeric for the VSX Gateway. The name cannot contain spaces or special characters except the underscore.
- **VSX Gateway IP Address**: Management interface IP address.
- **VSX Gateway Version**: Select the VSX version installed on the VSX Gateway from the drop-down list.

**Selecting Virtual Systems Creation Templates**
The **Creation Templates** page lets you provision predefined, default topology and routing definitions to Virtual Systems. This makes sure Virtual Systems are consistent and makes the definition process faster. You always have the option to override the default creation template when you create or change a Virtual System.

The Creation Templates are:

- **Shared Interface** - Not supported for the 61000/41000 Security System.
- **Separate Interfaces**: Virtual Systems use their own separate internal and external interfaces. This template creates a Dedicated Management Interface (DMI) by default.
- **Custom Configuration**: Define Virtual System, Virtual Router, Virtual Switch, and Interface configurations.

For this example, choose **Custom configuration**.

**Establishing SIC Trust**
Initialize Secure Internal Communication trust between the VSX Gateway and the management server. The gateway and server cannot communicate without Trust.

**Initializing SIC Trust**
When you create a VSX Gateway, you must enter the Activation Key that you defined in the installation wizard **setup** program. Enter and confirm the activation key and then click **Initialize**. If you enter the correct activation key, the **Trust State** changes to **Trust established**.

**Troubleshooting SIC Trust Initialization Problems**
If SIC trust was not successfully established, click **Check SIC Status** to see the reason for the failure. The most common issues are an incorrect activation key and connectivity problems between the management server and the VSX Gateway.

Troubleshooting to resolve SIC initialization problems:
- Re-enter and re-confirm the activation key.
- Verify that the IP address defined in **General Properties** is correct.
- Ping the management server to verify connectivity. Resolve connectivity issues.
- From the VSX Gateway command line, use the **cpconfig** utility to re-initialize SIC. After this process completes, click **Reset** in the wizard and then re-enter the activation key.

For more about resolving SIC initialization, see sk65385.
Defining Physical Interfaces

In the VSX Gateway Interfaces window, you can define physical interfaces as VLAN trunks. The page shows the interfaces currently defined on the VSX Gateway.

To define an interface as a VLAN trunk, select VLAN Trunk for the interface.

You can define VLAN trunks later. For this example, choose Next.

Virtual Network Device Configuration

If you chose the Custom Configuration option, the Virtual Network Device Configuration window opens. The options in this window are not supported for the 61000/41000 Security System.

Click Next.

VSX Gateway Management

In the VSX Gateway Management window, define security policy rules that protect the VSX Gateway. This policy is installed automatically on the new VSX Gateway.

Note - This policy applies only to traffic destined for the VSX Gateway. Traffic destined for Virtual Systems, other virtual devices, external networks, and internal networks is not affected by this policy.

The security policy consists of predefined rules for these services:

- UDP - SNMP requests
- TCP - SSH traffic
- ICMP - Echo-request (ping)
- TCP - HTTPS traffic

To Modify the Gateway Security Policy

1. Allow: Select to pass traffic on the selected services. Clear this option to block traffic on this service. By default, all services are blocked.
   For example, to be able to ping the gateway from the management server, allow ICMP echo-request traffic.

2. Source: Click the arrow and select a Source Object from the list.
   The default value is *Any. Click New Source Object to define a new source.

You can modify the security policy rules that protect the VSX Gateway later.

Click Next.

Configuring the Gateway Security Policy

1. Allow: Select to pass traffic on the selected services. Clear this option to block traffic on this service. By default, all services are blocked.
   For example, to be able to ping the gateway from the management server, allow ICMP echo-request traffic.

2. Source: Click the arrow and select a Source Object from the list.
   The default value is *Any. Click New Source Object to define a new source.

Completing the VSX Wizard

Click Next to continue and then click Finish to complete the VSX Gateway wizard.

This may take several minutes to complete.

If the process ends unsuccessfully, click View Report to see the error messages.

After the VSX gateway has finished successfully, other SGMs can be added to security group.
Reconfigure (vsx_util reconfigure)

Description
Restores a VSX configuration to a newly installed gateway

Syntax
vsx_util reconfigure

Input
- VSX gateway name
- SIC activation key assigned to the Security Management Server or Domain Management Server
- Retype to confirm the SIC activation key

Notes
- This command is also useful for restoring a gateway or cluster member after a system failure.
- Execute the command and follow the instructions on the screen.
- A new gateway must have the same hardware specifications and configuration as its replacement and other cluster members. Most importantly, it must have the same number of interfaces (or more) and the same management IP address.
- The new or replacement machine must be a new installation. You cannot use a machine with a previous VSX configuration.

Network Management

Working with IPv6
IPv6 support is disabled by default. You must enable IPv6 support on the 61000/41000 Security System before you can configure IPv6 addresses and static routes.

To prepare your 61000/41000 Security System to work with IPv6:
1. Enable IPv6 support.
3. Create IPv6 objects in SmartDashboard.
5. Reboot all SGMs.

Enabling/Disabling IPv6 Support (ipv6-state)
You use the ipv6-state command to:
- Enable IPv6 support for the all SGMs in the 61000/41000 Security System.
- Disable IPv6 support for the all SGMs in the 61000/41000 Security System.
- Show the IPv6 support status for all SGMs in the 61000/41000 Security System.

To complete the configuration you must reboot all SGMs at the same time. If you have a Chassis High Availability environment, you can enable IPv6 and reboot the SGMs one Chassis at a time. This feature makes it possible for network traffic to continue during configuration procedure.

Syntax
set ipv6-state [on|off]
show ipv6-state
### Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| on/off    | on = Enable IPv6 support  

### To Enable IPv6 Support on a single Chassis system:

1. Log into the 61000/41000 Security System.
2. Run:
   > set ipv6-state on
3. Run reboot -b all
   This reboots all SGMs.
4. Do the instructions on the screen.
5. Run > show ipv6-state
   Make sure that IPv6 is enabled for all SGMs.

### To Enable IPv6 on a dual Chassis System:

This procedure lets you reboot one Chassis at a time to prevent unnecessary downtime.

1. Log into the 61000/41000 Security System.
2. Run:
   > set ipv6-state on
3. Reboot all SGMs on the Standby Chassis:
   Run:
   > reboot -b <standby_chassis_name>
4. When the reboot completes, run this command on the Active Chassis:
   > asg chassis_admin -c <active_chassis_id> down
   This causes the Active Chassis to fail over to the Standby. The failover closes all active connections, which must be re-established.
5. Reboot all SGMs on the newly designated Standby Chassis:
   Run:
   > reboot -b <new_standby_chassis_name>

### Configuring IPv6 Static Routes - CLI (ipv6 static-route)

This section includes a complete command reference for the `ipv6 static-route` command. You can only use the `set` operation with this command, even when adding or deleting a static route.

**Description**  
Add, change or delete an IPv4 static route.

**Syntax**

```plaintext
set ipv6 static-route <Destination>  
   nexthop gateway <gw_ip>  
       [priority <p_value>] on|off  
       interface <gw_if> [priority <p_value>] on|off  
   nexthop blackhole  
   nexthop reject  
off
```
<table>
<thead>
<tr>
<th><strong>Parameter</strong></th>
<th><strong>Definition</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>nexthop</td>
<td>Defines the next hop path.</td>
</tr>
<tr>
<td>on</td>
<td>Enables the specified route or next hop.</td>
</tr>
<tr>
<td>off</td>
<td>Deletes the specified route or next hop. If you specify a next hop, only the specified path is deleted. If no next hop is specified, the route and all related paths are deleted.</td>
</tr>
<tr>
<td>gateway</td>
<td>Accepts and sends packets to the specified destination.</td>
</tr>
<tr>
<td>blackhole</td>
<td>Drops packets, but does not send an error message.</td>
</tr>
<tr>
<td>reject</td>
<td>Drops packets and sends an error message to the traffic source.</td>
</tr>
<tr>
<td>interface</td>
<td>Identifies the next hop gateway by the interface that connects to it. Use this option only if the next hop gateway has an unnumbered interface.</td>
</tr>
<tr>
<td>priority</td>
<td>Assigns a path priority when there are many different paths. The available path with the lowest priority value is selected. The gateway with the lowest priority value is selected.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Value</strong></th>
<th><strong>Definition</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;Destination&gt;</td>
<td>Destination IP address.</td>
</tr>
</tbody>
</table>
| <Route Type>  | gateway - Accepts and sends packets to the specified destination  
                reject - Drops packets and sends an error message to the traffic source  
                blackhole - Drops packets, but does not send an error message- |
| <gw_ip>       | Identifies the next hop gateway by its IP address. |
| <gw_if>       | Identifies the next hop gateway by the interface that connects to it. Use this option only if the next hop gateway has an unnumbered interface. |
| <p_value>     | Integer value between 1 and 8 (default=1). |

**Examples**

```
set ipv6 static-route 3100:192::0/64 nexthop 3900:172::1 priority 2 on

set ipv6 static-route 3100:192::0/64 nexthop 3900:172::1 interface eth3 priority 2 on

set ipv6 static-route 3100:192::0/64 nexthop off
set ipv6 static-route 3300:123::0/64 nexthop blackhole
```

**Notes**

There are no add or show commands for the static route feature.

**CLI Procedures - IPv6 Static Routes**

This section includes some basic procedures for managing static routes using the CLI.
To show IPv6 static routes, run

```
show ipv6 route static
```

Codes: C - Connected, S - Static, B - BGP, Rg - RIPng, A - Aggregate,
O - OSPFv3 IntraArea (IA - InterArea, E - External),
K - Kernel Remnant, H - Hidden, P - Suppressed

```
S     3100:55::1/64     is directly connected
S     3200::/64        is a blackhole route
S     3300:123::/64    is a blackhole route
S     3600:20:20:11::/64 is directly connected, eth3
```

To add an IPv6 static route, run:

```
set ipv6 static-route <Destination> nexthop gateway <gw_ip> on
set ipv6 static-route <Destination> nexthop gateway <gw_ip> interface <gw_if> on

Destination - Destination IPv6 address.
gw_ip - Next hop gateway IPv6 address.
gw_ip - Next hop gateway interface name.
```

Example:
```
set ipv6 static-route 3100:192::0/64 nexthop gateway 3900:172::1 on
set ipv6 static-route 3100:192::0/64 nexthop gateway 3900:172::1 interface eth3 on
```

To add an IPv6 static route with paths and priorities, run:

```
set static-route <Destination> nexthop gateway <gw_id> priority <P Value>
```

Destination - Destination IP address.
gw_ip - Next hop gateway IP address.
P Value - Integer between 1 and 8 (default =1)

Run this command for each path, assigning a priority value to each. You can define two or more paths using the same priority to specify a backup path with equal priority.

Example:
```
set ipv6 static-route 3100:192::0/64 nexthop gateway 3900:172::1 priority 3 on
```

To add an IPv6 static route where packets are dropped, run:

```
set ipv6 static-route <Destination> nexthop reject
set ipv6 static-route <Destination> nexthop blackhole
```

Destination - Destination IP address.
Reject - Drops packets and sends an error message to the traffic source.
Blackhole - Drops packets, but does not send an error message.

Examples:
```
set ipv6 static-route 3100:192::0/64 nexthop reject
or
set ipv6 static-route 3100:192::0/64 nexthop blackhole
```

To delete an IPv6 route and all related paths, run:

```
set ipv6 static-route <Destination> off
```

Destination - Destination IP address.

Example:
```
set ipv6 static-route 3100:192::0/64 off
```

To delete a path only, run:

```
set static-route <Destination> nexthop gateway <gw_ip> off
```

Destination - Destination IP address.
gw_ip - Next hop gateway IP address or interface name.

Example:
```
set ipv6 static-route 3100:192::0/64 nexthop gateway 3900:172::1 off
```
Configuring the 6in4 Internet Transition Mechanism

Use this command to move IPv6 traffic over a network that does not support IPv6. The command uses the 6in4 Internet transition protocol to encapsulate IPv6 traffic for IPv4 links.

To create 6in4 virtual interfaces, run these commands in this order:

- add interface <physical-if> 6in4 <6in4-id> remote <remote-ipv4-address> [ttl "ttl"]
- set interface <sit if name> ipv6-address <address> mask-length 64

Adding the Interface

Use this command to add the interface.

Syntax
add interface <physical_if> 6in4 <6in4_id> remote <remote_ipv4> [ttl "ttl"]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>physical-if</td>
<td>The physical interface encapsulated traffic will leave the system from, for example eth1-01.</td>
</tr>
<tr>
<td>6in4-id</td>
<td>A numerical identifier for the 6in4 Virtual Interface.</td>
</tr>
<tr>
<td>remote-ipv4-address</td>
<td>IPv4 address of the remote peer.</td>
</tr>
<tr>
<td>ttl</td>
<td>Time-to-live: the number of router hops before packets are discarded.</td>
</tr>
</tbody>
</table>

Example
> add interface eth1-01 6in4 999 remote 50.50.50.10
  1_01:
  Success

Notes

- Despite having specified a single physical interface (eth1-01) on the command line, the virtual (sit_6in4_) interface is created for eth1-01 on all SGMs.
- To see the virtual interfaces for each SGM, run: show interface eth1-01 6in4s.

Use this command to set the interface.

Syntax
set interface <sit if name> ipv6-address <address> mask-length 64

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sit if name</td>
<td>The name of the virtual interface, which begins: sit_6in4_&lt;ID_number given in previous command&gt;.</td>
</tr>
<tr>
<td>address</td>
<td>IPv6 address.</td>
</tr>
</tbody>
</table>

Example
> set interface sit_6in4_999 ipv6-address 30:30:30::1 mask-length 64
  1_01:
  Success
Setting the Interface

Example

> set interface sit_6in4_999 ipv6-address 30:30:30::1 mask-length 64
1_01:
Success

Deleting the 6in4 Virtual Interface

Run: delete interface <physical-if> 6in4 <6in4-id>. For example:

> delete interface eth1-01 6in4 999
1_01:
success

Asg Search and 6in4

- When using the asg search command to discover which SGM handles a specific connection (actively or as backup) and which Chassis, IPv4 addresses of a remote peer may show as being handled by more than 1 SGM.

- asg search run on IPv6 addresses show:
  - 1 SGM on the active Chassis
  - 1 SGM on the standby Chassis

Working with the Bridge Mode

Check Point security devices support bridge interfaces that implement native, Layer-2 bridging. Configuring an interface as a bridge lets network administrators deploy security devices in an existing topology without reconfiguring the existing IP routing scheme. This is an important advantage for large-scale, complex environments. Gaia does not support Spanning Tree Protocol (STP) bridges.

You configure Ethernet interfaces (including aggregated interfaces) on your Check Point security device to work like ports on a physical bridge. The interfaces then send traffic using Layer-2 addressing. You can configure some interfaces as bridge interfaces, while other interfaces on the same device work as layer-3 devices. Traffic between bridge interfaces is inspected at Layer-2. Traffic between two Layer-3 interfaces, or between a bridge interface and a Layer-3 interface is inspected at Layer-3.

Working with Chassis HA in the Bridge mode

A Dual Chassis 61000/41000 Security System deployment always works in the Active/Standby mode. Only the Active Chassis handles traffic. The 61000/41000 Security System maintains a MAC shadow table that caches MAC addresses handled by the system. When Chassis failover occurs, the new Active Chassis generates advertisement packets with the cached MAC addresses. This lets remote switches "learn" the MAC address, and start to handle STP bridge traffic.

Using the SSM60 in the Bridge Mode

To use the SSM60, with the Bridge mode:

1. Run:
   
   ```
   # g_update_conf_file simkern.conf bridge_mode_on_ssm60=1
   ```

2. Reboot the system.

Using the Bridge mode with VLAN Trunks

We recommend that you enable the VLAN performance enhancement feature when a Bridge interface handles VLAN trunks. To enable VLAN performance enhancement, run this command in the Expert mode:

```
# g_vlan_perf_enhancement -s
```

Distribution mode

The Bridge mode only supports the **General Distribution** mode.
Active/Active Bridge mode

By default the Active/Active Bridge Mode does not support asymmetric traffic between Chassis. When asymmetric traffic is enabled:

- Client-to-server traffic is handled by Chassis1.
- Server-to-client traffic is handled by Chassis2.

To enable asymmetric traffic:

1. Run (in the Expert mode):
   ```
   # g_update_conf_file $FWDIR/modules/fwkern.conf
   fwha_both_chassis_pass_traffic=1
   ```
2. Run:
   ```
   # g_fw ctl get int fwha_both_chassis_pass_traffic 1
   ```

Note: The `fwha_both_chassis_pass_traffic` command can decrease performance.

Bridge Mode Limitations

- Bridge Mode is only supported with 2 interfaces
- IPv6 is not supported

Configuring Bridge Interfaces

Use these commands to work with Bridge interfaces.

Syntax

```plaintext
add bridging group <group_id> [interface <if_name>]
delete bridging group <group_id> interface <if_name>
show bridging group <group_id> interface <if_name>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;group_id&gt;</code></td>
<td>Integer that identifies the bridging group</td>
</tr>
<tr>
<td><code>&lt;if_name&gt;</code></td>
<td>Interface name as configured on the system</td>
</tr>
</tbody>
</table>

Examples

```plaintext
> add bridging group 2 interface eth3

> show bridging group 2
Bridge Configuration
   Bridge Interfaces
      eth3
```

To use vlan interfaces in a bridging group:

1. Run this command to add the vlan to the physical interface:
   ```plaintext
   > Add interface <if_name> vlan <vlan_id>
   ```
2. Run this command to add the interface vlan to the bridging group:
   ```plaintext
   > Add bridging group <group_id> interface <if_name>,<vlan_id>
   ```

Note: All of the specified parameters are required to do these tasks.

Disabling BPDU Forwarding

Bridge Protocol Data Unit (BPDU) is a data message that is sent between switches in an extended LAN that uses a Spanning Tree Protocol (STP) topology. When VLAN translation is configured, BPDU frames can send the incorrect VLAN number to switch ports through the bridge. This mismatch can cause the switch port to block traffic.

To resolve this issue, it is necessary to disable BPDU forwarding in a manner that survives reboot. This solution also works well for layer 2 Virtual Systems.
To permanently disable BPDU forwarding:
2. Search for /etc/init.d/functions.
3. Add this new line after the above line:
   ```bash
   /sbin/sysctl -w net.bridge.bpdu_forwarding=0
   ```
4. Exit the editor and save the file.
5. Copy the file to all SGMs with this command:
   ```
   > asg_cp2blades /etc/rc.d/init.d/network
   ```
6. Reboot the system.
   If you are using a dual Chassis 61000/41000 Security System, reboot the Standby Chassis first and then reboot the Active Chassis.

To learn more, see sk98927 (http://supportcontent.checkpoint.com/solutions?id=sk98927).

**Configuring Link Aggregation (Bonding)**

Link aggregation combines multiple physical interfaces into a virtual interface called a bond. Bonded interfaces (known as slaves) add redundancy to a connection as well as increasing the connections throughput to a level beyond what is possible using a single physical interface.

To create an interface bond you need to run these commands in this order from the gclich shell:

<table>
<thead>
<tr>
<th>Commands in Running Order:</th>
<th>Purpose:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add bonding group &lt;bond_id&gt;</td>
<td>Creates a bonding group</td>
</tr>
<tr>
<td>set bonding group &lt;bond_id&gt; mode &lt;bond_mode&gt;</td>
<td>Sets a bonding mode: 802.3ad (LAPC) or XOR</td>
</tr>
<tr>
<td>set interface &lt;if_name&gt; state on</td>
<td>Sets the slave interface to on</td>
</tr>
<tr>
<td>add bonding group &lt;bond_id&gt; interface &lt;if_name&gt;</td>
<td>Enslaves interfaces to the bond</td>
</tr>
</tbody>
</table>

**Note** - Before running the link aggregation commands, make sure that the slave interfaces do not have an IP Address already assigned.

**Creating a Bonding Group.**

Use this command to create a bonding group. A bonding group is a single virtual interface or bond. A bond can contain multiple Slaves.

**Note:** the <bond_id> must be a number. The bond name is created automatically with the bond id. For example, entering 4 for the bond id creates a virtual interface named bond4.

**Syntax**

```
add bonding group <bond_id>
```

**Example**

```
> add bonding group 4
```

**Output**

```bash
1_01: success
1_02: success
1_03: success
2_01: success
2_03: success
>```
Running the command creates one virtual interface (bond4) that includes all SGM interfaces on each Chassis.

**Setting a Bonding Mode**

**Description**
Use this command to set a bonding mode.

The following Bond modes are supported in the 61000/41000 Security System:

- **8023AD (LACP):** Do dynamic bonding according to the IEEE 802.3ad protocol
- **Active/Backup:** Bond build up from one interface Active while other interface is in standby. When the active interface encounters a problem failover occurs to other Bond interface.
- **XOR:** Do load sharing based on layer2, or 3 and 4.

*Note:* round-robin mode is not supported on the 61000/41000 Security System.

**Syntax**

```bash
set bonding group <bond_id> mode <bond_MODE>
```

**Example**

```bash
set bonding group 4 mode 8023A
```

**Output**

```
1_01: success
1_02: success
1_03: success
2_01: success
2_03: success
>
```

**Explanation**
Physical interfaces enslaved to bond4 do load sharing according to the 802.3ad protocol
Setting a Polling interval
Use this command to set the polling interval.

Syntax
`set bonding group <bond_id> mii-interval 100`

Explanation
The polling interval is how often (in milliseconds) the OS checks to see if the bond is up.

Setting the Slave Interface to On

Description
Use this command to switch the interface on or off.

Note: Run this command from the Bash shell.

Syntax
`set interface <Interface_name> state on`

Example
`set interface eth1-02 state on`

Enslaving Interfaces
Use this command to enslave a physical interface to a named bond.

Syntax
`add bonding group <bond_id> interface <Interface_name>`

Example
`add bonding group 4 interface eth1-02`

Explanation
Adds interface eth1-02 to bond4

Removing Slaves from a Bond
To remove a slave interface from a bond run:

Syntax
`delete bonding group <bond_id> interface <interface_name>`

Example
`delete bonding group 1 interface eth1-02`

Note - There is no command to delete all slave interfaces at the same time.

Deleting a Bonding Group
To delete a bonding group you must first delete all slaves one by one. Then run:

Syntax
`delete bonding group <bond_id>`

Example
`delete bonding group 4`

Explanation
This command deletes bond4
Configuring VLANs

Description
Use this command to configure VLANs.

Syntax

add interface <interface> vlan <vlan-id>

set interface <interface>.<vlan-id> ip-address <ip-address> mask-length <mask-len>

delete interface <interface> vlan <vlan-id>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface</td>
<td>The name of the interface</td>
</tr>
<tr>
<td>vlan</td>
<td>Vlan ID number</td>
</tr>
<tr>
<td>mask-length</td>
<td>Network mask length</td>
</tr>
</tbody>
</table>

Example 1
add interface eth2-03 vlan 444

Output
> add interface eth2-03 vlan 444
  1_01:
  success

Example 2
set interface eth2-03.444 ipv4-address 30.30.30.1 mask-length 24

Output
> set interface eth2-03.444 ipv4-address 30.30.30.1 mask-length 24
  1_01:
  success

Example 3
show interface eth2-03 vlans

Output
> show interface eth2-03 vlans
  1_01:
  eth2-03.444

Notes
The output shows VLAN interfaces on physical interface eth2-03.

Example 4
delete interface eth2-03 vlan 444

Output
> delete interface eth2-03 vlan 444
  1_01:
  success
Configuring Dynamic Routing - Unicast

To decrease the administrative and operational overhead caused by static routes, the 61000/41000 Security System supports dynamic routing protocols OSPF and BGP to:

- Collect routing data for remote networks
- Automatically add routing data to the system's routing table
- Advertise destinations to other routers in the network
- Calculate the best path to each network
- Dynamically learn changes in routing topology

**Configuring OSPF on an Interface**

Use this command to enable the OSPF protocol on a specified interface. The ROUTED daemon listens and sends OSPF messages on this interface only.

**Note** - Before you configure an OSPF interface, you must first run:

```bash
> set router-id <ip_address>.
```

For example, to configure OSPF on interface eth1-01 (IP 40.40.40.1), run:

```bash
> set router-id 40.40.40.1
```


**Syntax**

```bash
set ospf interface <if_name> <area> [on|off]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;if_name&gt;</td>
<td>Name of the interface to be used for OSPF.</td>
</tr>
<tr>
<td>&lt;area&gt;</td>
<td>Specify one of these area values*</td>
</tr>
<tr>
<td></td>
<td>- An IPv4 address</td>
</tr>
<tr>
<td></td>
<td>- An integer value between 1 and 4294967295</td>
</tr>
<tr>
<td></td>
<td>- backbone</td>
</tr>
<tr>
<td>[on</td>
<td>off]</td>
</tr>
</tbody>
</table>

**Example**

```bash
> set ospf interface eth1-01 area backbone on
```

**Notes**

- To verify that the interface has OSPF enabled, run:
  ```bash
  > show ospf interfaces
  ```
- To show the OSPF state in relation to its neighbors, run:
  ```bash
  > show ospf neighbors
  ```
- To show OSPF statistics, run:
  ```bash
  > show ospf summary
  ```

**Configuring BGP**

To configure BGP:

- Set the ID of the Autonomous System
- Set at least one BGP neighbor
Defining the Autonomous System (AS)

Use this command to set the Autonomous System ID

Syntax

set as <as_id>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;as_id&gt;</td>
<td>Autonomous System (AS) ID</td>
</tr>
</tbody>
</table>

Example

set as 2

Defining a BGP Neighbor

Use this command to define a BGP neighbor

Syntax

set bgp [internal | external] remote-as <as_id> peer [ip_address] [on|off]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>internal</td>
<td>Autonomous System (AS) type</td>
</tr>
<tr>
<td>&lt;as_id&gt;</td>
<td>Autonomous System (AS) ID</td>
</tr>
<tr>
<td>&lt;ip_address&gt;</td>
<td>Remote peer IP address</td>
</tr>
</tbody>
</table>

Examples

- set bgp external remote-as 24 on
  Adds AS 24 to the system's configuration

- set bgp external remote-as 24 local-address 40.40.40.24 on
  Configures the local system interface 40.40.40.24 as a BGP peer for AS 24.

Notes

To verify that BGP is running:

- To show BGP peers, run:
  show bgp peers

- To show BGP state, run:
  show bgp summary

To deactivate BGP:

- set bgp external remote-as 24 off

Changing the Default VMAC (asg_unique_mac_utility)

By default, all 61000/41000 Security Systems have the same VMAC address. This makes sure that there can only be one 61000/41000 Security System (Dual or Single Chassis) on the same Layer-2 network segment.

If it is necessary to have more than one 61000/41000 Security System on the same Layer-2 network segment

Use the asg_unique_mac_utility command to change the:

- Interface default VMAC to a unique value
- Host name

Note - Changing the unique VMAC address causes dropped connections and lost traffic.
Syntax: asg_unique_mac_utility

Output

<table>
<thead>
<tr>
<th>Unique MAC Utility</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOSTNAME: cpmodule</td>
</tr>
<tr>
<td>Unique MAC: 254</td>
</tr>
</tbody>
</table>

Choose one of the following options:

1) Set Hostname with Unique MAC wizard
2) Apply Unique MAC from current HOSTNAME
3) Manual set Unique MAC
4) Back to Unique MAC Factory default (254)
5) Exit

Explanation

Use this command if it is necessary to deploy more than one 61000/41000 Security System on the same network segment.

The menu has four options:

1) Set Hostname with Unique MAC wizard

Using this option you enter:

- A setup name
- A unique MAC setup number between 1-254.

The option adds the _asg suffix and setup number to the setup name. For example:

<table>
<thead>
<tr>
<th>Setup Name</th>
<th>Suffix</th>
<th>Setup number</th>
</tr>
</thead>
<tbody>
<tr>
<td>armgdn</td>
<td>_asg</td>
<td>22</td>
</tr>
</tbody>
</table>

This results in a new Hostname with a unique MAC value of 22 (16 in HEX):

<table>
<thead>
<tr>
<th>New Host Name</th>
<th>Unique MAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>armgdn_asg22</td>
<td>22</td>
</tr>
</tbody>
</table>

The setup number replaces the default Magic MAC value of 254. After running this option, all interfaces of type ethX-YZ have the a unique MAC value of 22 (16 in HEX)

2) Apply Unique MAC from current host name

Use this option to change the system's VMAC. The option automatically sets a new VMAC on the relevant interfaces. The new VMAC is derived from the setup number within the hostname. For this reason, the existing hostname must first comply with the setup name/ asg suffix / setup number convention.

3) Manual Set Unique MAC

Use this option to change the unique MAC according to your own input without changing the host name. value. The existing host name does not have to comply with the setup name / asg suffix / setup number convention.

Note: Manually setting the unique MAC without changing the host name can lead to confusion when number of 61000/41000 Security System exist on the same network segment.

4) Revert to Unique MAC Factory Default

Use this option to set the unique MAC value to its default value (254)
Verifying the New MAC Address

Use these commands to make sure that the unique MAC value has changed:

- For the unique MAC database value, run (from the bash shell):
  ```
  g_allc dbget chassis:private:magic_mac
  # # g_allc dbget chassis:private:magic_mac
  -*- 4 sgms: 1_01 1_02 2_02 2_03 -*-
  22
  ```

- For the unique MAC Kernel value, run (from gclish):
  ```
  > fw ctl get int fwha_mac_magic
  -*- 4 sgms: 1_01 1_02 2_02 2_03 -*-
  fwha_magic_magic = 22
  ```

You can also display the magic attribute within type ethX-YZ interfaces by using the `ifconfig` command:

```
# ifconfig eth1
eth1-01   Link encap:Ethernet  HWaddr 00:1C:7F:81:01:16
inet6 addr: fe80::21c:7fff:fe81:116/64 Scope:Link
UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500 Metric:1
RX packets:154820 errors:0 dropped:0 overruns:0 frame:0
TX packets:23134 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0 RX bytes:15965660 (15.2 MiB)
TX bytes:2003398 (1.9 MiB)
```

Changing the Management Interface

Use this command to change the management interface for the SGMs.

- **Note** - This procedure is applicable for Security Gateway environments only. Management interface change are not supported for VSX.

To change the management interface:
1. Make sure that the management interface cable is connected to the network.
2. Run these commands in order:
   a) `set management interface <new_management_interface>`
   b) `delete interface <old_management_interface> ipv4-address`
   c) `set interface <new_management_interface> ipv4-address <ip>`
      `mask-length <length>`
   d) `set interface <new_management_interface> state on`
3. In SmartDashboard, get the new topology for the 61000/41000 Security System object.
4. Install policy.

**Parameters for these commands**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;new_management_interface&gt;</code></td>
<td>Interface name of the new management interface. For example: eth1-Mgmt3</td>
</tr>
<tr>
<td><code>&lt;old_management_interface&gt;</code></td>
<td>Interface name of the existing management interface that is to be changed or deleted. For example: eth1-Mgmt2.</td>
</tr>
<tr>
<td><code>ipv4-address &lt;ip&gt;</code></td>
<td>Interface IPv4 address</td>
</tr>
</tbody>
</table>
Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Commands</th>
</tr>
</thead>
<tbody>
<tr>
<td>mask-length &lt;length&gt;</td>
<td>Interface net mask</td>
</tr>
<tr>
<td>state</td>
<td>Interface state(on/off)</td>
</tr>
</tbody>
</table>

Configuring Policy Based Routing

This release supports Source Based Routing and all other Policy Based Routing features. These features are documented in the Policy Based Routing chapter of the R76 Gaia Advanced Routing Administration Guide (http://supportcontent.checkpoint.com/documentation_download?ID=22929). Use the set pbr command and procedure in the Configuring Policy Based Routing - CLI section.

ECMP Configuration

Description

Equal-cost multi-path routing (ECMP) is a routing strategy where you manually define a static route to a number of next-hop gateways. It potentially offers substantial increases in bandwidth by load-balancing traffic over multiple paths to reach the destination network defined in the static route.

Syntax

set static-route <network> nexthop gateway address <gw ip address> on

Parameter | Description
--- | ---
<network> | The IP address of the destination network
<gw ip address> | The IP address of the next-hop gateway

Example

set static-route 50.50.50.0/24 nexthop gateway address 20.20.20.101 on
data static-route 50.50.50.0/24 nexthop gateway address 20.20.20.102 on
set static-route 50.50.50.0/24 nexthop gateway address 20.20.20.103 on

Notes

To reach addresses on the 50.50.50.0/24 network, packets must first be forwarded to one of these gateways:

- 20.20.20.101
- 20.20.20.102
- 20.20.20.103

To make sure static routes to the next-hop gateways are being enforced, run:

> show route static

1_01:
Codes: C - Connected, S - Static, R - RIP, B - BGP, O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA) A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed

S 0.0.0.0/0 via 192.168.33.1, eth2-01, cost 0, age 2092
5.5.5.0/24 via 20.20.20.101, eth1-01, cost 0, age 322
via 20.20.20.102, eth1-01
via 20.20.20.103, eth1-01

The output shows that the static route to 50.50.50.0/24 is via three next-hop gateways.
Disabling ECMP

ECMP is enabled by default. To disable it:

1. Open this file for editing:
   
   ```
   $PPKDIR/boot/modules/simkern.conf
   
   If simkern.conf does not exist, create it.
   ```

2. Add this line:
   
   ```
   sim_routing_by_source=0
   ```

3. Save the file and reboot.

**Enhanced Failover of ECMP Static Routes**

**Description**

The enhanced routing features automatically start failover on detection of unreachable next hop gateways for ECMP static routes. It ensures that the required destination will be routed only from reachable next-hops by deleting unreachable next-hops from the routing table, and add it again when they are reachable.

The new functionality probes each next hop gateway of a static route to detect its reachability status. Probing is done on each SGM, with "ping", the standard ICMP echo protocol. If the next hop is unreachable it is being removed from the routing table and re-entered when it is detected as reachable.

**Syntax**

In order to activate enhanced failover on a static route run from gclish:

```
> set static-route <network>/<subnet length> ping on
```

Note: enhanced ECMP failover can be configured after you configured ECMP static route. (see Configuring)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;network&gt;</td>
<td>The IP address of the destination network</td>
</tr>
<tr>
<td>&lt;subnet length&gt;</td>
<td>The subnet length of the destination network</td>
</tr>
</tbody>
</table>

In order to adjust ping behavior, use:

```
> set ping count <VALUE>
> set ping interval <VALUE>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count &lt;VALUE&gt;</td>
<td>Number of packets to be sent before next hop is declared dead</td>
</tr>
<tr>
<td>Interval &lt;VALUE&gt;</td>
<td>Time in seconds to wait between two consecutive pings</td>
</tr>
</tbody>
</table>

**Example**

**Step 1: set ECMP for destination 5.5.5.0/24**

```
> set static-route 5.5.5.0/24 nexthop gateway address 10.33.85.2 on
> set static-route 5.5.5.0/24 nexthop gateway address 10.33.85.4 on
> set static-route 5.5.5.0/24 nexthop gateway address 10.33.85.100 on
> show route
```

1_01:

Codes: C - Connected, S - Static, R - RIP, B - BGP, O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)

A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed

<table>
<thead>
<tr>
<th>S</th>
<th>via (IP Address), cost, age</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.0/0</td>
<td>192.168.33.1, eth2-01, cost 0, age 2092</td>
</tr>
<tr>
<td>5.5.5.0/24</td>
<td>10.33.85.2, eth1-01, cost 0, age 322</td>
</tr>
<tr>
<td></td>
<td>10.33.85.4, eth1-01</td>
</tr>
<tr>
<td></td>
<td>10.33.85.100, eth1-01</td>
</tr>
</tbody>
</table>

**Step 2: enable failover ECMP on all static route configured for destination 5.5.5.0/24**

```
> set static-route 5.5.5.0/24 ping on
```
Step3: validation

When next-hop 10.33.85.2 is unreachable: (no ICMP replies), after 3 pings (by default) it will be removed from the routing table:

[Expert@CH_Lena-ch02-01]# tcpdump -nepi eth1-01 host 10.33.85.2

tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1-01, link-type EN10MB (Ethernet), capture size 96 bytes
14:40:48.388032 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800), length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28
14:40:58.388425 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800), length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28
14:41:08.387895 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800), length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28

The route has been deleted from the routing table

01 > show route
1_01:
Codes: C - Connected, S - Static, R - RIP, B - BGP,
O - OSPF IntraArea (IA - InterArea, E - External, N - NSSA)
A - Aggregate, K - Kernel Remnant, H - Hidden, P - Suppressed

0.0.0.0/0           via 192.168.33.1, eth2-01, cost 0, age 2511
S   5.5.5.0/24       via 10.33.85.4, eth1-01, cost 0, age 52
                  via 10.33.85.100, eth1-01

When 10.33.85.2 is reachable again we can see in the tcpdump that it replies to ping requests and it is added to the routing table
[Expert@CH_Lena-ch02-01]# tcpdump -nepi eth1-01 host 10.33.85.2
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1-01, link-type EN10MB (Ethernet), capture size 96 bytes
14:38:08.388224 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800),
length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28
14:38:08.388462 00:50:fc:58:80:0a > 00:1c:7f:0f:00:fe, ethertype IPv4 (0x0800), length 62: 10.33.85.2 > 10.33.85.1: ICMP echo reply, id 53007, seq 43981, length 28
14:38:18.387762 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800), length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28
14:38:18.387980 00:50:fc:58:80:0a > 00:1c:7f:0f:00:fe, ethertype IPv4 (0x0800), length 62: 10.33.85.2 > 10.33.85.1: ICMP echo reply, id 53007, seq 43981, length 28
14:38:28.388161 00:1c:7f:a1:01:55 > 00:50:56:a7:7f:f5, ethertype IPv4 (0x0800), length 62: 10.33.85.1 > 10.33.85.2: ICMP echo request, id 53007, seq 43981, length 28
14:38:28.388382 00:50:fc:58:80:0a > 00:1c:7f:0f:00:fe, ethertype IPv4 (0x0800), length 62: 10.33.85.2 > 10.33.85.1: ICMP echo reply, id 53007, seq 43981, length 28

> show route
1_01:
Codes: C – Connected, S – Static, R – RIP, B – BGP,
O – OSPF IntraArea (IA – InterArea, E – External, N – NSSA)
A – Aggregate, K – Kernel Remnant, H – Hidden, P – Suppressed
S 0.0.0.0/0    via 192.168.33.1, eth2-01, cost 0, age 2092
5.5.5.0/24    via 10.33.85.2, eth1-01, cost 0, age 322
            via 10.33.85.4, eth1-01
            via 10.33.85.100, eth1-01

Validation
1. Run from gclish:
   show route and verify that only ECMP static routes with reachable next-hops
   are shown
2. Run:
   tcpdump to verify that each few seconds there is a ping request on the
   interface with static route and ping on

Working with the ARP Table (asg_arp)

Description
This command shows the ARP cache for the whole 61000/41000 Security System or for the specified:
- SGMs
- Interface
- MAC address
- Host Name

You can show summary or detailed (verbose) information. You can also run MAC address verification on
both Chassis.
**Syntax**

```
asg_arp -h
asg_arp [-b <sgm_ids>] [-v] [--verify] [-i <if>] [-m <mac>] [<hostname>]
```

```
asg_arp --legacy
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-h</td>
<td>Shows command syntax and help information</td>
</tr>
<tr>
<td>-v</td>
<td>Verbose - Shows detailed SGM cache information</td>
</tr>
<tr>
<td>-b &lt;sgm_ids&gt;</td>
<td>Works with SGMs and/or Chassis as specified by &lt;sgm_ids&gt;.</td>
</tr>
<tr>
<td></td>
<td>The &lt;sgm_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>• No &lt;sgm_ids&gt; specified or all shows all SGMs and Chassis</td>
</tr>
<tr>
<td></td>
<td>• One SGM</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of SGMs (1_1,1_4)</td>
</tr>
<tr>
<td></td>
<td>• A range of SGMs (1_1-1_4)</td>
</tr>
<tr>
<td></td>
<td>• One Chassis (Chassis1 or Chassis2)</td>
</tr>
<tr>
<td></td>
<td>• The active Chassis (chassis_active)</td>
</tr>
<tr>
<td>-i &lt;if&gt;</td>
<td>Shows the ARP cache for the specified interface</td>
</tr>
<tr>
<td>-m</td>
<td>Shows the ARP cache for the specified MAC address</td>
</tr>
<tr>
<td>&lt;hostname&gt;</td>
<td>Shows the ARP cache for the specified host name</td>
</tr>
<tr>
<td>--verify</td>
<td>Run MAC address verification on both Chassis and show the results</td>
</tr>
<tr>
<td>--legacy</td>
<td>Shows the ARP cache per SGM in the legacy format</td>
</tr>
</tbody>
</table>
### Verbose Mode Output

This example shows the ARP cash in the detailed (verbose) mode for the active Chassis.

```
> asg_arp -v
Address      HWtype   HWaddress            Flags Mask  Iface        SGMs
172.23.9.198 ether 00:0C:29:87:AF:15    C           eth1-Mgmt1  1_1, 1_3, 1_4, 1_5
192.0.2.5    ether 00:1C:7F:05:04:FE    C           Sync         1_1, 1_3, 1_4
172.23.9.4    ether 00:17:65:3C:30:43    C           eth1-Mgmt1  1_1
192.0.2.3    ether 00:1C:7F:03:04:FE    C           Sync         1_1, 1_5
192.0.2.4    ether 00:1C:7F:04:04:FE    C           Sync         1_1, 1_3, 1_5
192.0.2.1    ether 00:1C:7F:01:04:FE    C           Sync         1_3, 1_4, 1_5
24.24.24.1    ether 00:04:23:C0:0E:98    C           eth2-01      1_3, 1_5
14.14.14.3    ether 00:04:23:C0:0F:5B    C           eth1-01      1_3, 1_5
198.51.100.32 ether 00:A0:12:99:E6:22    C           eth1-CIN     1_5
198.51.100.232 ether 00:A0:12:99:65:E2    C           eth2-CIN     1_5
198.51.100.33 ether 00:18:49:01:B3:82    C           eth1-CIN     1_5
```

### Verifying MAC Addresses

This example shows the output of the MAC address verification on the active Chassis.

```
> asg_arp --verify
Address      HWtype   HWaddress            Flags Mask  Iface        SGMs
172.23.9.4 ether 00:17:65:3C:30:43    C           eth1-Mgmt4  2_02
192.0.2.16 ether 00:1C:7F:10:04:FE    C           Sync         2_03,2_04
192.0.2.17 ether 00:1C:7F:11:04:FE    C           Sync         2_02,2_04
192.0.2.18 ether 00:1C:7F:12:04:FE    C           Sync         2_02,2_03
cmm         ether 00:18:49:01:60:89    C           eth1-CIN     2_02
ssm1        ether 00:A0:12:A4:63:41    C           eth1-CIN     2_02
ssm2        .        (incomplete)         .           eth2-CIN     2_02

Starting mac address verification on local chassis... (Chassis 2)
No inconsistency found on local chassis
Legacy Mode Output

This example shows the legacy mode output, per SGM.

```bash
> asg_arp --legacy
1_01:
Address  HWtype  HWaddress    Flags  Mask        Iface
172.23.9.198  ether  00:0C:29:87:AF:15  C     eth1-Mgmt1
192.0.2.5     ether  00:1C:7F:05:04:FE  C     Sync
172.23.9.4     ether  00:17:65:3C:30:43  C     eth1-Mgmt1
192.0.2.3     ether  00:1C:7F:03:04:FE  C     Sync
192.0.2.4     ether  00:1C:7F:04:04:FE  C     Sync
1_03:
Address  HWtype  HWaddress    Flags  Mask        Iface
192.0.2.5     ether  00:1C:7F:05:04:FE  C     Sync
24.24.24.1    ether  00:04:23:C0:0E:98  C     eth2-01
192.0.2.4     ether  00:1C:7F:04:04:FE  C     Sync
192.0.2.1     ether  00:1C:7F:01:04:FE  C     Sync
172.23.9.198   ether  00:0C:29:87:AF:15  C     eth1-Mgmt1
14.14.14.3    ether  00:04:23:C0:0F:5B  C     eth1-01
1_04:
Address  HWtype  HWaddress    Flags  Mask        Iface
192.0.2.1     ether  00:1C:7F:01:04:FE  C     Sync
172.23.9.198   ether  00:0C:29:87:AF:15  C     eth1-Mgmt1
192.0.2.5     ether  00:1C:7F:05:04:FE  C     Sync
1_05:
Address  HWtype  HWaddress    Flags  Mask        Iface
ssm1         ether  00:A0:12:99:E6:22  C     eth1-CIN
192.0.2.3     ether  00:1C:7F:03:04:FE  C     Sync
172.23.9.198   ether  00:0C:29:87:AF:15  C     eth1-Mgmt1
14.14.14.3    ether  00:04:23:C0:0F:5B  C     eth1-01
192.0.2.4     ether  00:1C:7F:04:04:FE  C     Sync
ssm2         ether  00:A0:12:99:65:62  C     eth2-CIN
192.0.2.1     ether  00:1C:7F:01:04:FE  C     Sync
cmm           ether  00:18:49:01:B3:82  C     eth1-CIN
24.24.24.1    ether  00:04:23:C0:0E:98  C     eth2-01
```

Proxy ARP for Manual NAT – (local.arp file)

Proxy ARP is a mechanism that allows the configuration of a Gateway to respond to ARP requests on behalf of other hosts. For a complete documentation regarding Proxy ARP configuration please refer to sk30197.

To configure the proxy ARP mechanism on the 61000/41000 Security System:

1. Add any IPs for which the 61000/41000 Security System should answer to ARP requests and the respective MAC addresses to be advertised to the $FWDIR/conf/local.arp file on the local SGM.

   Note: Interface VMAC value is different between Chassis when working on a Dual Chassis setup. When editing the local.arp file, MAC values should be taken from the local SGM.

   For example, in order to reply to ARP requests for IP 192.168.10.100 on interface eth2-01 with MAC address 00:1C:7F:82:01:FE, add the following entry to the local.arp file:

   ```text
   192.168.10.100 00:1C:7F:82:01:FE
   ```

2. Execute the command local_arp_update on the SGM with the updated file in order to distribute it among all the SGMS in the system. That command distributes the local.arp file to any SGM in the system, automatically changes the MAC values for SGMS on another Chassis.

3. Enable the Merge manual proxy ARP configuration option in SmartDashboard > Global Properties > NAT.

4. Install policy to apply the updated proxy ARP entries

Notes:

- When you add an SGM to a system with proxy ARP configured, the local.arp file is automatically copied to the new SGM from the SMO.
- Proxy ARP is also required when configuring Connect Control on the 61000/41000 Security System.
Verification:
In order to verify that all the entries in local.arp file are applied correctly on the system run 
`asg_local_arp_verifier`. Manual comparison can be done by running `g_fw ctl arp`.

Port speed configuration

**QSFP Data port speed configuration (40GbE / 4x10GbE)**

**Setting port speed to 40GbE**

Run the following procedure in order work in 40G mode. On dual Chassis configuration, run this procedure 
on the SSM of both Chassis.

1. Connect to the SSM shell (see SSM160 CLI section)
2. Run the following on the SSM:
   
   ```
   T-HUB4#unhide private
   Password: private (not shown)
   T-HUB4#show private shell
   /batm/var/scriptfs # /batm/binux/bin/ub_util -s ahub4_40G yes
   Writing field <ahub4_40G> with value <yes>
   Success
   /batm/var/scriptfs # exit
   T-HUB4#config terminal
   Entering configuration mode terminal
   T-HUB4(config)#system reload manufacturing-defaults
   Are you sure that you want to delete existing configuration and 
   reload manufacturing default configuration (yes/no)? yes
   ```

**Setting port to 4x10GbE (this is the default configuration)**

Run the following procedure in order work in 4x10G mode. On dual Chassis configuration, run this 
procedure on the SSM of both Chassis.

Connect to the SSM shell (see SSM160 CLI section)

Run the following on the SSM:

```
T-HUB4#unhide private
Password: private (not shown)
T-HUB4#show private shell
/batm/var/scriptfs # /batm/binux/bin/ub_util -s ahub4_40G
Erasing field <ahub4_40G>
Success
/batm/var/scriptfs # exit
T-HUB4#config terminal
Entering configuration mode terminal
T-HUB4(config)#system reload manufacturing-defaults
Are you sure that you want to delete existing configuration and 
reload manufacturing default configuration (yes/no)? yes
```

**Validation:**

The 40G ports are 1/1/1 and 1/2/1. In order to verify the speed, do as follows:
The output of the `show port 1/1/1 detailed` command is as follows:

**Ethernet Interface**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>1/1/1</td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>Admin State</td>
<td>up</td>
</tr>
<tr>
<td>Port State</td>
<td>down</td>
</tr>
<tr>
<td>Config Duplex</td>
<td>full</td>
</tr>
<tr>
<td>Operational Duplex</td>
<td>unknown</td>
</tr>
<tr>
<td>Config Speed</td>
<td>40000</td>
</tr>
<tr>
<td>Operational Speed(Mbps)</td>
<td>unknown</td>
</tr>
<tr>
<td>Flow Control</td>
<td>disabled</td>
</tr>
<tr>
<td>Dual Port</td>
<td>No</td>
</tr>
<tr>
<td>Active Link</td>
<td>No-Link</td>
</tr>
<tr>
<td>Default VLAN</td>
<td></td>
</tr>
<tr>
<td>MTU[Byte]</td>
<td>1544</td>
</tr>
<tr>
<td>MAC Learning</td>
<td></td>
</tr>
<tr>
<td>LAG ID</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Transceiver Data**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transceiver Type</td>
<td>Unknown</td>
</tr>
<tr>
<td>Cable Connector</td>
<td>MPO-Parallel-Optic</td>
</tr>
<tr>
<td>Vendor Name</td>
<td>AVAGO</td>
</tr>
<tr>
<td>Encoding</td>
<td>SONET-Scrambled</td>
</tr>
<tr>
<td>Manufacture Date</td>
<td>2010/11/18</td>
</tr>
<tr>
<td>Media</td>
<td>n/a</td>
</tr>
<tr>
<td>Serial Number</td>
<td>QA460230</td>
</tr>
<tr>
<td>TX Laser Wavelength</td>
<td>n/a</td>
</tr>
<tr>
<td>Part Number</td>
<td>AFBR-79E4Z-D</td>
</tr>
<tr>
<td>Revision Level</td>
<td>01Bh</td>
</tr>
<tr>
<td>Link Length Support</td>
<td>5000 for SMF</td>
</tr>
</tbody>
</table>

**Management Port Speed Configuration**

To set the speed of a management port on a dual Chassis configuration:

1. Connect to the SSM shell (see SSM160 CLI section)
2. Run the following on the SSM:
   ```
   #config
   #port<port>
   #speed <speed value>
   #commit
   #end
   ```
3. Run `show port <port>` to validate port speed.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| port      | In SSM160 use:  
• 1/5/3 for ethx-mgmt03  
• 1/5/4 for ethX-mgmt04  
In SSM60 use:  
• 1/5/1 for ethx-mgmt01  
• 1/5/2 for ethX-mgmt02 |
| speed value | Speed value is in Mbps could be 1000/100. |

**Example:**

```plaintext
> T-HUB4#config
Entering configuration mode terminal

--- WARNING -----------------------------------------------
Running db may be inconsistent. Enter private configuration mode and install a saved configuration.
---
T-HUB4(config)#port 1/5/4

--- WARNING -----------------------------------------------
Running db may be inconsistent. Enter private configuration mode and install a saved configuration.
---
T-HUB4(config-port-1/5/4)#speed 100

--- WARNING -----------------------------------------------
Running db may be inconsistent. Enter private configuration mode and install a saved configuration.
---
T-HUB4(config-port-1/5/4)#commit
% No modifications to commit.

--- WARNING -----------------------------------------------
Running db may be inconsistent. Enter private configuration mode and install a saved configuration.
---
T-HUB4(config-port-1/5/4)#end

T-HUB4#show port 1/5/4
```

```
Ethernet Interface
============================================================================
Interface : 1/5/4
Description :
Admin State : up Port State : up
Config Duplex : auto Operational Duplex : full
Config Speed : 100 Operational Speed(Mbps) : 100
Flow Control : disabled Dual Port : No
Active Link : RJ45
Default VLAN : 1 MTU[Bytes] : 1544
MAC Learning :
LAG ID : N/A
============================================================================
```
Multicast Configuration

Description
Multicast is a method of sending IP datagrams to a group of interested receivers in a single transmission. The Multicast group address is used to send and receive multicast messages. Sources use the group address as the IP destination address in their data packets. Receivers use this group address to inform the network that they are interested in receiving packets sent to that group.

For example, if some content is associated with group 239.1.1.1, the source will send data packets destined to 239.1.1.1. Receivers for that content will inform the network that they are interested in receiving data packets sent to the group 239.1.1.1. The receiver joins 239.1.1.1.

Dynamic Multicast Routing (PIM Dense Mode) Configuration
1. For each interface that uses PIM Dense mode, run:
   set pim interface <interface name> on
2. Set PIM mode to Dense. Run via gclish:
   set pim mode dense

To change the PIM Multicast Routing mode between dense and sparse:

⚠️ Important - You must use this procedure to change the mode. Failure to do so can cause unexpected behavior.

1. For each applicable interface, run:
   set pim interface <interface name> off
2. For each applicable interface, run:
   set pim mode dense|sparse
3. For each applicable interface, run:
   set pim interface <interface name> on

Validation
Run from gclish:
show pim interfaces

Example
> set pim interface eth1-01 on
  1_01: success
> set pim interface eth1-02 on
  1_01: success
> set pim interface eth2-01 on
  1_01: success
> set pim mode dense
  1_01: success
> show pim interfaces
  1_01: success
Status flag: V - virtual address option enabled
Mode flag: SR - state refresh enabled

<table>
<thead>
<tr>
<th>Interface</th>
<th>Status</th>
<th>State</th>
<th>Mode</th>
<th>DR Address</th>
<th>DR Pri</th>
<th>NumNbrs</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth2-01</td>
<td>Up</td>
<td>DR</td>
<td>dense</td>
<td>2.2.2.10</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>eth1-01</td>
<td>Up</td>
<td>DR</td>
<td>dense</td>
<td>12.12.12.10</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>eth1-02</td>
<td>Up</td>
<td>DR</td>
<td>dense</td>
<td>22.22.22.10</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Static Multicast Routing (SMCRoute) configuration

When working with SMCRoute, dynamic multicast routing should be disabled. The SMCRoute is not included in the OS and should be added manually. Please contact Check Point support.

SMCRoute daemon configuration:
g_all dbset process:smcroute:runlevel 4
g_all dbset process:smcroute:path /bin
g_all dbset process:smcroute:arg:1 -d
Start the SMCRoute daemon

g_all tellpm process:smcroute t
Stop the SMCRout daemon

g_all tellpm process:smcroute
g_all /bin/smcroute -k

SMCRoute Routing configuration

To add route:

g_all /bin/smcroute -a <InputIntf> <OriginIpAdr> <McGroupAdr> <OutputIntf> [<OutputIntf>]...

To remove route:

g_all /bin/smcroute -r <InputIntf> <OriginIpAdr> <McGroupAdr> - remove route

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>InputIntf</td>
<td>&lt;InputIntf&gt; can be any network interface as listed by ‘ifconfig’ but not the loopback interface.</td>
</tr>
<tr>
<td>OriginIpAdr</td>
<td>The source IP address of the multicast packets that will be routed by this entry. It is a unicast IP address not a multicast IP address</td>
</tr>
<tr>
<td>McGroupAdr</td>
<td>The IP address of the multicast group that will be forwarded.</td>
</tr>
<tr>
<td>&lt;OutputIntf&gt;</td>
<td>a list of one or more network interfaces to which the multicast packets will be forwarded</td>
</tr>
<tr>
<td>[&lt;OutputIntf&gt;]</td>
<td></td>
</tr>
</tbody>
</table>

Example:
g_all /bin/smcroute -a eth2-01 2.2.2.1 225.0.90.90 eth1-01 eth1-02

**Multicast restrictions**

Multicast access restrictions can be defined on each interface. These restrictions specify multicast groups (that is, addresses or address ranges) to allow or block.

Configuration

1. Open SmartDashboard and edit the Multicast Restrictions tab:
2. Go to Gateway Properties > Topology > Add or Edit interface > Multicast Restrictions tab

Parameter | Description
--- | ---
Drop multicast packets whose destination is in the list | Specifies that outgoing packets from this interface to the listed multicast destinations will be dropped.
Drop all multicast packets except those whose destination is in the list | Specifies that outgoing packets from this interface to all multicast destinations except those listed will be dropped.
Add | Add a Multicast address or address range to the list.
Remove | Remove a selected Multicast address or address range from the list.
Tracking | Allows you to choose whether and how to track when multicast packets are dropped.

Limitations:
Multicast restriction is not supported on bridge interfaces.
**Multicast acceleration**

Multicast acceleration allows SecureXL to accelerate multicast flow, also in Fan-out scenarios.

**Configuration**

Multicast acceleration is enabled by default. In order to enable/disable it run from gclish the flowing set of commands:

```plaintext
sim feature mcast_route_v2 {on | off}
fwaccel off
fwaccel on
```

**Limitations**

Multicast acceleration supports IPv4 only.

**Validation and Debugging**

```plaintext
> fwaccel stat
--- 4 blades: 1_01 1_02 2_01 2_02 ---
Accelerator Status : on
Accept Templates   : enabled
Drop Templates     : disabled
NAT Templates      : enabled
Accelerator Features : Accounting, NAT, Cryptography, Routing, 
HasClock, Templates, Synchronous, IdleDetection, 
Sequencing, TcpStateDetect, AutoExpire, 
DelayedNotif, TcpStateDetectV2, CPLS,McastRouting, 
WireMode, DropTemplates, NatTemplates, 
Streaming, MultiFW, AntiSpoofing, DoS Defender, 
ViolationStats, Nac, AsychronicNotif, McastRoutingV2, 
ConnectionsLimit
Cryptography Features : Tunnel, UDPEncapsulation, MD5, SHA1, NULL, 
3DES, DES, CAST, CAST-40, AES-128, AES-256, 
ESP, LinkSelection, DynamicVPN, NatTraversal, 
EncRouting, AES-XCBC, SHA256
```

Display the accelerator's connections table by running: `fwaccel conns`
Display multicast statistics by running: `fwaccel stats -m`
Enable SIM debug using the command: `sim dbg -m drv + routing`

**Example**:

The following example disables the feature.
> `sim feature mcast_route_v2 off`
> `-- 4 blades: 1_01 1_02 1_03 1_04` <~>
> Feature will be disabled the next time acceleration is started/restarted

> `fwaccel off`
> `-- 4 blades: 1_01 1_02 1_03 1_04` <~>
> SecureXL device disabled.

> `fwaccel on`
> `-- 4 blades: 1_01 1_02 1_03 1_04` <~>
> SecureXL device is enabled.

> `fwaccel stat`
> `-- 4 blades: 1_01 1_02 1_03 1_04` <~>
> Accelerator Status: on
> Accept Templates: enabled
> Drop Templates: disabled
> NAT Templates: enabled
> Cryptography Features: Tunnel, UDP encapsulation, MD5, SHA1, NULL, 3DES, DES, CAST, CAST-40, AES-128, AES-256, ESP, LinkSelection, DynamicVPN, NatTraversal, EncRouting, AES-XCBC, SHA256

**Configuring DHCP Relay (set bootp)**

Use this command to configure DHCP relay for a specified interface.

BOOTP/DHCP Relay extends BOOTP and DHCP operations across multiple hops in a routed network. With standard BOOTP, all LAN interfaces are loaded from one configuration server on the LAN. BOOTP Relay sends configuration requests to and from configuration servers located outside the LAN.

BOOTP/DHCP Relay has these advantages over standard BOOTP/DHCP:

- You can provide redundancy by configuring an interface on the Check Point system to relay client configuration requests to multiple servers. Configuration requests are sent to all configured relay servers simultaneously.
- Load balancing - You can configure interfaces to relay client configuration requests to different relay servers.
- It lets you centrally manage client configuration over multiple LANs. This is particularly useful in large enterprise environments.

**Syntax**

```
set bootp interface <if_name> on|off
set bootp interface <if_name> primary <ip> wait-time <0-65535>} on
set bootp interface <if_name relay-to <ip} on | off
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>interface &lt;if_name&gt;</td>
<td>The interface name as defined by the system. Press Tab after you enter this parameter to see a list of valid interface names.</td>
</tr>
<tr>
<td>primary ip_address</td>
<td>The IP address of the Security Gateway interface that always gets requests from the DHCP client. If you do not define a Primary IP address, the system automatically uses the IP address of the interface that the DHCP request comes from.</td>
</tr>
</tbody>
</table>
Parameter | Description
---|---
wait-time <0-65535> | The minimum wait time, in seconds, before a BOOTP request can be sent (default = 60 seconds). This includes the elapsed time after the client starts to boot. This delay lets a local configuration server reply, before it sends the relay to a remote server.
relay-to <ip> on|off | The IP address of the relay server to which BOOTP requests are sent. You can specify more than one server.
on | off | Enables or disables BOOTP on the specified interface.

Examples
This example enables DHCP Relay on eth0-4 with default values and no Primary IP. The IP address is automatically assigned by DHCP server.

> set bootp interface eth0-4 on

This example enables DHCP Relay on eth0-4 and defines the Primary IP address as 30.30.30.1. The wait time is the default value (60 seconds).

> set bootp interface eth0-4 primary 30.30.30.1 wait-time default on

This example enables DHCP Relay on eth1-04 and sends BOOTP requests to the relay server at 20.20.20.200.

> set bootp interface eth1-04 relay-to 20.20.20.200 on

Verification
Use this command to monitor and troubleshoot the BOOTP implementation:

> show bootp
  interface - BOOTP/DHCP Relay Interface
  interfaces - All BOOTP/DHCP Relay Interfaces
  stats - BOOTP/DHCP Relay Statistics

Configuring Netflow Export - CLI (netflow)

To add a collector:
add netflow collector ip VALUE port VALUE [srcaddr VALUE export-format VALUE]

To delete a collector:
delete netflow collector [for-ip VALUE [for-port VALUE]]

To change settings of a collector:
set netflow collector [for-ip VALUE [for-port VALUE]]
  export-format VALUE
  srcaddr VALUE
set netflow collector [for-ip VALUE]
  port VALUE
set netflow collector
  ip VALUE

Parameter | Description
---|---
ip VALUE | The IPv4 address to which NetFlow packets are sent. This is mandatory.
port VALUE | The UDP port number on which the collector is listening. This is mandatory. There is no default or standard port number for NetFlow.
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>srcaddr VALUE</td>
<td>Optional: The IPv4 address of the NetFlow packets source. This must be an IP address of the local host. The default (which is recommended) is an IP address from the network interface on which the NetFlow traffic is going out.</td>
</tr>
<tr>
<td>export-format VALUE</td>
<td>The NetFlow protocol version to send: 5 or 9. Each has a different packet format. The default is 9.</td>
</tr>
<tr>
<td>for-ip VALUE</td>
<td>The for-ip and for-port parameters specify the collector that the command operates on. If you only have one collector configured, you do not need these parameters. If you have two or three collectors with different IP addresses, use for-ip. If you have two or three collectors with the same IP address and different UDP ports, you must use for-ip and for-port to identify the one you want to work on.</td>
</tr>
<tr>
<td>for-port VALUE</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 3
System Optimization

In This Section:

- Firewall connections table size for Security Gateway ........................................ 165
- Firewall connections table size for VSX Gateway ........................................... 166
- Reserved connections ......................................................................................... 166
- Policy Acceleration – SecureXL Keep Connections ........................................ 170
- Extending SecureXL Templates ....................................................................... 170
- VPN Performance Enhancements ..................................................................... 172
- SCTP Acceleration ............................................................................................ 174
- Configuring DNS Session Rate ......................................................................... 176
- Fast packet drop .................................................................................................. 177
- Configuring Hyper-Threading ........................................................................... 179
- Configuring CoreXL on a VSX Gateway (g_cpconfig) ....................................... 179
- System Under Load ............................................................................................ 183
- Working with Jumbo Frames ............................................................................. 186
- TCP MSS Adjustment ......................................................................................... 190
- Working with Session Control (asg_session_control) ....................................... 190
- Hide NAT Behind Range – Sticky per SGM (asg_hide_behind_range) ............. 190
- Acceleration Not Disabled Because of Traceroute Rule (asg_tmpl_special_svc) 193
- Improving the Performance of Inbound HTTPS .................................................. 193

Firewall connections table size for Security Gateway

Description

Firewall connections table default size per SGM is set automatically with the following values, regardless of SmartDashboard configuration:

- SGMs with 12G RAM: 3,500,000
- SGMs with 24G RAM: 7,000,000

This behavior aims to minimize the additional settings, required by customer before deployment.

Note - setting the maximum limit for concurrent connections to Automatically (in the SmartDashboard Gateway object > Capacity Optimization) is not supported.

Configuration

To set a different value, instead of 3.5M/7M, run:

```
# fw ctl set int fwconn_tab_limit_user <new value, e.g. 4000000>
# update_conf_file fwkern.conf fwconn_tab_limit_user=<new value, e.g. 4000000>
```

Deactivation

To restore legacy behavior and configure firewall connections table size, from SmartDashboard Gateway Properties > Capacity Optimization > Maximum concurrent connections, run:
# update_conf_file fwkern.conf fwconn_tab_limit_from_policy=1
# reboot -b all

Verification
To verify firewall connections table size run:

`# fw tab -t connections -m 1`

And check limit attribute in each SGM.

Example
`fw tab -t connections -m 1`

```
1_01:
localhost:
-------- connections --------
dynamic, id 8158, attributes: keep, sync, aggressive aging, kbufs 18 19 20 21
22 23 24 25 26 27 28 29 30 31 32 33 34 35, expires 25, refresh, limit 3500000,
hashsize 4194304

1_02:
localhost:
-------- connections --------
dynamic, id 8158, attributes: keep, sync, aggressive aging, kbufs 18 19 20 21
22 23 24 25 26 27 28 29 30 31 32 33 34 35, expires 25, refresh, limit 3500000,
hashsize 4194304
```

Firewall connections table size for VSX Gateway
You configure the Firewall connections table for VSX Gateway, Virtual Systems and other VSX Virtual Devices in SmartDashboard.

To configure the Firewall connections table:
1. Open the Virtual Device object in SmartDashboard.
2. Select the applicable Virtual Device.
3. Select Optimizations in the navigation tree.
4. On the Optimizations page, select Manually in Calculate the maximum limit for concurrent connections.
5. Enter or select a value.

Reserved connections
Description
Normally, when the connection table limit is reached, no more connections are allowed, even ones critical for operating and managing the gateway. The reserved connections feature allows the gateway to process these critical connections, even after the connections table limit is reached. There is a user defined amount of space that is reserved in the connections table for these critical connections. If the Rule Base allows these connections, they are allowed even if no other connections can be accepted.

For example, when the connections table limit is reached, the administrator may not be able to install a new policy that increases the connections limit or open other essential connections, such as SSH to the gateway.
Notes

Enforcing the reserved connections limit

The connections table limit is defined in the Capacity Optimization tab, but a certain amount of connections table space is always available for reserved traffic. By default, the number of reserved connections is limited to 2000 and the actual limit of the connections table is increased by this amount.

Before a new connection is recorded, the system verifies that there is enough space in the connections table. If connections table limit is reached, the connection is recorded if it satisfies these conditions:

The limit is below the limit sum of 'connections table limit' and 'reserved connections limit'

- Connection matches one of the rules in the reserved connections table
- Otherwise the connection recording fails.

In VSX Reserved Connections is supported for VS0 only.

Syntax

asg_reserved_conns

Example 1

To display the initial list of connections which are allowed to be recorded in the connections table, even if it has reached its defined capacity, run the asg_reserved_conns command and choose 1) Print reserved connections table.

<table>
<thead>
<tr>
<th>Idx</th>
<th>Source</th>
<th>Mask</th>
<th>Destination</th>
<th>Mask</th>
<th>DPort</th>
<th>Ipp</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>1129</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>2</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>1130</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>3</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>4</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>4444</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>5</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>6</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>8888</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>7</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>2010</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>8</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>1131</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>9</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>256</td>
<td>6</td>
<td>Sync</td>
</tr>
<tr>
<td>10</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>1</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>11</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>17</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>12</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>23</td>
<td>17</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>13</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>161</td>
<td>17</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>14</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>623</td>
<td>17</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>15</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>eth1-CIN</td>
</tr>
<tr>
<td>16</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>6</td>
<td>eth2-CIN</td>
</tr>
<tr>
<td>17</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>23</td>
<td>6</td>
<td>eth2-CIN</td>
</tr>
<tr>
<td>18</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>161</td>
<td>17</td>
<td>eth2-CIN</td>
</tr>
<tr>
<td>19</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>623</td>
<td>17</td>
<td>eth2-CIN</td>
</tr>
<tr>
<td>20</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>22</td>
<td>6</td>
<td>Any</td>
</tr>
<tr>
<td>21</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>256</td>
<td>6</td>
<td>Any</td>
</tr>
<tr>
<td>22</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>18191</td>
<td>6</td>
<td>Any</td>
</tr>
<tr>
<td>23</td>
<td>0.0.0.0</td>
<td>0</td>
<td>0.0.0.0</td>
<td>0</td>
<td>18192</td>
<td>6</td>
<td>Any</td>
</tr>
</tbody>
</table>

Press enter to continue
Idx - The rule number.

Source and Mask - The IP address 0.0.0.0 stands for Any.

Destination and Mask - The destination IP address and mask.

Dport - The service number. In case of non-TCP/UDP protocol (6/17) it should be ignored.

Ipp - The IP protocol number – 6 for TCP, 17 for UDP, 1 for ICMP and so on.

Interface - The interface to which interface the rule applies.

Example 1

Adding new reserved connection rule:

Run the command `asg_reserved_conns` and choose 2) Add new reserved connection rule

Output

Enter source IP [0.0.0.0]:
>10.10.10.0
Enter source IP mask length [0]:
>24
Enter destination IP [0.0.0.0]:
>20.20.20.0
Enter destination IP mask length [0]:
>24
Enter destination port [0]:
>0
Enter IP protocol number (for example: tcp – 6, udp – 17):
>6
Enter interface number [0 = Any]:
0: Any
1: eth1-Mgmt4
2: eth2-Mgmt4
3: BPEth0
4: BPEth1
5: eth1-01
6: eth2-01
7: eth1-Mgmt1
8: eth1-CIN
9: eth2-Mgmt1
10: eth2-CIN
11: Sync
>0
OK to insert new reserved conn rule: <10.10.10.0/24, 20.20.20.0/24, 0, 6, 0, Any, 11, Sync>
y
entry inserted, rule will apply when new connection will be opened
Press enter to continue

Configuration

The feature works after installation without additional configuration. Use the `asg_reserved_conns` CLI to manage the reserved connections rules.

The rules are recorded in the `reserved_conns_table` kernel table.

Kernel global variables:

`fwconn_reserved_conn_active` (type int) enables or disables the feature. Default is 1 (enabled).

`fwconn_reserved_limit` (type int) contains the number of entries in the `reserved_conns_table` kernel table. Default is 2000
### Verification
To make sure the feature is configured properly do the following:

1. Check that the value of the kernel global parameter `fwconn_reserved_conn_active` is set to 1.
2. Run the command `asg_reserved_conns` and choose 1) Print reserved connections table.
3. Run `fw tab -t reserved_conns_table` and make sure that the table contains the entries for the rules above.
4. Check the contents of the file `$FWDIR/bin/reserved_conns_tab` and make sure it contains the rules above.

### Debugging
To enable reserved connections debugging, set the following kernel global parameter and use the CONN kernel debug flag to see reserved connections related debugs.

- `fwreserved_conns_debug`: (type int) used to enable reserved connections debug prints. Default 0 (disabled)

### Troubleshooting
1. Run "`fw tab -t reserved_conns_table" and make sure that the table contains the entries for the rules above.
2. Check the contents of the file `$FWDIR/bin/reserved_conns_tab` and make sure it contains the rules above. This file is not intended to be edited directly.
3. Run the `asg_reserved_conns -f` command to delete all current rules from kernel and reload the reserved rules table from the file `$FWDIR/bin/reserved_conns_tab` to kernel. It is useful if there were changes in network interface names or when the `$FWDIR/bin/reserved_conns_tab` file was edited directly.
Policy Acceleration – SecureXL Keep Connections

Description
Allow flow acceleration while pushing policy to the system.

Configuration
Select "Keep all connections" in the SmartDashboard gateway's properties Other->connection persistence

Note - Feature is enabled only if:
- SecureXL is enabled
- Firewall Software Blade only is enabled

In SmartDashboard:

Legacy mode
To allow Keep all connections while disabling SecureXL keep connections set cphwd_policy_accel=0 in $FWDIR/boot/modules/fwkern.conf

Verification
After policy installation, templates of the old policy should be deleted. This can be tracked in the following way:
1. Run g_fwaccel stats
2. Save the old value of the "Policy deleted tmpl" statistics
3. Install policy
4. Run g_fwaccel stats
5. Make sure that templates were deleted

Extending SecureXL Templates

Description
To enhance connection rate and throughput in a SecureXL enabled environment, the firewall groups together packets of a connection that share the same service (same source port). The first packets of the first connection are handled by the firewall. The firewall then offloads the connection to SecureXL (acceleration hardware or software) for processing.

SecureXL creates a connection template that matches the accept rule in the firewall Rule Base, but with a wildcard replacing the source port. New connections that match the template are processed by SecureXL.
On a busy network, repeated connections to the same DNS server clearly benefit from SecureXL acceleration, where the DNS source port (53) is replaced by a wildcard. However, multiple IP addresses can resolve to the same DNS name. In such an environment, replacing the source IP address with a second wildcard decreases the number of connections processed by the firewall.

To replace source IP addresses with a second wild card, you must extend the existing SecureXL templates.

**Note** - By default, SecureXL template extension is disabled.

### To enable SecureXL template extension for accelerated DNS connections:

**On the SMO:**

1. Exit gclish
   
   (To exit gclish, enter: shell.)

2. Open: `/etc/ppk.boot/boot/modules/simkern.conf` for editing.
   
   If the file does not exist, create it.

3. Add `sim_use_srcip wildcard_for_template=1` to the file.

4. Copy the file to all SGMs by running:
   
   ```
   g_cp2blades -a /etc/ppk.boot/boot/modules/simkern.conf
   ```

5. Open: `/etc/fw.boot/modules/fwkern.conf` for editing

6. Add `cphwd_src_ip_template_enabled=1` to the file.

7. Copy the file to all SGMs by running:
   
   ```
   g_cp2blades -a /etc/fw.boot/modules/fwkern.conf
   ```

8. Reboot all SGMs.

In the SecureXL acceleration template, the source IP address and source port are replaced with wildcards.

**Note** - Traffic is only accelerated if DNS is the destination port (53).

### To add other services to the template (for example HTTP and Telnet):

**On the SMO:**

1. Exit gclish
   
   (To exit gclish, enter: shell.)

2. Open: `/etc/fw.boot/modules/fwkern.conf` for editing

3. Add `cphwd_use_srcip wildcard_for_template=80,23` to the file.
   
   This adds ports 80 and 23 to the list of permitted destination ports.
   
   - Separate each port number with a comma
   - Do not add more than 4 port numbers
   
   For UDP services, add: `cphwd_src_ip_tmpl_udp_ports= <UDP port numbers>`.

4. Copy the file to all SGMs by running:
   
   ```
   g_cp2blades -a /etc/fw.boot/modules/fwkern.conf
   ```

5. Open /etc/ppk.boot/boot/modules/simkern.conf for editing

6. Add `sim_src_ip_tmpl_tcp_ports=80,23` to the file.
   
   For UPD services, add `sim_src_ip_tmpl_udp_ports=<UDP port numbers>`

7. Copy the file to all SGMs by running:
   
   ```
   g_cp2blades -a /etc/ppk.boot/boot/modules/simkern.conf
   ```

8. Reboot all SGMs.

### Verification

To make sure extended SecureXL templates are being used:

1. In gclish, run: `fwaccel templates`

2. Examine the output.
An asterisk (*) in the Source column and an increasing Conns counter means the extended template is being utilized.

VPN Performance Enhancements

These VPN performance enhancements are included in this release:

- **SPI Based Traffic Distribution for SSM160** - Uses all SGMs to handle VPN traffic based on the SPI instead of the IP address
- **SPI affinity** - Better traffic assignment to SGM CPU cores
- **VPN Templates** - Accelerates the session rate by adding VPN Templates to the SecureXL technology

**SPI Distribution on SSM160 (asg dxl spi)**

By default, the SSM160 distributes traffic to SGMs based on the IP address in the packet header. This methodology can be inefficient when working with a small number of remote peers in a Site-To-Site VPN topology. This is because the SSM160 only sees the VPN tunnel IP address and causes distribution only to some SGMs.

To resolve this issue, you can enable SPI distribution for VPN traffic.

⚠️ **Important** - You must not enable SPI distribution and Sticky SA ("VPN Sticky SA (for LTE)" on page 196) at the same time.

**Syntax**

```
> set distribution spi mode on|off
```

**Note:** SPI distribution mode is disabled by default.

**SPI Affinity (asg_spi_affinity)**

**Description**

The `asg_spi_affinity` command helps you improve VPN performance with more efficient traffic assignment to SGMs and SGM cores. Typically, most VPN traffic goes to the same tunnel IP addresses. Because traffic is normally assigned to SGMs based on the destination IP address, VPN traffic is often assigned to the same SGMs. The solution is to assign VPN traffic to SGMs based on the SPI field in the packet header instead of the IP address.

A related issue occurs with Multi-core VLAN traffic, where traffic is assigned to CPU cores based on IP addresses. As with VPN traffic, `asg_spi_affinity` can also assign VLAN traffic to CPU cores based on the SPI field.

You must run this command in the Expert mode.
Syntax:

```
# asg_spi_affinity mode <ssm_id|all> <on|off>
# asg_spi_affinity vlan <ssm_id|all> <on|off>
# asg_spi_affinity verify
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mode</td>
<td>Configure VPN affinity for specified SSM.</td>
</tr>
<tr>
<td>vlan</td>
<td>Configure VLAN affinity for the specified SSM interfaces.</td>
</tr>
<tr>
<td>verify</td>
<td>Show SPI affinity status.</td>
</tr>
<tr>
<td>&lt;ssm_id&gt;</td>
<td>SSM identifier (1-4 or all)</td>
</tr>
<tr>
<td>on/off</td>
<td>Enable\Disable SPI affinity. You must enable vlan and mode (VPN) affinity</td>
</tr>
<tr>
<td></td>
<td>separately.</td>
</tr>
</tbody>
</table>

Notes:

- When some SSM interfaces not configured as VLANs, we recommend that you enable VLAN affinity only if most traffic passes through VLAN interfaces.
- SPI affinity can affect the distribution of clear packets. We recommend that you use SPI affinity only if most of the inbound traffic is VPN traffic.

Examples

```
# asg_spi_affinity mode 1 on - Enable VPN affinity for SSM 1
# asg_spi_affinity mode 2 off - Disable VPN affinity for SSM 2
# asg_spi_affinity vlan all on - Enable VLAN affinity for all SSM interfaces
# asg_spi_affinity vlan all off - Disable VLAN affinity for all SSM interfaces
```

**VPN Templates**

You can now use VPN templates, which accelerate the session rate, particularly for short connections (HTTP, DNS). These templates, which are part of the SecureXL template set, let you create new connections in the acceleration layer. They only notify the Firewall layer if the connection is too long or if an F2F attack is detected. VPN templates are enabled by default.

**To disable VPN templates:**

1. Run:
   ```
   > update_conf_file fwkern.conf cphwd_offload_vpn_templates=0
   ```
2. Reboot all SGMs.

**To re-enable VPN templates,** change `cphwd_offload_vpn_templates` to `1`. 
**SCTP Acceleration**

**Smart Dashboard Configuration:**
Create SCTP as "other" service using IP protocol 132
Enable "Accept Replies" property in the advanced tab of the created SCTP service

![Advanced Other Service Properties](image)

To Configure the 61000/41000 Security System

1. Connect to the SMO expert mode. Run `shell`
2. Open: `$FWDIR/boot/modules/fwkern.conf` for editing. If the file does not exist, create it.
3. Add `sxl_accel_proto_list=132` to the file.
4. Open: `$PPKDIR/boot/modules/simkern.conf` for editing. If the file does not exist, create it.
5. Add `sim_accel_non_tcpudp_proto=1` to the file.
6. Copy the file to all SGMs by running:
   - `g_cp2blades $FWDIR/boot/modules/fwkern.conf`
   - `g_cp2blades $PPKDIR/boot/modules/simkern.conf`
7. Reboot all SGMs. Run `reboot -b all`
Configuring DNS Session Rate

Description
To improve the DNS session rate, the 61000/41000 Security System includes these enhancements:

- **Delayed Connection** - When a DNS connection matches a SecureXL template, the 61000/41000 Security System firewall is not immediately notified. The notification is delayed using the global parameter: `cphwd_udp_selective_delay_ha`. After a delay is set, the connection is handled completely by the acceleration device.

  Note - If the connection is not completely handled (and closed) by the acceleration device during the set delay period, then the firewall is notified in the usual manner.

- **Delete on Response** - After the DNS response is received, the connection is immediately deleted from the gateway instead of being kept for an additional 60 seconds (the UDP connection default timeout).

Syntax
From gcish, run these commands, in this order:

```bash
>fw ctl set int cphwd_udp_selective_delay_ha <delay in seconds>
>fwaccel off
>fwaccel on
```
Verification  To make sure that DNS connections are delayed by the set value:
1. Open several DNS connections from the same client to the same server
2. Run: `fwaccel templates`
   ```plaintext
   > fwaccel templates
   ```
   The delay you see for the DNS template (under DLY field) should match the value specified for `cphwd_udp_selective_delay_ha`.

Note - The default value for this parameter is 30 seconds. The maximum value is 60.

To make the enhancements Permanent:
Update `fwkern.conf` by running:
   ```plaintext
   > update_conf_file fwkern.conf cphwd_udp_selective_delay_ha=<delay>
   ```

To turn off the enhancements:
To turn off Delayed Connection and Delete on Response:
- Set `cphwd_udp_selective_delay_ha` to zero, or
- Remove all services from `cphwd_delayed_udp_ports`.

Note - this disables both enhancements.

Extending Session Rate Enhancements to other UDP Services
By modifying the value of `cphwd_delayed_udp_ports` in `fwkern.conf`, you can extend the benefits of these two DNS session rate enhancements to other services. For example, to add UDP service 100 to the list, from cglish run:
   ```plaintext
   > update_conf_file fwkern.conf cphwd_delayed_udp_ports=53,100,0,0,0,0,0,0
   ```

Note -
- The number of services is limited to 8.
- The command must contain 8 values. If you configure less than 8 services, enter 0 for the others.
- Directly updating `fwkern.conf` is the only way to extend DNS session rate enhancements to other UDP services (`fw ctl set int` is not supported).
- The configuration takes effect only after reboot.

Fast packet drop

Description  Fast packet drop can be used in situations, such as when under DoS attack, to drop unwanted packets as early as possible in the packet processing path. This makes the gateway's resources available to process legitimate traffic. The Rule Base is in a configuration file that defines which packets should be dropped.

Syntax    `sim dropcfg < -l|-f <file>|-r|-y|-h>`
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-l</td>
<td>Show current configuration</td>
</tr>
<tr>
<td>-f &lt;file&gt;</td>
<td>Set configuration file name</td>
</tr>
<tr>
<td>-r</td>
<td>Reset drop rules</td>
</tr>
<tr>
<td>-y</td>
<td>Do not require confirmation</td>
</tr>
<tr>
<td>-h</td>
<td>Show usage information</td>
</tr>
</tbody>
</table>

**Configuration**

1. Create the Rule Base configuration file (see details below)
2. Copy the configuration file to all SGMs. Run from gclish:

   `sim dropcfg -f <configuration file>`

The Rule Base configuration is specified using the `-f` CLI option. It contains drop rules, and each line should contain a single rule.

Each rule line must contain one or more of the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>src &lt;src ip&gt;/&lt;subnet&gt;</td>
<td>Source IP address and subnet. Subnet is optional</td>
</tr>
<tr>
<td>dst &lt;dst ip&gt;/&lt;subnet&gt;</td>
<td>Destination IP address and subnet. Subnet is optional</td>
</tr>
<tr>
<td>dport &lt;dst port&gt;</td>
<td>Destination port.</td>
</tr>
<tr>
<td>proto &lt;ip proto&gt;</td>
<td>IP Protocol (e.g. TCP=6,UDP=17,ICMP=1)</td>
</tr>
</tbody>
</table>

**Notes**

If subnet is not specified, a single IP address is used.

Use '*' to specify 'Any'. It is the same as not specifying the parameter.

Use '#' at the beginning of the line to add comments.

Empty lines are ignored.

**Examples**

Example configuration file:

```
src 1.1.1.1
dport 80 proto 6
src 1.1.1.0/24 dst 2.2.0.0/16 dport 53 proto 17
```

**Verification**

To make sure fast packet drop rules are being enforced, run the command:

`sim dropcfg -l`

The output shows list of active drop rules:

```
Drop rules (Match after conn lookup):
Source             Destination        DPort PR
----------------- ----------------- ---- ----
1.1.1.1/32                  *     *   *
              *                  *    80   6
1.1.1.0/24         2.2.0.0/16    53  17
```

**Disabling Fast Packet Drop**

If there are drop rules defined, run the following command to clear the fast packet drop Rule Base:

`sim dropcfg -r`
Configuring Hyper-Threading

Description
Hyper-threading lets a compatible operating system run more than one process run simultaneously on a processor core. A Hyper-threading processor adds one or more "logical" processors, which the operating system "sees" as independent processors.

To enable Hyper-threading, run `g_cpconfig` in the Expert mode.

Syntax

```
# g_cpconfig ht stat
# g_cpconfig ht enable
# g_cpconfig ht disable
# g_cpconfig ht show stat
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stat</td>
<td>Shows whether hyper-threading is enabled for the 61000/41000 Security System</td>
</tr>
<tr>
<td>enable</td>
<td>Enable Hyper-threading</td>
</tr>
<tr>
<td>disable</td>
<td>Disable Hyper-threading</td>
</tr>
<tr>
<td>show stat</td>
<td>Shows the hyper-threading status for all SGMs</td>
</tr>
</tbody>
</table>

Notes
You must reboot all SGMs after you enable or disable hyper-threading.

Configuring CoreXL on a VSX Gateway (`g_cpconfig`)

Use the `g_cpconfig` command to configure CoreXL on the VSX Gateway (VS0). The number of instances for the VSX Gateway is limited to the physical number of cores on the 61000/41000 Security System.

Note - If you run this command in a Virtual System, the output applies to VS0.

Syntax

```
g_cpconfig corexl stat
# g_cpconfig corexl enable [n] [-6 [k]]
g_cpconfig corexl disable
# g_cpconfig corexl instances [n] [-6 [k]]
g_cpconfig corexl show instances
# g_cpconfig corexl show stat
```

Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>stat</td>
<td>Show current status and number of instances on all SGMs.</td>
</tr>
<tr>
<td>enable [n] [-6 [k]]</td>
<td>Enable CoreXL with <code>&lt;n&gt;</code> IPv4 Firewall instances and <code>&lt;k&gt;</code> IPv6 Firewall instances. Minimum = 2. Maximum = 32. Default = 16</td>
</tr>
<tr>
<td>disable</td>
<td>Disable CoreXL.</td>
</tr>
<tr>
<td>instances [n] [-6 [k]]</td>
<td>Change the number of IPv4 Firewall instances to &lt;n&gt; and IPv6 Firewall instances to &lt;k&gt;. Minimum = 2. Maximum = 32. Default = 16</td>
</tr>
<tr>
<td>show instances</td>
<td>Show the number of instances on each blade</td>
</tr>
<tr>
<td>show stat</td>
<td>Show the status on each blade</td>
</tr>
</tbody>
</table>
Enabling Cores

> g_cpconfig corexl enable 8 -6 8

```
--*-- 5 blades: 1_01 1_02 2_01 2_02 2_04 --*-  
rx_num for ixgbe interfaces was set to: 16
```

CoreXL was successfully enabled with 8 IPv4 and 8 IPv6 firewall instances.

Important: This change will take effect after rebooting all blades.

Showing CoreXL status per SGM

> g_cpconfig corexl show stat

```
blade 1_01 corexl is enabled
blade 1_02 corexl is enabled
blade 1_03 corexl is enabled
```

CoreXL configuration on a VSX system

When you change the number of CoreXL instances in a Security Gateway environment, all CPUs not assigned to CoreXL are assigned to Performance Pack. When you change the number of CoreXL instances in a VSX Gateway environment, you only change the number of user-mode threads. This has no effect on Performance Pack affinity and the number of CPUs assigned to Performance Pack does not change.

This example shows a system with 12 CPUs and 3 Virtual Systems where:

- Each Virtual Systems has 1 CoreXL instance
- CPUs 0-7 are assigned to Firewall packet inspection
- CPUs 8-11 are assigned to Performance Pack

> g_cpconfig corexl instances 3

```
• The number of CoreXL instances (user-mode threads) changes from 1 to 3. Each Virtual System still has one CoreXL instance.
• CPUs 0-7 are still assigned to Firewall packet inspection
• CPUs 8-11 are still assigned to Performance Pack
```

**VSX Affinity Commands (fw ctl affinity -s -d)**

This section shows you how to use the `fw ctl affinity` command to set affinities in a VSX environment. When you run this command, the system automatically creates or updates the affinity configuration files. All affinity configurations are kept after reboot.

You can define specified processes as affinity exceptions. Affinity commands do not apply these processes. To define an exception, add the process name to the `$FWDIR/conf/vsaffinity_exception.conf` file. You cannot add kernel threads as affinity exceptions.

⚠️ **Important** - Do not add Check Point processes to the exception list. This can cause system instability.

**Affinity Priorities**

When a CPU core has more than one affinity, the affinity is applied based on these priorities:

1. Firewall instance
2. Process
3. Virtual System

**Setting Affinities**

Use the `fw ctl affinity -s -d` command to set these CPU affinities:

- Firewall instance
- Processes
- Virtual System
You can set Firewall instance affinity to one or more CPUs on each Virtual System individually.

Syntax

```
fw ctl affinity -s -d
fw ctl affinity -s -d [-vsid <vs_ids>] -cpu <cpu_id>
fw ctl affinity -s -d -pname <process> [-vsid <ranges>] -cpu <cpu_id>
fw ctl affinity -s -d -inst <instance_id> -cpu <cpu_id>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-s -d</td>
<td>Set affinity for a VSX environment.</td>
</tr>
<tr>
<td>-vsid &lt;vs_ids&gt;</td>
<td>The &lt;vs_ids&gt; can be:</td>
</tr>
<tr>
<td></td>
<td>• No &lt;vs_ids&gt; (default) - Shows the current Virtual System context.</td>
</tr>
<tr>
<td></td>
<td>• One Virtual System.</td>
</tr>
<tr>
<td></td>
<td>• A comma-separated list of Virtual Systems (1,2,4,5).</td>
</tr>
<tr>
<td></td>
<td>• A range of Virtual Systems (VS 3-5).</td>
</tr>
<tr>
<td></td>
<td>• all - Shows all Virtual Systems.</td>
</tr>
<tr>
<td></td>
<td>Note: This parameter is only relevant in a VSX environment.</td>
</tr>
<tr>
<td>-cpu &lt;cpu_id&gt;</td>
<td>One or more CPU cores. You can define a range from which the system</td>
</tr>
<tr>
<td></td>
<td>selects the instances. The format for a range is:</td>
</tr>
<tr>
<td></td>
<td>&lt;from_cpu_id&gt;-&lt;to_cpu_id&gt;.</td>
</tr>
<tr>
<td>-pname &lt;process&gt;</td>
<td>Configure affinity for the specified process.</td>
</tr>
<tr>
<td>-inst &lt;instance_id&gt;</td>
<td>One or more Firewall instances. You can define a range from which the</td>
</tr>
<tr>
<td></td>
<td>system selects the instances. The format for a range is</td>
</tr>
<tr>
<td></td>
<td>&lt;from_instance_id&gt;-&lt;to_instance_id&gt;.</td>
</tr>
</tbody>
</table>

Setting affinities for all SGMs from the SMO:

From gclish, run `fw ctl affinity -s -d <options>`

From the Expert mode run `g_fw ctl affinity -s -d <options>`

Setting affinities for a specified SGM:

1. Run:
   ```
   blade <sgm_id>
   ```
2. Run:
   ```
   fw ctl affinity -s -d <options>
   ```

Setting Firewall instance affinity with ranges

This example creates two Firewall instance affinities for the Virtual System on context 1. One affinity is assigned to instance 0 and the other is automatically assigned from the range of instances 2-4. These instances are automatically assigned to CPU cores in the range of 0-2.

```
vsenv 1
> fw ctl affinity -s -d -inst 0 2-4 -cpu 0-2

VDevice 0: CPU 0 1 2 - set successfully
```

Note: If there were previously configured processes/FWK instances, this operation has overridden them and deleted their configuration files.

```
Athens-ch01-02:0>
```

Setting VSX processes affinity (-pname)

Set the affinity of processes to one or more CPUs. You can use the -vsid parameter to set the affinity for a process to Virtual Systems in any context. If you do not use the -vsid parameter, the affinity of the current context is set.
Virtual System affinity (-vsid)

Use the `-vsid` parameter to define an affinity for specified Virtual Systems. This example sets the affinity for Virtual System contexts 0 and 1 to CPU cores 0 and 2. If you do not use the `-vsid` parameter, this command sets the affinity for the current VSX context.

```
> fw ctl affinity -s -d -vsid 0-1 -cpu 0 2
VDevice 0-1 : CPU 0 2 – set successfully
```

Setting Affinity for all Virtual Systems (fw ctl affinity -s -d -fwkall)

Use the `fw ctl affinity -s -d -fwkall` command to assign the specified number of CPU cores to all Virtual Systems at once.

Effect on Multi-queue settings for ixgbe interfaces

The use of this command to change the number of cores assigned to Virtual Systems, changes the number of cores available for `ixgbe` interface `rx queues`. Conversely, when you change the number of cores assigned to `ixgbe` interface queues, you also change the number of cores assigned to Virtual Systems.

For example, if your SGMs have 16 cores, and you assign 9 cores to Virtual Systems, the remaining 7 cores are available to the ixgbe interfaces.

Syntax

fw ctl affinity -s -d -fwkall <cores>

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-s</code> <code>-d</code></td>
<td>Set affinity for a VSX environment.</td>
</tr>
<tr>
<td><code>-fwkall</code> &lt;cores&gt;</td>
<td>Defines the number of cores assigned to all Virtual Systems.</td>
</tr>
</tbody>
</table>

Example

This example assigns three cores to Firewall instances for all Virtual Systems.

```
> fw ctl affinity -s -d -fwkall 3
VDevice 0-2 : CPU 0 1 2 – set successfully
```

Note: You can run this command from the vs0 context only.

Monitoring Process Affinity (fw ctl affinity -l -x)

You can monitor the affinity of processes and Virtual Systems on a VSX Gateway. You can use the `-vsid` parameter to show the affinity for a process to the specified Virtual Systems.

Syntax

```
> fw ctl affinity -l -x [-vsid <vsid>] [-flags [e|h|k|n|t|o]]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;vsid&gt;</code></td>
<td>Shows the affinity for processes for these Virtual System IDs. Use a dash to set a range of Virtual Systems.</td>
</tr>
<tr>
<td><code>e</code></td>
<td>Do not show processes that are affinity exceptions. You define affinity exceptions in the $FWDIR/conf/vsaffinity_exception.conf file.</td>
</tr>
<tr>
<td><code>h</code></td>
<td>Show CPU affinity mask in hexadecimal format.</td>
</tr>
<tr>
<td><code>k</code></td>
<td>Do not show kernel threads.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>n</td>
<td>Show the process name instead of /proc/&lt;pid&gt;/cmdline</td>
</tr>
<tr>
<td>t</td>
<td>Show information about process threads.</td>
</tr>
<tr>
<td>o</td>
<td>Print the list to a file.</td>
</tr>
</tbody>
</table>

**Example**

```bash
> fw ctl affinity -l -x -vsid 1 -flags tn
```

<table>
<thead>
<tr>
<th>PID</th>
<th>VSID</th>
<th>CPU</th>
<th>SRC</th>
<th>V</th>
<th>KT</th>
<th>EXC</th>
<th>NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>4756</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>pm</td>
</tr>
<tr>
<td>4773</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>confd</td>
</tr>
<tr>
<td>4774</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>searchd</td>
</tr>
<tr>
<td>5008</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---searchd</td>
</tr>
<tr>
<td>4780</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>httpd2</td>
</tr>
<tr>
<td>4781</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>monitord</td>
</tr>
<tr>
<td>24700</td>
<td>0</td>
<td>0 1 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---cpd</td>
</tr>
<tr>
<td>24704</td>
<td>0</td>
<td>0 1 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---cpd</td>
</tr>
<tr>
<td>24705</td>
<td>0</td>
<td>0 1 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---cpd</td>
</tr>
<tr>
<td>22800</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>mfpdaemon</td>
</tr>
<tr>
<td>24523</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fwx_forker</td>
</tr>
<tr>
<td>24525</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fwx_wd</td>
</tr>
<tr>
<td>24573</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fwx</td>
</tr>
<tr>
<td>24667</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fw</td>
</tr>
<tr>
<td>24668</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fw</td>
</tr>
<tr>
<td>24670</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fw</td>
</tr>
<tr>
<td>24671</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fw</td>
</tr>
<tr>
<td>25412</td>
<td>0</td>
<td>1 3 4 6 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fw</td>
</tr>
<tr>
<td>24642</td>
<td>0</td>
<td>2 3 4 5 6 7 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>fwx0_dev</td>
</tr>
<tr>
<td>24643</td>
<td>0</td>
<td>2 3 4 5 6 7 P</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>---fwm0_0</td>
</tr>
<tr>
<td>30186</td>
<td>0</td>
<td>all</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>clishd</td>
</tr>
</tbody>
</table>

**System Under Load**

**Description**

System Under Load feature (SUL) enables the Gateway to monitor high CPU load and also suspends setting remote SGMs to DOWN state when cannot receive CCP packets for a timeout of \texttt{BLADE\_DEAD\_INTERVAL} (default is 3 sec) and when SUL state ON.

It enables every SGM to act differently when they/other SGM are under load.

Being under load (SUL state ON) meaning at least one SGM has reported Kernel CPU Usage above threshold of 80% by default (CPU threshold)

Highest average Kernel CPU usage of a single core is being calculated locally and is published via CCP packets to remote SGMs

The average is based on 5 samples by default (Number of sample) – sample is taken every 2 HA Time Units (HTU=0.1s)

Every SGM calculates its own Kernel High CPU

Local Kernel High CPU usage and remote usage have almost the same handler with minor changes

- Local or Remote Kernel High CPU will set SUL state ON
- Local User space + Kernel High CPU will triggers PNOTE timeout postponer to all user-space PNOTEs (etc fwd) on local SGM
SUL state change

**SUL Feature flow**

- SUL set to ON - if reported high CPU
- SUL will set to OFF if no report has been received for at least 10 seconds by default from the last report (short timeout)
  - if system is continually under load (high CPU report gap is less than short timeout, SUL will stay ON for up to 3 minutes by default (Long interval)

**When / why SUL is ON?**

- Every SGM calculates CPU usage on all cores, picking the highest and stores in memory.
- On every CPU state check (called periodically) we take the average of recent 5 highest samples (Number of sample) and publish via CCP
- By receiving CCP with SGM CPU:
  - If > threshold (CPU threshold) → toggle SUL ON
- By calculating locally:
  a) If > threshold (CPU threshold) → toggle SUL ON
  b) local load is ON (for local user-space PNOTEs)

SUL ON mode will be delayed for a fixed timeout (Start timeout) (default=0) if at least one SGM continually reports high CPU more than 3min (Long interval) and the reason for setting OFF from the beginning was the long-timeout expiration

**When / why SUL is OFF?**

SUL can be toggle OFF after one of the following scenarios:

- System is idle - no SGM reported High CPU usage for at least 10 seconds (default timeout of Short timeout)
- System is Under Load for too long - after a fixed watermark of 3 minutes (Long interval) the SUL is in ON, it will be forced to toggle OFF, even if SGMs still reporting High CPU. SUL will be ON again if they will keep reporting high CPU after the shutdown but only after fix timeout – 0 by default is over (Start timeout)
- User decided to manually disable the feature while SUL was ON

**Syntax**

```
fw ctl set int fwha_pnote_timeout_mechanism_monitor_cpu <value>
```

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Turns SUL mechanism ON</td>
</tr>
<tr>
<td>1</td>
<td>Turns SUL mechanism OFF</td>
</tr>
</tbody>
</table>

**Example**

Enabling SUL feature: (SUL is enabled by default)

```
fw ctl set int fwha_pnote_timeout_mechanism_monitor_cpu 1
```

**Output**

Every state change (ON/OFF) is logged via SmartView Tracker & `/var/log/messages` (dmesg), when (only SMO sends the SVT messages)

Log Example in SmartView Tracker:
SUL feature can be modified and tuned to meet user specific needs.

Syntax

```
fw ctl set int <parameter> <numerical value>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fwha_pnote_timeout_mechanism_cpu_load_limit</td>
<td>(CPU threshold)</td>
</tr>
<tr>
<td></td>
<td>(highest average CPU usage of a single core)</td>
</tr>
<tr>
<td></td>
<td>default = 80</td>
</tr>
<tr>
<td>fwha_sul_num_sample_cpu_check</td>
<td>(Number of sample)</td>
</tr>
<tr>
<td></td>
<td>(on how many samples the CPU average will be based on; sample is taken every 2 HTUs)</td>
</tr>
<tr>
<td></td>
<td>default = 5</td>
</tr>
<tr>
<td></td>
<td>HTU - HA Time Unit (0.1s)</td>
</tr>
<tr>
<td>fwha_pnote_timeout_mechanism_disable_feature_timeout</td>
<td>(Long interval)</td>
</tr>
<tr>
<td></td>
<td>(maximum continues time allowed for SUL ON state)</td>
</tr>
<tr>
<td></td>
<td>default = 1800 HTU (3 minutes)</td>
</tr>
<tr>
<td></td>
<td>HTU - HA Time Unit (0.1s)</td>
</tr>
<tr>
<td>fwha_system_under_load_short_timeout</td>
<td>(Short timeout)</td>
</tr>
<tr>
<td></td>
<td>(low CPU usage period for setting SUL OFF)</td>
</tr>
<tr>
<td></td>
<td>default = 100 HTU (10 seconds)</td>
</tr>
<tr>
<td></td>
<td>HTU - HA Time Unit (0.1s)</td>
</tr>
<tr>
<td>fwha_system_under_load_start_timeout</td>
<td>(Start timeout)</td>
</tr>
<tr>
<td></td>
<td>(delay time between next SUL ON, if last ON period interrupted by Long interval)</td>
</tr>
<tr>
<td></td>
<td>default = 0 HTU (0 seconds)</td>
</tr>
<tr>
<td></td>
<td>HTU - HA Time Unit (0.1s)</td>
</tr>
</tbody>
</table>
Notes
In order for the modified SUL parameters, including state (ON/OFF) to survive reboot, add them to the fwkern.conf file using the g_update_conf_file utility

Working with Jumbo Frames

This release supports Jumbo Frames with payloads of up to 9,146 bytes for the SSM60 and 12,288 bytes for the SSM160. The configuration procedure for Jumbo Frames includes these steps:

1. Enable Jumbo Frames on the SGMs.
2. Configure Jumbo Frames on SGM interfaces.
3. Configure Jumbo Frames on the SSM.

Enabling Jumbo Frames (asg_jumbo_conf)

Use the asg_jumbo_conf command to enable or disable Jumbo Frames or to show the configuration status.

Syntax
asg_jumbo_conf {enable|disable|show} [-v]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>enable</td>
<td>Enable Jumbo Frames</td>
</tr>
<tr>
<td>disable</td>
<td>Disable Jumbo Frames</td>
</tr>
<tr>
<td>-v</td>
<td>Detailed report (verbose)</td>
</tr>
</tbody>
</table>

To enable Jumbo Frames, run this command in the Expert Mode:

# asg_jumbo_conf enable

To disable Jumbo Frames, run this command in the Expert Mode:

# asg_jumbo_conf disable

Example

# asg_jumbo_conf enable

Enabling Jumbo frames on SGMs
Enabling Jumbo Frames on SSMs
Chassis1
--------

Jumbo frames are enabled on SSM1
Jumbo frames are enabled on SSM2

Chassis2
--------

Jumbo frames are enabled on SSM1
Jumbo frames are enabled on SSM2
Jumbo frames enabled.
**Configuring Jumbo Frames on your SSMs**

**SSM160**

To configure Jumbo Frames on an SSM160, run this command for each SSM and port:

`asg_chassis_ctrl set_port_mtu <ssm_id> <port_id> <mtu_size>`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ssm_id&gt;</td>
<td>SSM identifier (1-4 or all)</td>
</tr>
<tr>
<td>&lt;port_id&gt;</td>
<td>Port number</td>
</tr>
<tr>
<td>&lt;mtu_size&gt;</td>
<td>This MTU size can be one of these values:</td>
</tr>
<tr>
<td></td>
<td>• Integer value up to 12,288</td>
</tr>
<tr>
<td></td>
<td>• max - Maximum supported MTU size</td>
</tr>
<tr>
<td></td>
<td>• default - System default MTU size (typically 1544)</td>
</tr>
</tbody>
</table>

**Examples**

> `asg_chassis_ctrl set_port_mtu 1 3 max`

MTU of port 3 on SSM1 was set to 12288

> `asg_chassis_ctrl set_port_mtu 2 4 9146`

MTU of port 4 on SSM2 was set to 9146

**SSM60**

Do this procedure for each SSM60 in the Chassis. In a Dual Chassis system, do this procedure for both Chassis.

1. Connect to the SSM with telnet: The default password is admin.
2. Run this command: to go to the Enable mode:

   ```
   # en
   ```
3. Run this command to go to the Configuration terminal:

   ```
   # conf t
   ```
4. Run this command to configure all the downlink interfaces:

   ```
   # interface range 1/2/1-1/14/1
   ```
5. Run this command to configure the MTU:

   ```
   # packet-size-limit 9146
   ```
6. Run this command to configure the required front panel ports:

   ```
   # interface range 1/2/1-1/14/1
   Interfaces 1/15/1 – 1/15/5 = SSM ports 1-5.
   ```
7. Run this command to set the required MTU:

   ```
   # packet-size-limit 9146
   ```
8. Run these commands to close Configuration terminal and save the configuration:

   ```
   # end
   # write
   ```
Example

# telnet 198.51.100.32
Trying 198.51.100.32...
Connected to 198.51.100.32.
Escape character is '^]'.

User Access Verification
Password:
> en
# conf t
# interface range 1/2/1-1/14/1
# packet-size-limit 9146
# interface range 1/15/1-1/15/5
# packet-size-limit 9146
# end
# write

Configuring SGMs (set interface)

You configure Jumbo Frames for each applicable interface on an SGM.

Syntax

```
set interface <if_name> mtu <size>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;if_name&gt;</td>
<td>Interface name as defined in the operating system</td>
</tr>
<tr>
<td>&lt;size&gt;</td>
<td>Maximum MTU size (9,124 for SSM60, 12,288 for SSM160)</td>
</tr>
</tbody>
</table>

Example

```
> set interface eth2-04 mtu 9000
```

Running Validation Tests

We recommend that you run validations for the SSMs, SGMs, and SGM interfaces before you use the system for production traffic.

SGMs and SGM Interfaces (asg_jumbo_conf show)

You use the `asg_jumbo_conf show` command to:

- Make sure that Jumbo Frames are enabled on the SGMs
- See the configured MTU values on SGM interfaces configured for Jumbo Frames

There is an option for a summary and detailed report.

Syntax

```
asg_jumbo_conf show [-v]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Detailed report (verbose)</td>
</tr>
</tbody>
</table>
Example

```
# asg_jumbo_conf show -v
Jumbo frames are enabled on SGMs (SSM1 max MTU: 12288 SSM2 max MTU: 12288)
Retrieving SSMs Jumbo frames configuration
Chassis1

SSMs:
Jumbo frames are enabled on SSM1
Jumbo frames are enabled on SSM2
Interfaces MTU configuration:
interface: BPEth0: mtu 12288
interface: BPEth1: mtu 12288
The MTU of all the interfaces which are not in the list is 1500
```

SSM160

To run the validation tests:

1. Run this command to show the Jumbo Frames configuration on the specified SSM:
   
   ```
   # asg_chassis_ctrl jumbo_frames show <ssm_id>
   ```

2. Run this command to show the configured MTU on the specified port.
   
   ```
   # asg_chassis_ctrl get_port_mtu <ssm_id> <port_id>
   ```

Example

```
# asg_chassis_ctrl jumbo_frames show 1
Jumbo frames are enabled on SSM1
# asg_chassis_ctrl get_port_mtu 1 1
MTU of port 1 on SSM1 is 9000
```

SSM60

To run the validation test:

1. Connect to the SSM with telnet

   The default password is admin.

   1. Run this command: to go to the Enable mode:

      ```
      # en
      ```

   2. Run this command to display the running configuration:

      ```
      # show run
      ```

   3. Make sure that all applicable interfaces (downlinks and front panel ports) show the required packet size limit.

      ```
      # telnet 198.51.100.32
      Trying 198.51.100.32...
      Connected to 198.51.100.32.
      Escape character is '^]'.
      
      User Access Verification
      Password:
      FI_cp>en
      #show run
      .
      .
      .
      !
      interface 1/2/1
      flow-control disable
      packet-size-limit 9146
      !
      ```
TCP MSS Adjustment

**Description**

TCP MSS Adjustment allows MSS (Maximum Segment Size) clamping of TCP traffic. This enables the configuration of the MSS that is part of the OPTIONS in the TCP header. This feature provides a method to prevent fragmentation when the MTU value on the communication path is lower than the MSS value.

**Syntax**

```
fw ctl set int <fw_clamp_tcp_mss|fw_tcp_mss_value> <num>
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fw_clamp_tcp_mss &lt;num&gt;</td>
<td>• Enable or Disable MSS Adjustment:</td>
</tr>
<tr>
<td></td>
<td>• 0, Disable (default)</td>
</tr>
<tr>
<td></td>
<td>• 1, Enable</td>
</tr>
<tr>
<td>fw_tcp_mss_value &lt;num&gt;</td>
<td>• Set the MSS value. If value is set to 0, the MSS</td>
</tr>
<tr>
<td></td>
<td>value is taken from the interface MTU</td>
</tr>
</tbody>
</table>

**Note:** In order for the modified parameters, including state (ON/OFF), to survive reboot - add them to the `$FWDIR/boot/modules/fwkern.conf` file using `g_update_conf_file` utility from Expert shell.

**Verification**

Monitoring can be done using Packet Sniffers to verify that indeed MSS is clamped when the feature is enabled according to configuration.

**Note:** MSS value is applied on all interfaces, including Management

**Debug**

1. Enable SIM debug using the command: `sim dbg -m pkt + pkt`
2. Start fw debugging using the command: `fw ctl zdebug + packet`
3. Look for prints that contain the string MSS

---

Working with Session Control (asg_session_control)

**Description**

Use the `asg_session_control` command to set the rate at which new communication sessions are opened, based on a predefined set of rules. This functionality is also known as Session Rate Throttling. You can only run `asg_session_control` from the Expert shell.

You create session control rules in the `$FWDIR/conf/control_rules` file. Session rate control is disabled by default.

**Syntax**

```
# asg_session_control <apply | disable | stats | verify>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No parameters</td>
<td>Shows command syntax and helpful information</td>
</tr>
<tr>
<td>apply</td>
<td>Applies session rate rules to all SGMs</td>
</tr>
<tr>
<td>disable</td>
<td>Disables session rate rules for all SGMs</td>
</tr>
<tr>
<td>stats</td>
<td>Shows all session rate rules and dropped traffic statistics</td>
</tr>
<tr>
<td>verify</td>
<td>Makes sure that the session rate rules are the same on all SGMs</td>
</tr>
</tbody>
</table>
Defining Session Control Rules

You define session rate rules in the `$FWDIR/conf/control_rules` file. Use one line for each rule. Each rule must contain the `limit` parameter. The other parameters are optional.

**Important** - Define rules as specifically as possible, so that more than one rule cannot apply to the same traffic. Overlapping rules can cause unpredictable results. We recommend that you explicitly define all parameters in each rule.

**Rule Syntax**
```
src <ip/mask> dst <ip/mask> dport <port> proto <protocol_id> limit <rate>
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>src &lt;ip/mask&gt;</td>
<td>Source IP address and net mask.</td>
</tr>
<tr>
<td>dst &lt;ip/mask&gt;</td>
<td>Destination IP address and net mask.</td>
</tr>
<tr>
<td>dport &lt;port&gt;</td>
<td>Destination port.</td>
</tr>
<tr>
<td>proto &lt;protocol_id&gt;</td>
<td>Protocol code, typically 6 (TCP) or 17 (UDP). To learn more about protocol codes, IANA protocol codes (<a href="http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml">http://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml</a>).</td>
</tr>
<tr>
<td>limit &lt;rate&gt;</td>
<td>Maximum number of new connections allowed per second.</td>
</tr>
</tbody>
</table>

**Rule Examples**
```
src * dst 1.1.1.0/24 dport 67 proto 17 limit 20
```
This rule defines a limit of 20 new connections per second for traffic going from any source to:
- Network 1.1.1.0/24
- Port 67
- Using protocol 17 (UDP)

```
dst 1.1.1.1/32 dport 80 proto 6 limit 13
```
This rule defines a limit of 13 new connections per second for traffic going from any source to:
- Network 1.1.1.1/32
- Port 80
- Using protocol 6 (TCP)

**Notes**
- New connections in excess of the specified limit are dropped.
- If you do not include a parameter, the rule applies to all values for that parameter. For example, if you do not include the `src` parameter, the rule applies to all servers.
- The `*` character as a parameter value explicitly says that a rule applies to all values.

Enabling and Disabling Session Control

**To enable Session Control:**
1. Define Session Control rules.
2. Run `# asg_session_control apply`.

**To disable Session Control:**
Run `# asg_session_control disable`. 
asg_session_control disable
**--*--2 blades: 1_01 1_02 --*--**
Resetting session rate entries
Session rate entries configured successfully

**Applying Session Control Rules**

```bash
# asg_session_control apply
**--*--2 blades: 1_01 1_02 --*--**
```

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>Source</th>
<th>Destination</th>
<th>DPort</th>
<th>PR</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
<td>1.1.1.0/24</td>
<td>67</td>
<td>17</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>*</td>
<td>2.2.2.2/32</td>
<td>80</td>
<td>6</td>
<td>13</td>
</tr>
</tbody>
</table>

The output shows the Session Control rules applied.

**Showing Session Control Statistics**

```bash
# asg_session_control stats
```

1_01:

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>Source</th>
<th>Destination</th>
<th>DPort</th>
<th>PR</th>
<th>Limit</th>
<th>Drops</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
<td>1.1.1.0/24</td>
<td>67</td>
<td>17</td>
<td>20</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>*</td>
<td>2.2.2.2/32</td>
<td>80</td>
<td>6</td>
<td>13</td>
<td>0</td>
</tr>
</tbody>
</table>

1_02:

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>Source</th>
<th>Destination</th>
<th>DPort</th>
<th>PR</th>
<th>Limit</th>
<th>Drops</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>*</td>
<td>1.1.1.0/24</td>
<td>67</td>
<td>17</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>*</td>
<td>2.2.2.2/32</td>
<td>80</td>
<td>6</td>
<td>13</td>
<td>2</td>
</tr>
</tbody>
</table>

The output shows the session control rules for each SGM and the connections dropped by each rule.

**Hide NAT Behind Range – Sticky per SGM**

*(asg_hide_behind_range)*

This feature uses the capability of hide NAT behind range to increase the amount of hide NAT ports per SGM.

When defining NAT rules with a range of translated sources, each SGM can receive a separate hide NAT address, and therefore can use a full range of hide NAT ports (instead of the range being divided between the SGMs).

**Note:** To safely use this feature, the security policy must be configured such that every NAT rule uses a range object (of at least 24 addresses) as a translated source (see comments).

**Syntax**

```bash
asg_hide_behind_range [-v|-s|on|off]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Make sure that the current policy does not contain hide NAT rules with a translated source smaller than 24 addresses.</td>
</tr>
<tr>
<td>-s</td>
<td>Show current status</td>
</tr>
<tr>
<td>on</td>
<td>Enable feature</td>
</tr>
<tr>
<td>off</td>
<td>Disable feature</td>
</tr>
</tbody>
</table>
Example

> asg_hide_behind_range on

Configuration succeeded.

Note: In order to apply the changes all SGMs must be rebooted.

Important:
This feature will only affect NAT rules which have a range of at least 24 addresses defined as the translated source.

Note: Manual NAT rules require local.arp configuration.

Notes

- Changes are applied after a reboot.
- Hide NAT behind range rules are manual NAT rules (see Proxy ARP for Manual NAT).
- It is not guaranteed that a given source address will always be translated to the same NAT address. This is only a certainty if all connections from the source address are handled by the same SGM.
- Hide NAT rules with a translated source that is either a range smaller than 24 addresses, or a single hide address, are not compatible with this feature. The above applies to implied rules as well.
- If the security policy contains such rules, it is not guaranteed that each SGM will hide traffic that matches them behind an address different than all other SGMs. This may result in port conflicts; e.g. different connections might appear as one and the same after NAT, both in terms of IP address and source port.

Acceleration Not Disabled Because of Traceroute Rule
(asg_tmpl_special_svcs)

This feature safely prevents security policy rules with the traceroute service from disabling acceleration for all subsequent rules.

Syntax

asg_tmpl_special_svcs [on|off]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>on</td>
<td>Acceleration is not disabled because of traceroute rules</td>
</tr>
<tr>
<td>Off</td>
<td>Acceleration is disable because of traceroute rules</td>
</tr>
</tbody>
</table>

Example

asg_tmpl_special_svcs

- This functionality requires a patch on the Management side. To get it, contact Check Point Support.
- For this feature to function correctly, the traceroute service object in SmartDashboard must remain with default settings and not customized.

Improving the Performance of Inbound HTTPS

You can improve the performance of inbound HTTPS traffic. That is, traffic from outside the organization to internal HTTPS servers.

To Improve the performance of Inbound HTTPS:

Run

fw ctl set int choose_active_streaming 0

To restore the default HTTPS performance settings:

Run:

fw ctl set int choose_active_streaming 1
Supported SSL Ciphers

These SSL ciphers are supported on internal HTTPS servers when the parameter `choose_active_streaming` is set to 0:

- RSA+AES
- RSA+RC4
- RSA+3DES

You must update the list of supported SSL ciphers on the protected HTTPS servers.
Chapter 4

LTE Features

This release adds many new features that support for advanced LTE telecommunication. Most of these new features are configured with SmartDashboard or on the management server. See the R76 LTE Release Notes (http://downloads.checkpoint.com/dc/download.htm?id=29339) for detailed information and configuration procedures. Configuration procedures for SGMs are included in this section for your convenience.

These LTE features are included in this release.

- LTE S1 VPN
- Firewall GX support
- GTPv2 support
- GTP CoreXL support
- GTP Signaling rate limit
- SCTP support
- Diameter inspection
- Third-Party Syslog
- MSS adjustment
- CGNAT
- Stateless NAT46 translation
- NAT 64
- Large Scale VPN

In This Section:

- Enabling LTE Support
- VPN Sticky SA (for LTE)
- Configuring SCTP Acceleration on SGMs
- Configuring SCTP NAT on SGMs

Enabling LTE Support

LTE configuration includes hundreds or thousands of eNodeB VPN peers. Each eNodeB has its own IPSec tunnel to the 61000/41000 Security System. eNodeB encrypts GTP traffic from mobile clients behind the eNodeB.

You must enable LTE support to use LTE features and S1 VPN.

To enable LTE support for all SGMs:


   **Note:** If not all SGMs are in UP state while running this command (e.g. not all SGMs are present in the Chassis), copy `$CPDIR/tmp/.CPprofile.sh` from the SGM you ran the command from, to the newly added SGMs.

2. Run `reboot -b all`.
VPN Sticky SA (for LTE)

To support LTE environments, you must enable the VPN sticky Security Association (SA) feature. This feature makes sure that an LTE device has only one outgoing SA to the 61000/41000 Security System, which is a requirement for an LTE device.

Limitations

- Connections are synchronized to all SGMs (instead synchronizing only to the backup SGM).
- Third-party VPN peers are not enabled by default.

⚠️ Important - You must not enable SPI distribution and Sticky SA ("VPN Sticky SA (for LTE)" on page 196) at the same time.

Configuration

SGMs are typically configured for sticky SA by default during LTE configuration. You must enable LTE support to use LTE features.

To configure this feature without configuring LTE:

1. Run from the Expert mode:
   ```
   # g_update_conf_file $FWDIR/modules/fw Kern.conf
   fwha_vpn_sticky_tunnel_enabled=1
   ```
2. Reboot all SGMs:
   ```
   # reboot -b all
   ```

Verification:

If SecureXL is enabled, make sure that the **VPN Sticky Tunnel Enabled** parameter is set to **yes** in the
```
# /proc/ppk/conf file. To do so, run this command from the Expert node:
```
```
# g_cat /proc/ppk/conf | grep VPN
```

Configuring SCTP Acceleration on SGMs

To enable SCTP acceleration, run this command in gclish:
```
> sim feature sctp on
```
To disable SCTP acceleration, run this command in gclish:
```
> sim feature sctp off
```

Notes:

- You must configure SCTP in SmartDashboard before you can use this feature. See the R76 LTE Release Notes (http://downloads.checkpoint.com/dc/download.htm?ID=29339) for detailed information and configuration procedures.
- If SCTP acceleration is activated and SCTP inspection is deactivated, the Performance Pack accelerates all SCTP packet types.

Configuring SCTP NAT on SGMs

SCTP NAT overrides the currently defined NAT policy. When this feature is not activated, SCTP connections do not use NAT.

To activate SCTP NAT, run this command in gclish:
```
> fw ctl set int fwx_enable_sctp_nat 1
```
To deactivate SCTP NAT, run this command in gclish:
```
> fw ctl set int fwx_enable_sctp_nat 0
```
Chapter 5

61000/41000 Security System Concepts

In This Section:

- Single Management Object and Policies ................................................................. 197
- SGM Policy Management ......................................................................................... 200
- MAC Addresses and Bit Conventions ................................................................. 202
- SyncXL ......................................................................................................................... 203
- Security Group (asg security_group) ................................................................. 204
- Working with the Distribution Mode ................................................................. 204
- NAT and the Correction Layer on Security Gateway .................................. 210
- NAT and the Correction Layer on a VSX Gateway ........................................ 210
- Hybrid System ........................................................................................................ 213
- GARP Chunk Mechanism .................................................................................. 214

Single Management Object and Policies

*Single Management Object* is a Check Point technology that manages the 61000/41000 Security System as one large Security Gateway with one management IP address. All management tasks, are handled by one SGM (the SMO Master), which updates all other SGMs. All management tasks, such as Security Gateway configuration, policy installation, remote connections and logging are handled by the SMO master. The active SGM with the lowest ID number is automatically assigned to be the SMO.

Use this command to identify the SMO and see how tasks are distributed on the SGMs:

```
> asg stat -i tasks
Chassis ID: 1
----------
Task (Task ID)          SGM ID
General    (1)          3
LACP       (2)          4
CH Monitor (3)          5

Chassis ID: 2
----------
Task (Task ID)          SGM ID
SMO        (0)          2(local)
DR Manager (4)          2(local)
General    (1)          3
LACP       (2)          4
CH Monitor (3)          5
```
Installing and Uninstalling Policies

To install a policy on the 61000/41000 Security System, select **Policy > Install** in SmartDashboard. The installation procedure includes these steps:

1. The Security Management server installs the policy on the SMO Master.
2. The SMO copies the policy to all SGMs.
3. Each SGM installs the policy locally

During the installation, each SGM sends and receives policy status updates to/from the other SGMs. This is because the SGMs need to install their policies in a synchronized manner. Policy installation has these stages:

- **Policy Started** - Policy installation started on the SGM.
- **Policy Ready2Finish** - Policy installation is completed, but the SGM is waiting for other SGMs to reach the same stage.
- **Policy Completed** - The policy is synchronized with the other SGMs.
- **Enforcing Security** - The SGM enforces the new policy.

**Note** - When installing the 61000/41000 Security System, SGMs enforce an initial policy where only the implied rules necessary for management are enforced.

To Uninstall Policy, open a serial connection to the 61000/41000 Security System and run:

```plaintext
> asg policy unload
```

**Notes:**

- You cannot uninstall policies with SmartDashboard.
- To learn more about the working with policies, see **asg policy** (*Working with Policies (asg policy)* on page 198).

Working with Policies (asg policy)

Use the **asg policy** command to do these policy-related actions:

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verify</td>
<td>Make sure that the correct policies are installed on all SGMs.</td>
</tr>
<tr>
<td>verify_amw</td>
<td>Makes sure that the correct Anti-malware policies are installed on all SGMs.</td>
</tr>
<tr>
<td>unload</td>
<td>Uninstall the policy from SGMs.</td>
</tr>
</tbody>
</table>

**Syntax**

```plaintext
asg policy -h
asg policy verify|verify_mw [-vs <vs_ids>] [-a] [-vs] [-v]
asg policy unload [--disable_pnotes] [-a]
```

**Parameter**

- **-h**  
  Show syntax and help information.

- **-vs <vs_ids>**  
  Shows verification results for each Virtual System. The `<vs_ids>` can be:
  - No `<vs_ids>` (default) - Shows the current Virtual System context.
  - One Virtual System.
  - A comma-separated list of Virtual Systems (1,2,4,5).
  - A range of Virtual Systems (VS 3-5).
  - all - Shows all Virtual Systems.

**Note**: This parameter is only relevant in a VSX environment.
Example - Detailed Virtual System Output

```
$ asg policy verify -vs all -v
+------------------------------------------------------------------------------+-
| Policy Verification                                                          |
+------------------------------------------------------------------------------;-
| VS  | SGM  | Policy Name | Policy Date | Policy Signature | Status |
+----------------------------------------------------------------------------+-
| 0   | _01  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _03  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _04  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _05  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _06  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _11  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
|     | _12  | Standard    | 26Nov12 21:11 | 996eee5e6       | Success |
| 1   | _01  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _03  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _04  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _05  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _06  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _11  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
|     | _12  | Standard    | 27Nov12 13:03 | 836fa2ec1       | Success |
| 2   | _01  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _03  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _04  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _05  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _06  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _11  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
|     | _12  | Standard    | 26Nov12 21:11 | 10eef9ced       | Success |
+------------------------------------------------------------------------------;-

| Summary |---------------------------------------------------------------|
+----------+---------------------------------------------------------------|
| Policy Verification completed successfully                       |
+------------------------------------------------------------------|

Example - Uninstall Policy

```
$ asg policy unload
You are about to perform unload policy on blades: all
All SGMs will be in DOWN state, beside local SGM. It is recommended to run the procedure via serial connection
Are you sure? (Y - yes, any other key - no) y
```

Unload policy requires auditing
Enter your full name: ploni
Enter reason for unload policy [Maintenance]: WARNING: Unload policy on blades: all, User: ploni, Reason: Maintenance
+----------+-
| Unload policy | |
+----------+---
| SGM | Status |
+-----+-------
| _1_ | Success |
+-----+-------
| _2_ | Success |
+-----+-------
| _3_ | Success |
+-----+-------
| _2_ | Success |
+-----+-------
| _1_ | Success |
+-----+-------

| Summary |---------------------------------------------------------------|
+----------+---------------------------------------------------------------|
| Policy completed policy успешно                          |
+-----------------------------------------------------------|

Note - It is recommended to run this command over a serial connection.
SGM Policy Management

Because the 61000/41000 Security System works as one large Security Gateway, all SGMs are configured with the same policy. When you install a policy from the management server, it first installs the policy on the SMO. The SMO copies the policy and SGM configuration to all SGMs in the Up state. When an SGM enters the Up state, it automatically gets the currently installed policy and configuration from the SMO. If there is no SMO (when there is only one SGM in the Up state), that SGM uses its local policy and configuration.

If there are problems with the policy or configuration on an SGM, you can manually copy the information from a different SGM.

An SGM configuration has these components:

- The Firewall policy, which includes the Rulebase.
- Set of configuration files defined in the /etc/xfer files list file. This file contains the location of all related configuration files. It also defines the action to take if the copied file is different from the one on the local SGM.

**Copying the Policy and Configuration (asg_blade_config pull_config)**

Use this command to manually copy the policy and, optionally, the configuration files from a specified SGM to the local SGM. The physical copy operation occurs automatically when you reboot the SGM or run these commands:

- cpstart
- asg sgm_admin up

**Syntax**

`asg_blade_config pull_config all <sdm_sync_ip >`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;ip&gt;</td>
<td>Remote SGM synchronization IP address</td>
</tr>
<tr>
<td>policy</td>
<td>Get only the policy</td>
</tr>
<tr>
<td>vsx_conf</td>
<td>Get the VSX configuration</td>
</tr>
<tr>
<td>all</td>
<td>Run full synchronization: Policy, VSX configuration, Firewall configuration</td>
</tr>
<tr>
<td>force</td>
<td>Ignore the defined configuration group</td>
</tr>
</tbody>
</table>

**Note** - If necessary, run `asg stat -i all_sync_ips` to get a list of all SGM synchronization IP addresses.

**Example:** Copy the Firewall policy only

```
# asg_blade_config pull_config policy 192.168.2.21
```

**Example:** Full Synchronization

```
# asg_blade_config pull_config all 192.168.2.21
```

**Example:** Copy the VSX configuration

```
# asg_blade_config pull_config policy 192.168.2.21
```

**Understanding the Configuration File List**

The xfer_file list file contains pointers to the related configuration files on an SGM. Each record defines the path to a configuration file, followed by the action to take if the imported file is different from the local file. This table shows an example of the record structure.
### Context

<table>
<thead>
<tr>
<th>Context</th>
<th>File name and path</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>global_context</td>
<td>$FWDIR/modules/fwkern.conf</td>
<td>/bin/false</td>
</tr>
</tbody>
</table>

The context field defines the type of configuration file:

- **global_context** - Security Gateway configuration file
- **all_vs_context** - Virtual Systems configuration file

The action field defines that action to be taken when the imported (copied) file is different that the local file:

- **/bin/true** - Reboot is required
- **/bin/false** - No reboot is required
- **String enclosed in double quotes** - Name of a "callback script" that selects the applicable action.

### Example of a configuration file list:

```
global_context  $PPKDIR/boot/modules/sim_aff.conf "sim affinityload"
global_context  $PPKDIR/boot/modules/simkern.conf /bin/false
all_vs_context  $FWDIR/conf/fwauthd.conf /bin/false
all_vs_context  $FWDIR/conf/discndt.if /bin/false
global_context  /var/opt/fw.boot/ha_boot.conf /bin/false
all_vs_context  $FWDIR/conf/sync_exceptions_tab "g_sync_exception -f"
all_vs_context  $FWDIR/bin/reserved_conns_tab "g_reserved_conns -f"
global_context  /config/active  /usr/bin/confd_clone /config/db/cloned_db
global_context  /etc/passwd /bin/true
global_context  /etc/shadow /bin/true
global_context  /etc/smd_user.conf "smd restart"
global_context  /etc/smd_admin.conf "smd restart"
all_vs_context  $FWDIR/bin/iproute.load /bin/true
all_vs_context  $FWDIR/conf/gre_loader.conf /bin/true
all_vs_context  $FWDIR/conf/fwha_ch_uptime /bin/true
all_vs_context  $FWDIR/modules/mq_aff.conf "mq affinity -s"
all_vs_context  $FWDIR/conf/pingable_hosts.conf "pingable_hosts local on"
all_vs_context  $FWDIR/conf/pingable_hosts.ips /bin/true
all_vs_context  $FWDIR/conf/alert.conf /bin/true
all_vs_context  $FWDIR/conf/asg_log_servers.conf "log_servers_util refresh"
all_vs_context  $FWDIR/modules/vlan mq.conf "vlan perf enhancement -c"
all_vs_context  $FWDIR/modules/vlan_mq.conf "vlan_perf_enhancement -c"
all_vs_context  $FWDIR/conf/fw_global_params.conf "cpha blade_config fw_global_params_changed"
global_context  $FWDIR/boot/mq.conf "cpmq reconfigure"
global_context  /etc/modprobe.conf asg_update_modprobe_conf /tmp/modprobe.conf.new
global_context  $FWDIR/boot/modules/vpnkern.conf /bin/false
global_context  /etc/ssm_port_speed.conf /bin/asg_update_port_speed /tmp/ssm_port_speed.conf.new
all_vs_context  $FWDIR/conf/selective_template_exclude.conf /bin/true
global_context  /etc/syslog_servers list.conf asg_syslog_helper
all_vs_context  $FWDIR/conf/vsaffinity_exception.conf /bin/false
all_vs_context  $FWDIR/conf/manual.affinity.conf "check smo affinity files manual"
global_context  $FWDIR/conf/fwkall.affinity.conf "check smo affinity files fwdir" $FWDIR/tmp/
all_vs_context  $CPDIR/conf/*.affinity.conf "check smo affinity files cpdir" $CPDIR/tmp/
global_context  $FWDIR/conf/resctrl "$FWDIR/bin/fw vsx resctrl load_configuration"
```

---

61000/41000 Security System Administration Guide R76SP | 201
MAC Addresses and Bit Conventions

MAC Addresses

MAC addresses divide into three types:

- **BMAC.** A MAC address assigned to all interfaces with the "BPEthX" naming convention. Unique per member. It does not rely on the interface index number.
- **VMAC.** A MAC address assigned to all interfaces with "ethX-YZ" naming convention. Unique per Chassis, it does not rely on the interface index number.
- **SMAC.** A MAC address assigned to Sync interfaces. Unique per member, it does not rely on the interface index number.

Bit Conventions

**BMAC**

- 1 - 1 bit stating if this address is BMAC/SMAC(0) or VMAC(1) to avoid possible collision with VMAC space.
- 2,...,8 - 7 bits that state the member ID (starting from 1) - limited to 127 members
- 9,...,13 - zero bits.
- 14 - 1 bit stating if this address is BMAC(0) or SMAC(1) to avoid possible collision with SMAC space
- 15,16 - 2 bits that state the absolute interface number (taken from interface name: i.e. in BPEthX, X is the interface number - limited to four interfaces.)

**SMAC**

- 1 - 1 bit stating if this address is BMAC/SMAC(0) or VMAC(1) to avoid possible collision with VMAC space
- 2,...,8 - 7 bits that state the member ID (starting from 1) - limited to 127 members
- 9 - 1 bit stating whether it is Sync1(0) or Sync2(1)
- 9,...,13 - zero bits
- 14 - 1 bit stating if this address is BMAC(0) or SMAC(1) to avoid possible collision with BMAC space
- 15 - Zero bit
- 16 - 1 bit stating whether it is Sync1(0) or Sync2(1)

**VMAC**

- 1 - 1 bit stating if this address is BMAC/SMAC(0) or VMAC(1) to avoid possible collision with BMAC/SMAC space
- 2,...,3 - 2 bits to indicate Chassis id (starting from 0) - limited to 4 boxes
- 4,...,8 - 5 bits to indicate switch number - limited to 32 switches
- 9,...,16 - 8 bits to indicate port number - limited to 256 ports per switch.

**MAC Address Resolver (asg_mac_resolver)**

Description

All three types of MAC address (BMAC, VMAC,SMAC) can be verified using the `asg_mac_resolver` utility. From the given MAC address, `asg_mac_resolver` determines the:

- MAC type
- Chassis ID
- SGM ID
- Assigned interface
Syntax

`asg_mac_resolver <mac_address>`

Example

`asg_mac_resolver 00:1C:7F:01:00:FE`

```plaintext
[00:1C:7F:01:00:FE, BMAC] [Chassis ID: 1] [SGM ID: 1] [Interface: BPEth0]
```

Notes

- The specified MAC Address comes from the BPEth0, on SGM 1 on Chassis 1.
- 00:1C:7F:01:00:FE is the Magic MAC attribute, which is identified by FE.
- The index is 16 bits (2 Bytes) identified by 01:00 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16.

SyncXL

SyncXL is a Check Point technology that makes sure active connections are only synchronized to one SGM on the Active Chassis and one SGM on the Standby Chassis. This means that the SGM in the Active Chassis can only synchronize with its counterpart in the Standby Chassis.

When an SGM or Chassis state changes, all SGMs update their counterpart SGM. Synchronization is triggered automatically by these events:

- **SGM Failure** – Connections with a backup connection on an SGM are synchronized to a backup SGM
- **SGM Recovery** – The newly recovered SGM can be a backup for connections that are active on other SGMs
- **Chassis HA failover** – When the Active Chassis fails over to the Standby Chassis, a backup entry is defined for each of the connections that it handles.

The SyncXL mechanism can be configured via the `asg_sync_manager`. See the `asg_sync_manager` section.

**Standby Chassis/Active SGMs ratio:**

To handle load and capacity, the Standby Chassis must have at least 50% of SGMs in the UP state, compared with the Active Chassis. For example, if there are 10 SGMs that are UP on the Active Chassis, there must be at least five UP SGMs on the Standby Chassis. SyncXL is automatically disabled if this condition is not successful. You can change the ratio parameter ("asg_sync_manager" on page 113).

To make sure that each active connection has backups on both Chassis in a Dual Chassis system, run in the Expert mode:

`asg_sync_manager ('Searching for a Connection (asg search)" on page 50)

To see the last connection backup operation, run this command in the Expert mode:

`# asg_blade_stats ('Showing SGM Forwarding Statistics (asg_blade_stats)" on page 36)

Last Iterator Statistics:

```
Start time: Thu Sep 13 10:48:18 2012
Running time: 0 Seconds
Status: Finished
Reason: Chassis ID 2 state was changed to STANDBY
Total connections iterated 38
Connections w/ sync action 0
```

Notes:

- VoIP connections are synchronized to all SGMs
- Local connections (To/from the 61000/41000 Security System pseudo IP) are not synchronized
- SyncXL does not work on the Sync interface or the Management Interface
Security Group (asg security_group)

Description
To be part of the Security Gateway, an SGM must belong to the Security Group. SGMs are added to the Security group using the `asg security_group` command. SGMs in the security group:

- Are selected during the initial installation procedure (after running `#setup`)
- Are automatically installed once installation of the first SGM has completed
- Can be changed by using the `asg security_group` command

Syntax
`asg security_group`

Example
```
> asg security_group
```

Output
```
+--------------------------------------+
|       Security Group Utility         |
+--------------------------------------+

Current Security Group:

```
+------------------------+
<table>
<thead>
<tr>
<th>Chassis</th>
<th>Security Gateway Modules</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1,2,3</td>
</tr>
<tr>
<td>2</td>
<td>1,2,3</td>
</tr>
</tbody>
</table>
```

Choose one of the following options:
```
1) Add SGMs to Security Group
2) Remove SGMs from Security Group
3) Exit
```

Notes
Select which SGMs should be added or removed from the security group. Note that:

- An SGM added to the security group automatically joins the single management object of the Security Gateway and then reboots
- Before you remove an SGM from the security gateway, make sure that its state is DOWN.
- To optimize connection distribution amongst the SGMs, keep the security group updated with the actual number of SGMs in the appliance.

⚠️ Important - Run: `asg security_group verify` to make sure that the security group is correctly configured.

Working with the Distribution Mode

The Distribution Mode is the way that an SSM assigns incoming traffic to SGMs. These are the supported Distribution Modes:

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
<th>Applies to</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>Packets are assigned to an SGM based on the packet destination.</td>
<td>An SSM</td>
</tr>
<tr>
<td>Network</td>
<td>Packets are assigned to an SGM based on the packet source.</td>
<td>An SSM</td>
</tr>
<tr>
<td>Mode</td>
<td>Description</td>
<td>Applies to</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>General</td>
<td>Packets are assigned to an SGM based on both the packet source and destination.</td>
<td>All SSMs in the 61000/41000 Security System</td>
</tr>
<tr>
<td>Per-Port</td>
<td>Each SSM data interface is configured separately as User mode or Network mode.</td>
<td>SSM data interface</td>
</tr>
</tbody>
</table>

**Note:** User and Network modes always work together and are known collectively as the User/Network mode.

By default, the 61000/41000 Security System automatically configures the Distribution Mode. You can manually assign the General mode as necessary. There can be some scenarios where you must manually assign the General mode. The system does not automatically assign the General mode, with these exceptions:

- For Security Gateway deployments, the General mode is automatically assigned if there is at least one Bridge Mode interface.
- For VSX environments, the General mode is automatically assigned if there is at least one Virtual System configured in the Bridge mode.

**Automatic Distribution Configuration (Auto-Topology)**

By default, the 61000/41000 Security System automatically configures the Distribution Mode. The optimal Distribution Mode is derived from the Gateway topology as defined in SmartDashboard.

The Distribution Mode is derived from these interfaces:

- Physical, other than the management and sync.
- VLAN
- Bond
- VLAN over bond
- Bridge

These examples show how the distribution Mode can be automatically configured for each interface.

**Physical Interfaces**

<table>
<thead>
<tr>
<th>Physical Interface</th>
<th>Topology</th>
<th>SSM</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth1-01</td>
<td>Internal</td>
<td>1</td>
<td>User</td>
</tr>
<tr>
<td>eth1-02</td>
<td>Internal</td>
<td></td>
<td></td>
</tr>
<tr>
<td>eth2-01</td>
<td>External</td>
<td>2</td>
<td>Network</td>
</tr>
<tr>
<td>eth2-02</td>
<td>External</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In this example ports on each SSM are either all Internal or all External. Therefore, the Distribution Mode for the two SSMs is automatically configured as **User** or **Network**.

**Physical interfaces**

<table>
<thead>
<tr>
<th>Interface</th>
<th>Topology</th>
<th>SSM</th>
<th>Port</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth1-01</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>User</td>
</tr>
<tr>
<td>eth1-02</td>
<td>External</td>
<td>1</td>
<td>2</td>
<td>Network</td>
</tr>
<tr>
<td>eth2-01</td>
<td>External</td>
<td>2</td>
<td>1</td>
<td>Network</td>
</tr>
<tr>
<td>eth2-02</td>
<td>External</td>
<td>2</td>
<td>2</td>
<td>Network</td>
</tr>
</tbody>
</table>
On at least one of the SSMs, some ports are Internal and others are External. Therefore, the Distribution Mode for the SSMs is automatically configured as **Per Port**.

### Physical and VLAN interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Topology</th>
<th>SSM</th>
<th>Port</th>
<th>VLAN</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth1-01</td>
<td>External</td>
<td>1</td>
<td>1</td>
<td>NA</td>
<td>Network</td>
</tr>
<tr>
<td>eth1-01.100</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>100</td>
<td>User</td>
</tr>
<tr>
<td>eth1-01.200</td>
<td>External</td>
<td>1</td>
<td>1</td>
<td>200</td>
<td>Network</td>
</tr>
<tr>
<td>eth1-01.300</td>
<td>Internal</td>
<td>1</td>
<td>1</td>
<td>300</td>
<td>User</td>
</tr>
</tbody>
</table>

Three VLANs are defined on one SSM port. On at least one of the SSMs, some VLANs are Internal and others are External. Therefore, the Distribution Mode of the SSMs is automatically configured to be Per-Port. **Note:** Not supported in SSM60. In an SSM60 the Distribution Mode of all the VLANs on each port must be the same as the Distribution Mode of the port.

### VSX Virtual Systems

<table>
<thead>
<tr>
<th>Interface</th>
<th>Topology</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth1-01</td>
<td>External</td>
<td>N/A</td>
</tr>
<tr>
<td>wrpj64</td>
<td>Internal</td>
<td>Network</td>
</tr>
<tr>
<td>wrpj128</td>
<td>Internal</td>
<td>Network</td>
</tr>
<tr>
<td>wrpj192</td>
<td>Internal</td>
<td>User</td>
</tr>
</tbody>
</table>

Because a Virtual Switch does not have topology, the Distribution Mode is calculated based on the topologies of the WARP interfaces connected to the Virtual Systems, as shown. In this example, the Distribution Mode is calculated to be **Network**.

### Bond interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Topology</th>
<th>Slaves</th>
<th>SSM</th>
<th>Port</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond1</td>
<td>Internal</td>
<td>eth1-01</td>
<td>1</td>
<td>1</td>
<td>User</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eth2-01</td>
<td>2</td>
<td>1</td>
<td>User</td>
</tr>
<tr>
<td>bond2</td>
<td>External</td>
<td>eth1-02</td>
<td>1</td>
<td>2</td>
<td>Network</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eth2-02</td>
<td>2</td>
<td>2</td>
<td>Network</td>
</tr>
</tbody>
</table>

Bond interface bond1 is defined on two SSM1 and SSM2 ports. bond2 is defined on another two SSM1 and SSM2 ports. On at least one of the SSMs, some ports are Internal and others are External. Therefore, the Distribution Mode of the SSMs is automatically configured to be Per-Port.

### VLAN over Bond Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>Topology</th>
<th>Slaves</th>
<th>SSM</th>
<th>Port</th>
<th>VLAN</th>
<th>Distribution Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>bond1.100</td>
<td>Internal</td>
<td>eth1-01</td>
<td>1</td>
<td>1</td>
<td>100</td>
<td>User</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eth2-01</td>
<td>2</td>
<td>1</td>
<td>100</td>
<td>User</td>
</tr>
<tr>
<td>bond1.200</td>
<td>External</td>
<td>eth1-01</td>
<td>1</td>
<td>1</td>
<td>200</td>
<td>Network</td>
</tr>
<tr>
<td></td>
<td></td>
<td>eth2-01</td>
<td>2</td>
<td>1</td>
<td>200</td>
<td>Network</td>
</tr>
</tbody>
</table>

Interface bond1.100 is defined on two SSM1 and SSM2 VLANs. Interface bond1.200 is defined on the same SSM1 and SSM2 VLANs. On at least one of the SSMs, some VLANs are Internal and others are External. Therefore, the Distribution Mode is automatically configured to be Per-Port.
**Note**: Not supported in SSM60. In an SSM60 the Distribution Mode of all the VLANs must be the same.

**Bridge interfaces**

If there is a Layer-2 Bridge Interface, the Distribution Mode of all the SSMs is automatically configured to be General.

**SSM60 VLAN Legacy Support**

The SSM60 does not support the new VLAN scheme used by the SSM160. SSM60 users should continue to use the legacy VLAN scheme.

**To activate the legacy VLAN scheme on SSM60:**

1. Run these commands from the Expert shell:
   - `# dbset chassis:1:SSM1:legacy_vlan on`
   - `# dbset chassis:1:SSM2:legacy_vlan on`
   - `# dbset chassis:2:SSM1:legacy_vlan on`
   - `# dbset chassis:2:SSM2:legacy_vlan on`
2. Reboot the SGMs.
   
   You can reboot the SGMs one Chassis at a time to maintain connectivity during this procedure.

**Manual Distribution Configuration (Manual-General)**

In some deployments, you must manually configure a distribution Mode of General. Search for *General Distribution Mode* in this guide. Or, you may want to force the system configure to work in General Mode.

When the Distribution Mode is manually configured (Manual-General Mode), the Distribution Mode of each SSM is General. In this configuration, the topology of the interfaces is irrelevant.

**Note**: We do not recommend that you manually change the Distribution mode of a Virtual System. This can cause performance degradation.

**Setting and Showing the Distribution Configuration**

Use these gclish commands to set and show the distribution configuration.

**Syntax**

```
set distribution configuration {auto-topology|manual-general}
show distribution configuration
```

**Note**

When working with Virtual Systems, you must move to the applicable Virtual System context before you can change the Distribution mode. To do this, run:

```
> set virtual-system <vs_ids>
<vs_ids> = Virtual System context
```
Changing the Distribution Mode to Manual-General

> set distribution configuration manual-general
1_01: configuration update completed successfully
1_02: configuration update completed successfully
1_03: configuration update completed successfully

Showing the distribution

> show distribution configuration
1_01: manual-general
1_02: manual-general
1_03: manual-general

Configuring the Interface Distribution Mode (set distribution interface)

You can use these commands to:

- Set the Distribution Mode for an interface when the system is not working in the General mode.
- Show the currently assigned Distribution Mode and whether that mode is assigned by Auto-Topology or manually configured.

Syntax

set distribution interface <if_name> configuration {user|network|policy}
show distribution interface <if_name> configuration

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>if_name</td>
<td>Interface name as assigned by the operation system</td>
</tr>
<tr>
<td>user</td>
<td>Manually assign the user Distribution Mode</td>
</tr>
<tr>
<td>network</td>
<td>Manually assign the network Distribution Mode</td>
</tr>
<tr>
<td>policy</td>
<td>Use Auto-Topology to assign the automatically assign Distribution Mode</td>
</tr>
</tbody>
</table>

Example

This example shows how to:

1. Manually change the Distribution Mode for interface eth1-01 from policy to network.
2. Change the Distribution Mode on interface eth1-01 from network to policy.
> set distribution interface eth1-01 configuration network
 1_01: configuration update completed successfully

1_02: configuration update completed successfully

1_03: configuration update completed successfully

> set distribution interface eth1-01 configuration policy
 1_01: configuration update completed successfully

1_02: configuration update completed successfully

1_03: configuration update completed successfully

**Showing Distribution Status**

Use this command to show a summary or detailed status report of the Distribution mode.

**Syntax**

```
show distribution status [verbose]
```

**Parameter** | **Description**
---|---
verbose | Shows a detailed report for all SGMs and SSMs

**Example**

```
> show distribution status verbose
```

**Explanation of the output data**

- **distribution mode** - Currently configured Distribution mode.
- **ssm 1 mode** - Distribution Mode assignment for each SSM.
- **ipv6 mode** - Shows if IPv6 is enabled for this system (on/off).
- **spi mode** - Shows if SPI affinity is enabled for this system (on/off).
- **40g mode** - Shows if QSFP ports are working at 40GbE (On) or at 4 x 10GbE (Off).
- **matrix size** - The size of the Distribution matrix. The Distribution matrix is a table containing SGM IDs that are used for traffic assignment.
- **interface** - Shows the Distribution mode assignment for each interface.
Running a Verification Test (show distribution verification)

Use this command to run a verification test of the Distribution Mode configuration. This test compares the SGM and SSM configuration with the actual results. You can see a summary or a detailed (verbose) report of the test results.

Syntax

```
show distribution verification [verbose]
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>verbose</td>
<td>Shows a detailed report for all SGMs and SSMs</td>
</tr>
</tbody>
</table>

Example

**Note:** This example shows only a small sample of the data. The checksums are truncated to fit on the page.

```
> show distribution verification verbose
Test:
chassis 1 blade 1 dxl-general-mode          off           off          Passed
chassis 1 blade 1 dxl-md5sum               5be67561a...  5be675611...  Passed
chassis 1 blade 1 dxl-size                2048          2048         Passed
chassis 1 blade 2 dxl-general-mode         off           off          Passed
chassis 1 blade 2 dxl-md5sum               5be67561a...  5be675611...  Passed
chassis 1 blade 2 dxl-size                2048          2048         Passed
chassis 1 blade 3 dxl-general-mode         off           off          Passed
chassis 1 blade 3 dxl-md5sum               5be67561a...  5be675611...  Passed
chassis 1 blade 3 dxl-size                2048          2048         Passed
chassis 1 ssm 1 ipv6-mode                off           off          Passed
chassis 1 ssm 1 mask ipv4 general destination  0000001f      0000001f     Passed
chassis 1 ssm 1 mask ipv4 general source    0000001f      0000001f     Passed
chassis 1 ssm 1 mask ipv4 user-network destination 000007ff      000007ff     Passed
chassis 1 ssm 1 mask ipv4 user-network source          000007ff      000007ff     Passed

Summary:
verification passed successfully
```

NAT and the Correction Layer on Security Gateway

For optimal system performance, a session from start to finish should be handled by the same SGM. With NAT, packets sent from the client to the server may be distributed to a different SGM than packets from the same session sent from the server to the client. The system Correction Layer then has to forward the packet to the correct SGM.

Correctly configuring Distribution Modes keeps corrections situations to a minimum and optimizes system performance. To achieve optimal distribution between SGMs on the gateway:

- **When not using NAT rules:** Set the General Distribution Mode.
- **When using NAT rules:** Set the hidden network(s) to User Mode, and the destination network(s) to Network Mode.

NAT and the Correction Layer on a VSX Gateway

In a VSX Gateway, the guidelines in the “NAT and the Correction Layer on Security Gateway” section apply to each Virtual System individually. In particular, a session from start to finish should be handled by the same SGM by a given Virtual System. When a Virtual Router or Virtual Switch (“Junction”) connects several Virtual Systems, the same session may be handled by one Virtual System on one SGM, and by another Virtual System on a different SGM.

When a packet reaches a Virtual System from a Junction, the system VSX Stateless Correction Layer rechecks the distribution according to the Warp interface’s Distribution Mode, and may decide to forward the packet to a different SGM.

In addition, on each Virtual System the system Correction Layer, which is stateful, may forward session’s packets, similarly to Security Gateway.

All forwarding operations have a performance impact, so the Distribution Mode configuration should minimize forwarding operations.

To achieve optimal distribution between SGMs on the VSX Gateway:
- **When not using NAT rules on any Virtual System**: Set the General Distribution Mode.

- **When using NAT rules on one or more Virtual Systems**: Set the hidden network(s) to User Mode, and the destination network(s) to Network Mode.

For the remaining Virtual Systems (not using NAT rules), set internal network(s) to User Mode, and the external network(s) to Network Mode.

**Common Scenarios With A Virtual Router**

The following are examples for common scenarios with a Virtual Router. The examples also apply to a Virtual Switch. The examples show the recommended Distribution Mode configuration for optimal performance.

In both examples there are two Virtual Systems (VS1 and VS2), and one Virtual Router (VR). VS1 and VS2 protect internal networks A and B, respectively. VR connects VS1, VS2, and network C, which is an external network. VS1 has NAT rules that hide Network A behind it. VS2 does not use NAT rules.

**Example 1**

In this example most of the traffic is from Networks A and B toward Network C.

Because only VS1 uses NAT rules, we will start configuring the interfaces’ Distribution Mode according to it. VS1 hides Network A. Therefore the Distribution Mode of eth1-01 is User.

Traffic from Network A leaves VS1 on wrp64, so the Distribution Mode of wrp64 is the opposite, Network.

Interface eth2_01 is configured to Network as well, since the VR does not change the packet.
Packets from Network A to network C are distributed by their destination (User).

Packets from Network C to network A are distributed by their source (Network). Since eth2-01 and wrp64 have the same Distribution Mode, the VSX Stateless Correction Layer does not forward them to a different SGM. Therefore, no Forwarding operations are required by the correction Layer.

We now configure the Distribution Mode for VS2, which does not use NAT rules. Because the Distribution Mode of eth2_01 is Network, The Distribution Mode of wrp128 is also set to Network.

Finally, the Distribution Mode of eth1_02 is set to User (the opposite of wrp128). It is easy to see that with this configuration no Forwarding operations is required by the correction Layer for traffic between Network B and Network C.

**Example 2**

In this example most of the traffic is from Network A toward Network B, and from Network B toward Network C.

As in the previous example, because only VS1 uses NAT rules, we start configuring the Distribution Mode of the interfaces according to VS1. VS1 hides Network A. Therefore the Distribution Mode of eth1-01 is User.

Traffic from Network A leaves VS1 on wrp64 so the Distribution Mode of wrp64 is the opposite: Network.

Most of the traffic from network A is toward network B, meaning it will be inspected by VS2 as well. To prevent forwarding by the system VSX Stateless Correction Layer, wrp128 has the same Distribution Mode as eth1-01. That is, the distribution for both is determined by the packet’s destination address, which is not changed by the NAT rules.
To complete the configuration of VS2, set the Distribution Mode of eth1-02 to Network (the opposite of wrp128).

Finally, set the Distribution Mode of eth2-01. Note that wrp64 is configured to Network, and wrp128 is configured to User. Since there is more traffic from Network B to Network C than from Network A to Network C, we configure eth2-01 to User (same as wrp128).

With this configuration, no Forwarding operations are required by the correction Layer for traffic between Network B and Network C, or for traffic between Network A and Network B.

Hybrid System

A 61000/41000 Security System Hybrid System is a deployment that includes SGMs with different numbers of physical CPU cores. In a Hybrid System, the total number of CoreXL and Performance Pack instances that can run on one SGM is equal to the number of physical installed CPU cores. All SGMs must have the same number of CoreXL instances. The number of Performance Pack instances can be different.

Note - While it is possible to mix SGM220 and SGM260 units in the same environment, we do not recommend this configuration.

For example, a Hybrid System can contain these SGMs:

<table>
<thead>
<tr>
<th>SGM</th>
<th>Physical Cores</th>
<th>CoreXL Instances</th>
<th>Performance Pack Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_01</td>
<td>12</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>1_03</td>
<td>20</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>1_04</td>
<td>40</td>
<td>10</td>
<td>20</td>
</tr>
</tbody>
</table>

How this works:

When an SGM boots, the 61000/41000 Security System makes sure that the number of CoreXL instances on the SGM matches the number defined for all other SGMs. Typically, this information comes from the SMO.

If the SGM has too many CoreXL instances, the system automatically reassigns these instances as Performance Pack instances. If the SGM has insufficient CPU cores, the SGM stays in the Down state. You must manually change the number of CoreXL instances and then reboot the SGM.

To see the number of CoreXL instances defined for ALL SGMs:

Run:

> asg_cores_stats

To manually change the number of CoreXL instances for ALL SGMs:

Run:

> cpconfig corexl instances <#_of_instances>

<#_of_instances> is the number of CoreXL instances for all SGMs.

Important Notes:

- There will always be at least one core configured as a CoreXL instance and one as a Performance Pack instance.
- The maximum number of Performance Pack instances on an SGM is the lesser of Physical cores -1 or 16.
- The maximum number of CoreXL instances on an SGM is Physical cores -1.
- If manual Performance Pack core configuration for one SGM causes an invalid configuration on a different SGM, it automatically goes back to the default Performance Pack configuration.
- It is possible to have overlapping CoreXL and Performance Pack instances, where the number of instances is greater than the number of physical cores. We do not recommend this configuration.
GARP Chunk Mechanism

Description:
When Proxy ARP is enabled, the Firewall responds to ARP requests for hosts other than itself. When Chassis failover occurs, the new Active Chassis sends GARP to update the network ARP tables.

To prevent network congestion during Chassis failover, GARP requests/responses are sent in user defined groups called "chunks". Each chunk contains a predefined number of GARP messages based on these parameters:

- The number of GARP messages in each chunk
- HTU (High Availability Time Unit) - Time interval, after which a chunk is sent.

The chunk mechanism is iterating on the proxy ARP IPs, and each time sends GARP only for some of them until it completes the entire list.

In each HA Time Unit (HTU=0.1s) - a chunk of the GARP list is sent.
Whenever the iteration is finished send all the list, it waits N HTU and sends the list again.

Configuration:
In each HTU (=0.1 second) - a chunk of the GARP list is sent.
For example, if we want that 10 GARPs will be sent in each second
fwha_refresh_arps_chunk should be set to 1.
(command: # fw ctl set int fwha_refresh_arps_chunk 1)

For 50 GARPs/seconds,
fwha_refresh_arps_chunk should be set to 5.
(command: # fw ctl set int fwha_refresh_arps_chunk 5)

Whenever the iteration is finished sending GARPs for the entire list, it waits N HTU and re-sends the GARPs again. The time between the iterations can be configured with:
fwha_periodic_send_garps_interval1 = (1 HTU) /* should not be changed, send immediately after failover */
fwha_periodic_send_garps_interval2 = (10 HTU) /* 01 seconds */
fwha_periodic_send_garps_interval3 = (20 HTU) /* 02 seconds */
fwha_periodic_send_garps_interval4 = (50 HTU) /* 05 seconds */
fwha_periodic_send_garps_interval5 = (100 HTU) /* 10 seconds */

In the above (default) configuration, after finishing iterate the list, wait 1 seconds and start send again
wait 2 seconds and start send again.
wait 5 seconds and start send again.
wait 10 seconds and start send again.

To change interval:
fw ctl set int fwha_periodic_send_garps_interval<1-5> 1
To apply intervals:
fw ctl set int fwha_periodic_send_garps_apply_intervals 1

Verification:
In order to initiate manual garp sending:
On the Chassis monitor blade, run:
fw ctl set int test_arp_refresh 1
This will cause garp sending (same as was failover)

Debug:
fw ctl zdebug -m cluster + ch_conf | grep fw_refresh_arp_proxy_on_failover
Hardware Components

Chassis Management Module (CMM) CLI

The Chassis Management Module (CMM) monitors and controls hardware modules in the Chassis. Communication with a CMM occurs by SNMP requests from a dedicated SGM. If a hardware sensor reports a problem, the CMM automatically takes action or sends a report. CMMs also have a Command Line Interface.

There are two procedures to connect to a CMM CLI:

1. Connect to the serial port on the front panel of the CMM
2. In your terminal emulation program, set the baud rate to 9600
3. Enter admin for the user name and password
4. Open a telnet or SSH session from one of the SGMs
5. First make sure that you have connectivity to the CMMs by pinging both addresses:
   6. 198.51.100.33 (routed via SSM1)
   7. 198.51.100.233 (routed from SSM2)
8. Telnet or SSH from the SGM to the CMM
9. Enter admin for the user name and password

When connected:

1. Modify the Chassis configuration, including the Chassis ID (1 or 2) by editing 
   /etc/shmm.cfg
2. Useful commands: (Should be run after entering CLI shell, via clia command)
3. Show the full list of commands:
   CLI> help
4. Show alerts
   CLI> alarm
5. Reset alerts
   CLI> alarm 0
6. Show power consumption information
   CLI> shelf pd
7. Retrieve event logs
   CLI> sel
8. Reboot the CMM
   CLI> reboot
   Note: reboot initiates a failover to the standby CMM
9. Make sure that a board is recognized in its slot
   CLI> board
10. Reset the specified board
    CLI> boardreset <slot number>
11. Display FRU information
    CLI> fru [fru_id]

This table shows the mapping between SGM ID in CMM and SGM ID in Chassis.
Example: SGM 5 shows as 8a on CMM

61000 slot information

<table>
<thead>
<tr>
<th>Physical slot</th>
<th>IPMB</th>
<th>SGM/SSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9a</td>
<td>SGM1</td>
</tr>
<tr>
<td>2</td>
<td>96</td>
<td>SGM2</td>
</tr>
<tr>
<td>3</td>
<td>92</td>
<td>SGM3</td>
</tr>
<tr>
<td>Physical slot</td>
<td>IPMB</td>
<td>SGM/SSM</td>
</tr>
<tr>
<td>--------------</td>
<td>------</td>
<td>---------</td>
</tr>
<tr>
<td>4</td>
<td>8e</td>
<td>SGM4</td>
</tr>
<tr>
<td>5</td>
<td>8a</td>
<td>SGM5</td>
</tr>
<tr>
<td>6</td>
<td>86</td>
<td>SGM6</td>
</tr>
<tr>
<td>7</td>
<td>82</td>
<td>SSM1</td>
</tr>
<tr>
<td>8</td>
<td>84</td>
<td>SSM2</td>
</tr>
<tr>
<td>9</td>
<td>88</td>
<td>SGM7</td>
</tr>
<tr>
<td>10</td>
<td>8c</td>
<td>SGM8</td>
</tr>
<tr>
<td>11</td>
<td>90</td>
<td>SGM9</td>
</tr>
<tr>
<td>12</td>
<td>94</td>
<td>SGM10</td>
</tr>
<tr>
<td>13</td>
<td>98</td>
<td>SGM11</td>
</tr>
<tr>
<td>14</td>
<td>9c</td>
<td>SGM12</td>
</tr>
</tbody>
</table>

### 41000 Security System slot information

<table>
<thead>
<tr>
<th>Physical slot</th>
<th>IPMB</th>
<th>SGM/SSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upper most Slot</td>
<td>8C</td>
<td>SGM1</td>
</tr>
<tr>
<td></td>
<td>8A</td>
<td>SGM2</td>
</tr>
<tr>
<td></td>
<td>88</td>
<td>SGM3</td>
</tr>
<tr>
<td></td>
<td>86</td>
<td>SGM4</td>
</tr>
<tr>
<td></td>
<td>84</td>
<td>SSM2</td>
</tr>
<tr>
<td>Lowest Slot</td>
<td>82</td>
<td>SSM1</td>
</tr>
</tbody>
</table>

### CMM debug commands – How to activate the log function:

1. Log into the active CMM
2. Run
   `/etc/summary`
   can take several minutes
3. Run
   `cat /tmp/debug.log`
   prints debug log with all basic information
4. Run
   `i2c_test`
   tests the internal CMM I2C and prints all devices connected on the I2C
5. Run
   `cat /etc/shmm.cfg`
   prints ShMM's custom configuration
6. Run
   `clia fruinfo 20 x`
   on the 61000: Where x is between 0 to 16
   on the 41000 Security System: Where x is between 0 to 9
7. Run
   `clia fruinfo y 0`
   16 times where y is 10,12,82,84,86,88,8a,8c,8e,90,92,94,96,98,9a,9c
8. Close your terminal program. /tmp/debug.log file will hold the debug information.

Security Switch Module (SSM) CLI

**Description**
The Security Switch Module (SSM):
- Distributes network traffic to the Security Gateway Modules (SGMs)
- Forwards traffic from the SGMs to the network
- Shares the load amongst the SGMs
- Communication between the SSMs and SGMs occurs automatically via SNMP requests, but you can also connect directly to the SSM and run commands.
- There are two ways to connect to the SSM CLI:
- Connect to a serial port on the front panel of the SSM.
- Open a telnet session from one of the SGMs

**SSM60 CLI**

1. Connect to a serial port on the front panel of the SSM.
The SSM60 has two serial ports, one for the fabric switch (data ports) and one for the base switch (management ports).

2. In your terminal emulation program, set the baud rate to 9600.
3. Enter admin for the password.
4. Enter enable. This gives read and write permissions to the system. Not entering enable results in read-only permissions.
5. Enter ? for a list of available commands and usage.

   ![SSM60 CLI Diagram]

   **Note** - Load balancing commands are run on the fabric switch only.

6. Open a telnet session from one of the SGMs.
- First make sure that you have connectivity to the SSMs by pinging these addresses:

<table>
<thead>
<tr>
<th>SSM</th>
<th>Switch</th>
<th>IP address</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Base</td>
<td>198.51.100.31</td>
</tr>
<tr>
<td></td>
<td>Fabric</td>
<td>198.51.100.32</td>
</tr>
<tr>
<td>2</td>
<td>Base</td>
<td>198.51.100.231</td>
</tr>
<tr>
<td></td>
<td>Fabric</td>
<td>198.51.100.232</td>
</tr>
</tbody>
</table>

1. Telnet from the SGM to the SSM
2. Enter admin for the password.
3. Enter enable. This gives read and write permissions to the system. Not entering enable results in read-only permissions.
4. Enter ? for a list of available commands and usage.
When connected, use these useful troubleshooting commands:

<table>
<thead>
<tr>
<th>To</th>
<th>Run:</th>
</tr>
</thead>
<tbody>
<tr>
<td>View the current configuration</td>
<td># show running-config</td>
</tr>
<tr>
<td>View current ports status</td>
<td># show interface</td>
</tr>
<tr>
<td>View interface statistics</td>
<td># show interface &lt;interface id&gt; statistics [extended]</td>
</tr>
<tr>
<td>View SSM logs</td>
<td># show log buffer</td>
</tr>
</tbody>
</table>
| Modify the group of SGMs amongst which the load is distributed | # configure terminal
  (config)# load-balance mtx-bucket [SGM ID, SGM ID,]
  (config)# load-balance apply
  Note: the command will not work if you have an odd number of SGMs in the group. For example, do not run:
  #load-balance mtx-bucket 1,2,3
  Run:
  #load-balance mtx-bucket 1,2,3,1,2,3 |

### SSM160 CLI

**Description**

The SSM (Security Switch Module) is the networking module of the gateway. The SSM transmits traffic to and from the SGM and performs the load distribution among the SGMs.

The SSM includes two modules:

- Fabric switch - includes the Data ports
- Base switch - includes the Management ports.

Most of the communication with the SSM is done automatically by SNMP requests from the SGM but on some events connecting directly to the SSM can be useful.

**Configuration**

Connection to the SSM CLI can be established in two ways:

- The administrator can connect with a serial console to the "CLI" port on the SSM front panel (baud rate 9600).
- From one of the SGMs use SSH to connect to the SSM.
  The SSM IPs can be retrieved from CLISH/GCLISH:
    - show Chassis id <1|2|all> module SSM<1|2> ip
    - The password for the SSM is admin.
    - Once connected to the SSM CLI you can do the following:

  **View the current configuration**
  # show running-config <feature name>
  Since the entire configuration is very long it is recommended to specify the feature that you are interested in its configuration. For example, show running-config load-balance to see the Load Balance configuration. You can press tab to see a complete list of the features.

  **View current ports status**
  # show port

  **View detailed port information (speed, administrative state, link state, etc.)**
  # show port <port id>

  **View interface statistics**
# show port <port_id> statistics

T-HUB4#show port 1/3/1 statistics

Port Statistics

<table>
<thead>
<tr>
<th></th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unicast Packets</td>
<td>5003</td>
<td>7106</td>
</tr>
<tr>
<td>Multicast Packets</td>
<td>568409</td>
<td>1880</td>
</tr>
<tr>
<td>Broadcast Packets</td>
<td>122151</td>
<td>1972</td>
</tr>
<tr>
<td>Flow Control</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Discards</td>
<td>16</td>
<td>0</td>
</tr>
<tr>
<td>Errors</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Total: 695563 10958

Ethernet Statistics in Packets

<table>
<thead>
<tr>
<th></th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>RX CRC Errors</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>RX Undersize</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Packets</td>
<td>71085491</td>
<td>706521</td>
</tr>
<tr>
<td>Packets of 64 Octets</td>
<td>2290</td>
<td>0</td>
</tr>
<tr>
<td>Packets of 65 to 127 Octets</td>
<td>689951</td>
<td>4122</td>
</tr>
<tr>
<td>Packets of 128 to 255 Octets</td>
<td>6009</td>
<td>0</td>
</tr>
<tr>
<td>Packets of 256 to 511 Octets</td>
<td>258</td>
<td>0</td>
</tr>
<tr>
<td>Packets of 512 to 1023 Octets</td>
<td>994</td>
<td>0</td>
</tr>
<tr>
<td>Packets of 1024 to 1518 Octets</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Packets of 1519 or more Octets</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Total: 695563 10958

Rates in Bytes per Second

<table>
<thead>
<tr>
<th></th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate for last 10 sec</td>
<td>1477</td>
<td>25</td>
</tr>
<tr>
<td>Rate for last 60 sec</td>
<td>1435</td>
<td>50</td>
</tr>
</tbody>
</table>

Pay special intention to "Discards" and "Errors" fields which might indicate on a problem if they are constantly increasing.

View SSM logs
#unhide private (default password is "private")
#show private shell
# tail /var/log/messages
Modify load distribution SGM group
# configure terminal
(config)# load-balance mtx-bucket 1 buckets [<SGM ID>:<SGM ID>:<SGM ID>:<SGM ID>…]
(config)# commit
(config)# exit
#load-balance apply

Note
You need to provide a full list of the SGMs as the SGM list parameter to the load-balance mtx-bucket command.
Otherwise, traffic might be dropped on the SSM.

Switch between Ports modes for 40G ports (4X10G or 1X40G):
#unhide private (default password is "private")
#show private shell
For switching to 1X40G mode:
# /batm/binux/bin/ub_util -s ahub4_40G yes
For switching to 4X10G mode:
# /batm/binux/bin/ub_util -s ahub4_40G
# exit
#config terminal
(config)#system reload
Note This procedure requires to reload the SSM. It is recommended to do it one SSM at a time.

View the current version information
#show version
Logout from current session
#logout

Changing SSM160 admin password
Login via SSH/Serial console to an SGM which resides in the same Chassis you wish to change SSMs password
From Expert shell Login to either of the SSMs in the Chassis using:
ssh admin@ssm<ssm_id>
Enter admin password when prompted.
In SSMs shell run the following commands:
#conf t
#system security user admin
#password
Enter new password
#commit
#end
#logout

Notes
This procedure should be done separately on each SSM in the system.
This procedure does not cause any traffic interruption

Example
# ssh ssm2
admin@ssm2's password:
BATM T-HUB4
admin connected from 198.51.100.215 using ssh on T-HUB4
T-HUB4#conf t
Entering configuration mode terminal
T-HUB4(config)#system security user admin
T-HUB4(config-user-admin)#password
(<MD5 digest string>): *****
T-HUB4(config-user-admin)#commit
Commit complete.
T-HUB4(config-user-admin)#end
T-HUB4#log
Connection to ssm2 closed.

Each port ID on the SGM maps to a port on the SSM. Below table maps SSM port ID to SGM port ID.
Note that this table relates to SSM1. For SSM2 replace eth1-X with eth2-X:

<table>
<thead>
<tr>
<th>SGM</th>
<th>SSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth1-01</td>
<td>1/3/1</td>
</tr>
<tr>
<td>eth1-02</td>
<td>1/3/2</td>
</tr>
<tr>
<td>eth1-03</td>
<td>1/3/3</td>
</tr>
<tr>
<td>eth1-04</td>
<td>1/3/4</td>
</tr>
<tr>
<td>eth1-05</td>
<td>1/3/5</td>
</tr>
<tr>
<td>eth1-06</td>
<td>1/3/6</td>
</tr>
<tr>
<td>eth1-07</td>
<td>1/3/7</td>
</tr>
<tr>
<td>eth1-Sync</td>
<td>1/3/8</td>
</tr>
<tr>
<td>eth1-09</td>
<td>1/1/1</td>
</tr>
<tr>
<td>eth1-10</td>
<td>1/1/2</td>
</tr>
<tr>
<td>eth1-11</td>
<td>1/1/3</td>
</tr>
<tr>
<td>eth1-12</td>
<td>1/1/4</td>
</tr>
<tr>
<td>eth1-13</td>
<td>1/1/5</td>
</tr>
<tr>
<td>eth1-14</td>
<td>1/1/6</td>
</tr>
<tr>
<td>eth1-15</td>
<td>1/1/7</td>
</tr>
<tr>
<td>eth1-16</td>
<td>1/1/8</td>
</tr>
<tr>
<td>eth1-Mgmt1</td>
<td>1/5/1</td>
</tr>
<tr>
<td>eth1-Mgmt2</td>
<td>1/5/2</td>
</tr>
<tr>
<td>eth1-Mgmt3</td>
<td>1/5/3</td>
</tr>
<tr>
<td>eth1-Mgmt4</td>
<td>1/5/4</td>
</tr>
</tbody>
</table>

Verification
To verify that you have connectivity to the SSMs from the SGMs ping all the SSM modules IPs. You can also verify that SNMP connectivity is available by running from SGM shell:
`asg_chassis_ctrl get_ssm_firmware all`
Security Gateway Modules

The Security Gateway Modules (SGMs) in the Chassis work together as a single, high performance Security Gateway or VSX Gateway. Adding a Security Gateway Module scales the performance of the system. A Security Gateway Module can be added and removed without losing connections. If an SGM is removed or fails, traffic is distributed to the other active SGMs.

These SGM versions are available:
- SGM220 (Not supported in a 4-SSM configuration or the 41000 Security System.)
- SGM220T (for NEBS only - Not supported for the 41000 Security System)
- SGM260 (Supports 4-SSM configuration)

The SGM260 has more powerful CPUs and uses a more advanced technology. It also has a different front panel layout and different LEDs.

**Identifying SGMs in the Chassis (asg_detection)**

**SGM260 LEDs**

<table>
<thead>
<tr>
<th>Item</th>
<th>LED</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Out of service</td>
<td>Red</td>
<td>SGM out of service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SGM hardware is normal</td>
</tr>
<tr>
<td>6</td>
<td>Health</td>
<td>Green (Normal)</td>
<td>SGM core operating system is active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green blinking</td>
<td>SGM core operating system is partially active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>SGM operating system is in standby mode</td>
</tr>
<tr>
<td>7</td>
<td>Hot-swap</td>
<td>Blue</td>
<td>SGM can be safely removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blue blinking</td>
<td>SGM is going to standby mode. Do not remove</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SGM is active. Do not remove</td>
</tr>
<tr>
<td>CTRL 1</td>
<td>Link 1</td>
<td>Yellow</td>
<td>Link enabled</td>
</tr>
<tr>
<td>CTRL 2</td>
<td>Link 2</td>
<td>Yellow blinking</td>
<td>Link is active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>Link is disabled</td>
</tr>
<tr>
<td>CTRL S</td>
<td>SSM1 and</td>
<td>Yellow</td>
<td>Link enabled</td>
</tr>
<tr>
<td>SPEE D 1</td>
<td>SSM2</td>
<td>Yellow blinking</td>
<td>Link is active</td>
</tr>
<tr>
<td></td>
<td>management</td>
<td>Off</td>
<td>Link is disabled</td>
</tr>
<tr>
<td></td>
<td>ports</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTRL S</td>
<td>SSM1 and</td>
<td>Yellow</td>
<td>10 Gbps</td>
</tr>
<tr>
<td>SPEE D 2</td>
<td>SSM2</td>
<td>Green</td>
<td>1 Gbps</td>
</tr>
<tr>
<td></td>
<td>management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CTRL SPEED 2</td>
<td>t ports</td>
<td>Off</td>
<td>100 Mbps</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
<td>-----</td>
<td>----------</td>
</tr>
</tbody>
</table>

| Traffic 1 2 3 4 | On | Data and sync traffic in SSM1, SSM2, SS3, SSM4 |

| L2 | Off | Not used |
| L1 | Red. Lower Right | Installation started |
| | Red blink, in sequence | Installation in progress |
| | Red. All | Installation failure |
| | Yellow. Left | Installation completed |
| | Green. Right | SGM is being configured. (Using First Time Configuration Wizard or adding a new SGM into a Chassis) |
| | Off | SGM is configured and ready |
### SGM220 LEDs

<table>
<thead>
<tr>
<th>Item</th>
<th>LED</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>Out of service</td>
<td>Red SGM out of service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SGM hardware is normal</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>Health</td>
<td>Green (Normal) SGM core operating system is active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green blinking</td>
<td>SGM core operating system is partially active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>SGM operating system is in standby mode</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Hot-swap</td>
<td>Blue SGM can be safely removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blue blinking</td>
<td>SGM is going to standby mode. Do not remove</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SGM is active. Do not remove</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Link</td>
<td>Yellow Link enabled</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yellow blinking</td>
<td>Link is active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>Link is disabled</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>Data port speed</td>
<td>Yellow 10 Gbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green</td>
<td>1 Gbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>100 Mbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Management port speed</td>
<td>Yellow 1 Gbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Green</td>
<td>100 Mbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>10 Mbps</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>L LEDs 2 and 4 - Green</td>
<td>SGM is being configured. (Using First Time Wizard or adding a new SGM into a Chassis)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>All LEDs - Off</td>
<td>SGM is configured and ready</td>
</tr>
</tbody>
</table>
### Security Switch Module LEDs

<table>
<thead>
<tr>
<th>Item</th>
<th>LED</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Out of service</td>
<td>Red</td>
<td>SSM out of service</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SSM hardware is normal</td>
</tr>
<tr>
<td>2</td>
<td>Power</td>
<td>On (Normal)</td>
<td>Power on</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>Power off</td>
</tr>
<tr>
<td>3</td>
<td>Hot-swap</td>
<td>Blue</td>
<td>SSM can be safely removed</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Blue blinking</td>
<td>SSM is going to standby mode. Do not remove</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off (Normal)</td>
<td>SSM is active. Do not remove</td>
</tr>
<tr>
<td>4</td>
<td>SYN ACT</td>
<td>On (Normal)</td>
<td>Normal operation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>N/A</td>
</tr>
<tr>
<td>5</td>
<td>Link</td>
<td>On</td>
<td>Link enabled</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Yellow blinking</td>
<td>Link is active</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Off</td>
<td>Link is disabled</td>
</tr>
</tbody>
</table>
Chapter 6

Software Blades Support

In This Section:

Software Blades Updates ........................................................................................................... 226
IPS Bypass under Load .................................................................................................................. 226
IPS Cluster Failover Management .................................................................................................. 227

Software Blades Updates

61000/41000 Security System periodically updates Anti-Virus, Anti-Malware and URL Filtering databases, same as other Check Point products.

In order to manually update Anti-Virus, Anti-Malware and URL Filtering databases, use g_avsu_update command. This command is available from Expert shell only.

Upon execution, the command will update the database of the relevant SGMs

Syntax:

> g_avsu_update -b <sgm_ids> <urlf/av/all>

Note:

Update configuration (proxy, username, etc.) should be set in SmartDashboard before issuing this command. Policy should be installed afterwards.

Manual updates of Anti-Virus, Anti-Malware and URL Filtering from Management are not supported.

IPS Bypass under Load

Bypass under Load allows the administrator to define a gateway resource load level at which IPS inspection will temporarily be suspended until the gateway's resources return to acceptable levels.

IPS inspection can make a difference in connectivity and performance. Usually, the time it takes to inspect packets is not noticeable; however, under heavy loads it may be a critical issue.

You have the option to temporarily stop IPS inspection on a gateway if it comes under heavy load.

For more about this feature, see the R76 IPS Administration Guide (http://supportcontent.checkpoint.com/documentation_download?ID=22915).
IPS Cluster Failover Management

**Description**
You can configure how IPS is managed during a cluster failover (when one member of a cluster takes over for another member to provide High Availability).

**To configure failover behavior for a cluster:**
Run from expert shell: `asg_ips_failover_behavior [connectivity|security]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connectivity</td>
<td>change the behavior upon cluster failover to prefer connectivity: Close connections for which IPS inspection cannot be guaranteed</td>
</tr>
<tr>
<td>security</td>
<td>change the behavior upon cluster failover to prefer security: Keep connections alive even if IPS inspections cannot be guaranteed</td>
</tr>
</tbody>
</table>
Chapter 7

Troubleshooting

In This Section:

Collecting System Information (asg_info) .......................................................... 228  
Verifiers ................................................................................................................. 232  
Resetting SiC (g_cpconfig sic init) ................................................................. 239  
Troubleshooting Hardware .................................................................................. 240  
Debug files ......................................................................................................... 245

Collecting System Information (asg_info)

Description
Use this command to collect system information. This command runs many commands that generate data files and command line output. The main categories of collected information include:

- Log files
- Configuration files
- System status
- Indication for possible errors

The information is collected from all SGMs and sent to a compressed folder at:
/var/log/asg_report.<timestamp>

Commands
The commands that are being run by asg_info are divided into three groups.

- System commands run on the SMO
- Commands run on only one SGM for each Chassis
- Commands that run on all SGMs
- VSX Mode only: Commands are divided into two groups:
  - Per VS: Run on all vs range given by user
  - Global: Run only on VS0 context

Files
asg_info collects certain files from all SGMs. Files are sent to specific folders. For example, all core dump files are located under the folder core_dump. SGM ID is added to file names, in order to indicate where data was collected from.

For example:

File name format for files that are part of core_dump folder:

- global_1_02_coredump.gz
- global_2_03_coredump.gz

The first one was collected from SGM 2 in Chassis 1, and the second was collected from SGM 3 in Chassis 2. No other files exist in core_dump folder, which means that all the other SGM didn’t have any information to send.
General

Information about core dumps created by the system can be found in core_dump_global.txt.

Syntax

```
```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[SGMs list]</td>
<td>List of SGMs, default: all up SGMs</td>
</tr>
<tr>
<td></td>
<td>Example: asg_info -a will attempt to collect information from all SGMs,</td>
</tr>
<tr>
<td></td>
<td>including down SGMs</td>
</tr>
<tr>
<td>-vs [&lt;range&gt;</td>
<td>all]</td>
</tr>
<tr>
<td>-f</td>
<td>Collect and zip information files</td>
</tr>
<tr>
<td>-c</td>
<td>Collect and zip cores</td>
</tr>
<tr>
<td>-i</td>
<td>Collect and zip cpinfo</td>
</tr>
<tr>
<td>-all</td>
<td>Collect and zip all above files - this operation may take several minutes</td>
</tr>
<tr>
<td>-h</td>
<td>Display usage message</td>
</tr>
<tr>
<td>-v</td>
<td>Display verbose output</td>
</tr>
</tbody>
</table>

**Example: asg_info -c**

```
asg_info -c
Collecting asg_info data to file
Starting processes in background : 100%
Collecting processes output: 100%
Collecting files from remote sgms : 100%
Generating /var/log/asg_report.Brussels-vsx-84_2012.11.29_18.01.59.tar.gz...
```

Notes: This option collects relatively light-weight information. It should finish in a few minutes.

**Example 2: asg_info -all**

```
asg_info -all -v
Collecting asg_info data to file
Starting processes in background : 100%
Collecting CP info... [ OK ]
Collecting ore dump... [ OK ]
Collecting ASG System Verbose Status... [ OK ]
Collecting ASG SGM Proces and State... [ OK ]
Collecting VSX stat... [ OK ]
Collecting ASG diag print... [ OK ]
Collecting Policy Verification... [ OK ]
Collecting AMW Policy Verification... [ OK ]
Collecting backup file... [ OK ]
Collecting ASG Tasks Status... [ OK ]
Collecting ASG var logs... [ OK ]
Collecting Interface Information + Performance... [ OK ]
Collecting ASG HW Monitor... [ OK ]
Collecting SGMs serial numbers... [ OK ]
Collecting Hardware serial numbers... [ OK ]
```
Collecting Versions Manager... [ OK ]
Collecting DXL Statistics... [ OK ]
Collecting DXL distribution matrix... [ OK ]
Collecting Verify SSMs and DXL [ OK ]
Distribution Signatures...
Collecting SNMl information... [ OK ]
Collecting ASG Ifconfig Analyze Verbose Mode...
Collecting ASG GRE Stat... [ OK ]
Collecting ASG GRE Verify... [ OK ]
Collecting SGMs Cores Stats... [ OK ]
Collecting Pingable Hosts Status... [ OK ]
Collecting Topology Interfaces...
Collecting Chassis Ports Link States... [ OK ]
Collecting ConnectControl Table: Check Alive...
Collecting ConnectControl Table: Logical Requests... [ OK ]
Collecting ConnectControl Table: Logical Servers...
Collecting ConnectControl Table: Logical Servers List...
Collecting ConnectControl Table: Cache... [ OK ]
Collecting Proxy Arp Entries in FW... [ OK ]
Collecting Fwaccel stat information... [ OK ]
Collecting Fwaccel stats information... [ OK ]
Collecting Fwaccel stats f2f information...
Collecting Fwaccel stats drop information...
Collecting Fwaccel stats multicast information...
Collecting UIFC Status... [ OK ]
Collecting System Audited Operations Log...
Collecting System Ports Log... [ OK ]
Collecting Audit Log... [ OK ]
Collecting Multi-Queue For Vlan... [ OK ]
Collecting ASG cmd logs... [ OK ]
Collecting Sel info... [ OK ]
Collecting CMM(s) Status... [ OK ]
Collecting CMM: Attached SGMs... [ OK ]
Collecting CMM: Chassis Fans... [ OK ]
Collecting CMM: Chassis Power Supply Units...
Collecting CMM: CPUs temperatures... [ OK ]
Collecting SSM 1: LB Mode... [ OK ]
Collecting SSM 2: LB Mode... [ OK ]
Collecting SSM 1: distribution matrix... [ OK ]
Collecting SSM 2: distribution matrix... [ OK ]
Collecting SSM 1: overall throughput... [ OK ]
Collecting SSM 2: overall throughput... [ OK ]
Collecting SSM 1: ports status... [ OK ]
Collecting SSM 2: ports status... [ OK ]
Collecting SSM 1: SGMs MACs... [ OK ]
Collecting SSM 2: SGMs MACs... [ OK ]
Collecting Chassis PSU type... [ OK ]
Collecting Ccutil Logs... [ OK ]
Collecting Dist_mode Logs... [ OK ]
Collecting ASG If Error... [ OK ]
Collecting ASG If... [ OK ]
Collecting Outputs to log messages... [ OK ]
Collecting Interfaces Configuration... [ OK ]
Collecting ASG Performance VSX Global... [ OK ]
Collecting ASG Performance... [ OK ]
Collecting ASG Path Distribution Table
VSX Global...
Collecting ASG Path Distribution Table... [ OK ]
Collecting ASG Performance VSX Global... [ OK ]
Collecting ASG Performance... [ OK ]
Collecting ASG Path Distribution Table IPv6...
Collecting ASG Performance IPv6... [ OK ]
Collecting ASG Performance IPv6... [ OK ]
Collecting ASG Connections... [ OK ]
Collecting Correction Layer Statistics [ OK ]
Per Service...
Collecting Correction Layer Statistics [ OK ]
Per Service (verbose)...
Collecting Correction Layer Statistics... [ OK ]
Collecting SMO Statistics & Logs... [ OK ]
Collecting ARP Forwarding Statistics... [ OK ]
Collecting VPN Forwarding Statistics... [ OK ]
Collecting Last Iterator Statistics... [ OK ]
Collecting Processes Affinity... [ OK ]
Collecting Interfaces Affinity... [ OK ]
Collecting Interfaces Affinity [ OK ]
Interrupts...
Collecting Time and Date... [ OK ]
Collecting LV Info... [ OK ]
Collecting Disk Info... [ OK ]
Collecting Blade CPUs... [ OK ]
Collecting Fwaccel stat information... [ OK ]
Collecting Fwaccel stats information... [ OK ]
Collecting Sync bond info... [ OK ]
Collecting CPU Info... [ OK ]
Collecting Mac Magic... [ OK ]
Collecting SUL Status... [ OK ]
Collecting CPU threshold... [ OK ]
Collecting SUL Number of Samples... [ OK ]
Collecting Long Timeout... [ OK ]
Collecting Short Timeout... [ OK ]
Collecting Start Timeout... [ OK ]
Collecting Configuration Database... [ OK ]
Collecting CP Scheduler... [ OK ]
Collecting Licences Log... [ OK ]
Collecting PNOTE status... [ OK ]
Collecting Top Output... [ OK ]
Collecting Core dump files... [ OK ]
Collecting Core crash info... [ OK ]
Collecting FW statistics... [ OK ]
Collecting Uptime... [ OK ]
Collecting inconsistent_routes files from SGMs... [ No Files Found ]
Collecting local.arp files from SGMs... [ No Files Found ]
Collecting policy_backup files from SGMs... [ Copied: 4 Files ]
Collecting cpd_elg files from SGMs... [ Copied: 8 Files ]
Collecting smd_smo files from SGMs... [ No Files Found ]
Collecting mbs files from SGMs... [ Copied: 7 Files ]
Collecting start_mbs files from SGMs... [ Copied: 7 Files ]
Collecting chassis_conf files from SGMs... [ Copied: 7 Files ]
Collecting send_alert files from SGMs... [ Copied: 77 Files ]
Collecting anaconda files from SGMs... [ No Files Found ]
Collecting fwd_elg files from SGMs... [ Copied: 7 Files ]
Collecting fwk_elg files from SGMs... [ Copied: 7 Files ]
Collecting bond_init_log files from SGMs... [ No Files Found ]
Collecting routed_conf files from SGMs... [ Copied: 35 Files ]
Collecting routed_log files from SGMs... [ No Files Found ]
Collecting cpha_policy files from SGMs... [ Copied: 7 Files ]
Collecting blade_conf files from SGMs... [ Copied: 97 Files ]
Collecting reboot_log files from SGMs... [ Copied: 7 Files ]
Collecting alert_conf files from SGMs... [ Copied: 7 Files ]
Collecting fw_kern_conf files from SGMs... [ Copied: 7 Files ]
Collecting simkern_conf files from SGMs... [ Copied: 7 Files ]
Collecting vsaffinity_exception files from SGMs... [ Copied: 7 Files ]
Collecting vsx_temp_info files from SGMs... [ Copied: 61 Files ]
Collecting vsx_local_info files from SGMs... [ Copied: 56 Files ]
Collecting cp_info files from SGMs... [ Copied: 5 Files ]
Collecting core_dump files from SGMs... [ Copied: 7 Files ]
Collecting asg_peaks_history files from SGMs... [ Copied: 1012 Files ]
Collecting asg_peaks_history files from SGMs... [ Copied: 1067 Files ]
Generating /var/log/asg_report.Brussels-vsx-84_2012.11.29_18.02.19.tar.gz...
Operation finished successfully
File: asg_report.Brussels-vsx-84_2012.11.29_18.02.19.tar.gz is located at: /var/log

Notes: This command collects all available data. Its run time is relatively high and may exceed 10 minutes.
Example 3: `asg_info -c`

Notes: This command collects core dump from the SGM if available.

Example 4: `asg_info --vs all --f`

Notes: This option handles the collection of relatively light-weight information from all Virtual Systems. It should finish in a few minutes.

Example 5: `asg_info --vs 1-2,5,7-8 --f`

Notes: This option handles the collection of relatively light-weight information from Virtual Systems 1, 2, 5, 7, 8. It should finish within few minutes.

Verifiers

**MAC Verification (mac_verifier)**

Each MAC address contains information about the Chassis ID, SGM ID and interfaces. Use this command to make sure that the virtual MACS on physical and bond interfaces are the same for all SGMs on each Chassis. Run this command in the expert mode.

**Syntax**

```bash
mac_verifier [-l -v]
mac_verifier -h
```

**Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-l</td>
<td>Shows MAC address consistency on the active Chassis</td>
</tr>
<tr>
<td>-v</td>
<td>Shows information for each interface MAC</td>
</tr>
<tr>
<td>-h</td>
<td>Help screen</td>
</tr>
</tbody>
</table>

**Example**

```bash
# mac_verifier
Starting mac address verification on local chassis... (Chassis 1)
No inconsistency found on local chassis

Starting mac address verification on remote chassis... (Chassis 2)
MAC address inconsistency found on interface eth2-11
```

**L2 Bridge Verifier (asg_br_verifier)**

**Description**

The `asg_br_verifier` is a utility which check if there are bridge configuration problems.

**Syntax**

```bash
asg_br_verifier
asg_br_verifier -v
```

**Example**
asg_br_verifier

Checking for distribution mode conflicts:
Distribution mode is General
Status: OK

Number of entries in fdb_shadow table:
-** 2 blades: 2_01 2_02 2_03 -**
  15
Status: OK

Output of asg_br_verifier –v when there is a misconfiguration:
Port Connectivity Verification (asg_pingable_hosts)

Description
The Port Connectivity Verification feature makes sure that 61000/41000 Security System ports are connected to their hosts. When this feature is enabled, the system automatically adds a predefined value (default=50) to the Chassis Grade.

When Port Connectivity Verification detects a host connectivity error, this value is subtracted from the Chassis Grade. The system continuously runs connectivity tests at a predefined interval (Default = 4 seconds). You can change the interval with the `pingable_hosts enable -i <interval>` command.

Notes and Limitations
- Port Connectivity Verification is not supported for VSX
- Port Connectivity Verification only supports IPv4 addresses
- A port is considered to be down when all connected hosts fail to respond to pings
Syntax

pingable_hosts --help p
pingable_hosts status
pingable_hosts load_ips
pingable_hosts disable
pingable_hosts enable

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>--help</td>
<td>Show commands and syntax</td>
</tr>
<tr>
<td>status</td>
<td>Show Port Connectivity Verification status and parameters</td>
</tr>
<tr>
<td>load_ips</td>
<td>Load</td>
</tr>
<tr>
<td>disable</td>
<td>Disable Port Connectivity Verification</td>
</tr>
<tr>
<td>enable</td>
<td>Enable Port Connectivity Verification and configure options</td>
</tr>
<tr>
<td>-i &lt;interval&gt;</td>
<td>Enter a verification interval in seconds (Default = 4)</td>
</tr>
<tr>
<td>-monitor</td>
<td>Enable the monitor only mode, which does not change the Chassis grade if connectivity verification detects an error.</td>
</tr>
</tbody>
</table>

Notes:

- `asg stat` shows the Pingable Posts and verification results in the bottom row for each Chassis.

```
> asg stat

<table>
<thead>
<tr>
<th>System Status - 61000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up time</td>
</tr>
<tr>
<td>Current CPUs load average</td>
</tr>
<tr>
<td>Concurrent connections</td>
</tr>
<tr>
<td>Health</td>
</tr>
<tr>
<td>Pingable Hosts</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 1</th>
<th>ACTIVE</th>
<th>UP / Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGMs</td>
<td>3 / 3</td>
<td></td>
</tr>
<tr>
<td>Ports</td>
<td>0 / 0</td>
<td></td>
</tr>
<tr>
<td>Fans</td>
<td>4 / 4</td>
<td></td>
</tr>
<tr>
<td>SSMs</td>
<td>2 / 2</td>
<td></td>
</tr>
<tr>
<td>CNMs</td>
<td>2 / 2</td>
<td></td>
</tr>
<tr>
<td>Power Supplies</td>
<td>6 / 6</td>
<td></td>
</tr>
<tr>
<td>Pingable Hosts</td>
<td>1 / 1</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chassis 2</th>
<th>ACTIVE</th>
<th>UP / Required</th>
</tr>
</thead>
<tbody>
<tr>
<td>SGMs</td>
<td>3 / 3</td>
<td></td>
</tr>
<tr>
<td>Ports</td>
<td>0 / 0</td>
<td></td>
</tr>
<tr>
<td>Fans</td>
<td>4 / 4</td>
<td></td>
</tr>
<tr>
<td>SSMs</td>
<td>2 / 2</td>
<td></td>
</tr>
<tr>
<td>CNMs</td>
<td>2 / 2</td>
<td></td>
</tr>
<tr>
<td>Power Supplies</td>
<td>6 / 6</td>
<td></td>
</tr>
<tr>
<td>Pingable Hosts</td>
<td>0 / 1  (!)</td>
<td></td>
</tr>
</tbody>
</table>
```

- The `UP/Required` column shows the verification status, not the number of pingable hosts up or required. The status means:
  - 1 / 1 = OK
  - 0 / 1 when one of the pingable hosts on the list fails to reply

- Port Connectivity log files are stored at `/var/log/pingable_hosts`

- The default Port Connectivity Verification value added to the Chassis Score is 50. To change this value, run

```
> set chassis high-availability factors pnote pingable_hosts <factor>
```
Working with Pingable Hosts

Before you can use Port Connectivity Verification, you must first define your interfaces and host IPv4 addresses in the `$FWDIR/conf/pingable_hosts.ips` configuration file. When this task is completed, you import the definitions to your SGMs and the enable Port Connectivity Verification.

Port Connectivity Verification is disabled by default.

**To define interfaces and host IP addresses:**

1. On an SGM, open `$FWDIR/conf/pingable_hosts.ips` in a text editor.
2. Enter the interface and host IPv4 address with this syntax:
   `<if_name>;ipv4;<Host_ip>,<Host_ip>...`
   
   Example: `eth0-01;ipv4;192.168.2.41,192.168.2.88,192.168.2.123`
   
   Each line contains one port definition, which can include one interface and many host IP addresses separated by commas. Do not put any other data in this file.
3. Run `pingable_hosts load_ips`.
   
   Example:
   
   `pingable_hosts load_ips`
   
   New IPs loaded successfully
   
   Ports and IPs:
   
   -----------
   `eth0-1;ipv4;192.168.2.88,192.168/2.123`
   `eth1-01;ipv4;10.2.2.1,10.10.2.2,10.30.2.3`
   
   Pingable hosts is DISABLED

**To enable Port Connectivity Verification:**

Run `pingable_hosts enable`.

Example:

```
# pingable_hosts enable
1_01:
1_02:
1_03:
No additional settings, using default values:
enable=1 interval=4 monitor=0
```

This action updates the Chassis Grade.

**To disable Port Connectivity Verification:**

Run `pingable_hosts disable`.

This action updates the Chassis Grade.

---

**Verifying VSX Gateway Configuration (asg vsx_verify)**

**Description**

Use this command to verify that all SGMs have the same VSX Configuration: Interfaces, Routes, and Virtual Systems configuration.

**Note** – Run `asg vsx_verify` only from the VS0 context
Syntax

`asg vsx_verify [-a] [-v] [-c]`

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-v</td>
<td>Include Virtual Systems Configuration Verification table</td>
</tr>
<tr>
<td>-a</td>
<td>Include SGMs in admin_down state</td>
</tr>
<tr>
<td>-c</td>
<td>Compare:</td>
</tr>
<tr>
<td></td>
<td>- Database configuration between SGMs</td>
</tr>
<tr>
<td></td>
<td>- Operating system and database configuration on each SGM.</td>
</tr>
</tbody>
</table>

Example 1

`asg vsx_verify -v`

Output

```
+-----------------------------------------------------------------------------|
| Chassis 1 SGMs:                                                             |
| 1_01* 1_02 1_03 1_04                                                       |
+-----------------------------------------------------------------------------|
| Chassis 2 SGMs:                                                             |
| 2_01 2_02 2_03 2_04                                                        |
+-----------------------------------------------------------------------------|

+-----------------------------------------------------------------------------|
<p>| VSX Global Configuration Verification                                      |
| +-----------------------------------------------------------------------------|
| |VSX Configuration Signature|VSX Configuration ID |State |
| +-----------------------------------------------------------------------------|
| all |8ef02b3e73386afd6e044c78e466ea82 |5\25 |UP |</p>
<table>
<thead>
<tr>
<th>+-----------------------------------------------------------------------------</th>
</tr>
</thead>
</table>

Comparing Routes DB & OS. This procedure may take some time...  
Press 'y' to skip this procedure...  
Comparing..  

+-----------------------------------------------------------------------------|
| Summary                                                                     |
| +-----------------------------------------------------------------------------|
| VSX Configuration Verification completed successfully                        |
+-----------------------------------------------------------------------------|

All logs collected to /var/log/vsx_verify.136046320.log```
Example 2

```
Example 2  

asg vsx_verify -v -a
```

```
<table>
<thead>
<tr>
<th>Chassis 1 SGMs:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_01* 1_02 1_03 1_04</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>Chassis 2 SGMs:</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
<tr>
<td>2_01 2_02 2_03 2_04</td>
</tr>
<tr>
<td>-----------------</td>
</tr>
</tbody>
</table>

VSX Configuration Verification

<table>
<thead>
<tr>
<th>SGM</th>
<th>VSX Configuration Signature</th>
<th>Virtual Systems</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>1_01</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>1_02</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>1_03</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>1_04</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>DOWN</td>
</tr>
<tr>
<td>2_01</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>2_02</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>2_03</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
<tr>
<td>2_04</td>
<td>8ef02b3e73386af6e044c78e466eaa82</td>
<td>5\25</td>
<td>UP</td>
</tr>
</tbody>
</table>

Comparing Routes DB & OS. This procedure may take some time...
Press 'y' to skip this procedure...
Comparing...

Virtual Systems Configuration Verification

<table>
<thead>
<tr>
<th>VS</th>
<th>SGM</th>
<th>VS Name</th>
<th>VS Type</th>
<th>Policy Name</th>
<th>SIC State</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>all</td>
<td>VSX_OBJ</td>
<td>VSX Gateway</td>
<td>Standard</td>
<td>Trust</td>
<td>Success</td>
</tr>
<tr>
<td>1</td>
<td>all</td>
<td>Virtual Router</td>
<td>Default Policy</td>
<td>Trust</td>
<td>Success</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>all</td>
<td>VSW-INT</td>
<td>Virtual Switch</td>
<td>Not Applicable</td>
<td>Trust</td>
<td>Success</td>
</tr>
<tr>
<td>3</td>
<td>all</td>
<td>VS-1</td>
<td>Virtual System</td>
<td>Standard</td>
<td>Trust</td>
<td>Success</td>
</tr>
<tr>
<td>4</td>
<td>all</td>
<td>VS-2</td>
<td>Virtual System</td>
<td>Standard</td>
<td>Trust</td>
<td>Success</td>
</tr>
</tbody>
</table>

Comparing Routes DB & OS. This procedure may take some time...
Press 'y' to skip this procedure...
Comparing...

Summary

VSX Configuration Verification completed with the following errors:
1. [1_02:1] eth1-06 operating system address doesn't match
2. [1_02:1] eth1-06 DB address doesn't match
3. [1_01:1] Found inconsistency between addresses in operating system ,DB and NCS of eth1-06

All logs collected to /var/log/vsx_verify.1360886320.log
# Resetting SIC (g_cpconfig sic init)

**Description**

Use this command to reset Secure Internal Communication (SIC) between the gateway and the Security Management server. For example if you replace the management server you must reset the SIC.

**Important**

This procedure causes downtime for the system and traffic outage because all SGMs are rebooted.

## Resetting SIC on a Security Gateway or VSX Gateway (VS0)

The procedure for resetting SIC on a Security Gateway or VSX Gateway (VS0) has a few stages.

### Stage 1: Initializing SIC on the Gateway

1. Use a serial console to connect to the gateway.
2. Enter the Expert shell
3. Find out which SGM is the SMO. Run `asg stat -i tasks`
4. Run:
   ```
   g_cpconfig sic init <activation key>
   ```

**Notes**

- The SIC Reset procedure lasts several (about 3 to 5) minutes.
- During the SIC reset procedure:
  - On a Security Gateway: All SGMs other than the SMO reboot
  - On a VSX Gateway: It is Mandatory to perform Stage 2 immediately when the SIC procedure is done.

### Stage 2: Initializing SIC In SmartDashboard

1. On the Gateway object, open the General Properties > Communication window.
2. Click Reset.
3. Enter the same activation key used in Stage 1.
4. Click Initialize.
5. On a VSX Gateway:
   a) Install Policy on the VSX gateway.
   b) At the serial console connection to the gateway, press ‘c’ to complete procedure. Note At this stage, all SGMs except the SMO do a reboot.

### Stage 3. Verifying Trust is established on the Gateway

Run `g_cpconfig sic state`:
```
[Expert@61000/41000 Security System-Box:0]# g_cpconfig sic state
  |- 6 blades: 1_01 1_02 1_03 2_01 2_02 2_03  
Trust State: Trust established
```

## Reset SIC for non-VS0 Virtual Systems

To reset SIC on Virtual Systems that are not VS0 (a non-VSX object).

1. Log into the SMO with a SSH client.
2. Go the Expert mode.
3. Run this command to go to the applicable context ID:
   ```
   vsenv <vsid>.
   ```
4. Run this command to initialize SIC:
   ```
   # g_cpconfig sic init
   ```
5. Revoke the VSID certificate defined in the management server.
7. In SmartDashboard, open and then save Virtual System object. This action "pushes" the configuration to the management server and re-establishes SIC trust with the SMO.
8. Install a policy on the Virtual System.

**Troubleshooting SIC reset**

SIC reset requires 3-5 minutes. If SIC reset was interrupted (for example by loss of network connectivity), run: `g_cpconfig sic state` to get the SIC state. If the SIC State is:

<table>
<thead>
<tr>
<th>SIC state</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trust established</td>
<td>Reboot all SGMs</td>
</tr>
<tr>
<td>Initialized but Trust</td>
<td>1. In SmartDashboard &gt; General Properties &gt; Communication window</td>
</tr>
<tr>
<td>was not established</td>
<td>initialize SIC</td>
</tr>
<tr>
<td></td>
<td>2. Install the policy</td>
</tr>
</tbody>
</table>

**SIC Cleanup**

To resolve other SIC issues, do a SIC cleanup. There are two ways to do a SIC cleanup:

Run `asg_blade_config reset_sic -reboot_all <activation_key>`

OR

1. Shutdown all SGMs (but not the SMO) using the `ccutil` command in Expert shell.
2. Connect to the SMO using a serial console.
3. Initialize SIC in SmartDashboard > General Properties > Communication.
4. Install policy on the SMO.
5. Turn on all SGMs.

**Troubleshooting Hardware**

This section describes common problems encountered with 61000/41000 Security System hardware components along with its corresponding resolution.

**Hardware components:**

**Security Gateway Module (SGM)**

<table>
<thead>
<tr>
<th>Problem</th>
<th>SGM does not detect part of its RAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cause</td>
<td>One or more DIMMs are not properly installed</td>
</tr>
<tr>
<td>Resolution</td>
<td>Re-assemble DIMMs in problematic SGM</td>
</tr>
<tr>
<td>Validation</td>
<td>Run <code>asg resource</code> to verify all SGMs properly report its RAM size</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Problem</th>
<th>SGM fails to boot and does not enter BIOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cause</td>
<td>SGM BIOS is corrupted</td>
</tr>
<tr>
<td>Resolution</td>
<td>CMOS reset:</td>
</tr>
<tr>
<td></td>
<td>1. On the left side of the SGM there is a yellow jumper</td>
</tr>
<tr>
<td></td>
<td>This jumper resides on the leftmost two pins (pins 1 and 2)</td>
</tr>
<tr>
<td></td>
<td>2. Pull the jumper out and put it on pins 2 and 3</td>
</tr>
<tr>
<td></td>
<td>3. Keep state 2 for 10 seconds</td>
</tr>
<tr>
<td></td>
<td>4. Pull the jumper out and put it back on pins 1 and 2</td>
</tr>
</tbody>
</table>
Validation  
SGM will start loading and the user will be able to enter BIOS

Problem  
SGM fails to start, SSD not detected at boot time
Cause  
SD is not properly assembled
Resolution  
1. Re-assemble SSD connectors.
2. Attach 1 connector to the SSD itself and the other 2 connectors to the motherboard
Validation  
SGM will start loading

Problem  
SGM fails to boot and constantly searching for network installation (PXE)
Cause  
There is no image loaded on the SGM
Resolution  
1. Install image from CD/PXE or USB flash drive.
2. Make sure BIOS setup is set to boot from the option you chose
Validation  
SGM will start installing new image

Problem  
Blue LED is on
Cause  
SGM is not properly attached to its slot
Resolution  
Re-seat the SGM, tighten its thumb screws and make sure handles are firmly closed
Validation  
Blue LED should disappear and SGM will start loading

Problem  
SGM speed LEDs are not yellow/orange
Cause  
SGM is connected to SSM with wrong speed
Resolution  
1. Restore manufacturing-defaults on the associated SSM.
2. RMA the SGM
Validation  
Verify that all speed LEDs on SGM are yellow/orange

Problem  
SGM constantly boots after it was down and major configuration changes were made to the system
Cause  
Old configuration conflicts with existing configuration
Resolution  
1. Export snapshot from both problematic and stable SGMs and attach to support ticket
2. Make sure FCD image is aligned with existing SGMs
   Note: FCD image is created during clean installation
3. Revert to FCD image
Validation  
SGM should join security group, pull the configuration and become up

Problem  
CPU type does not match customer's order, CPU type test fails in asg diag
Cause  
Customer received SGM220 instead of SGM220T or vice versa
Resolution  
MA the SGM
Validation  
N/A

Problem  
All SGMs beyond certain amount of time fail to boot and enter blue LED
Cause  Some power supply units are not connected. Minimum of 4 PSUs is required for fully populated system

Resolution  Make sure all PSUs are properly attached to the Chassis by pushing the insertion latch. Verify that all power cords are plugged

Validation  All SGMs in the Chassis should be able to start

**Chassis Management Module (CMM)**

**Problem**  Blue LED on the CMM

**Cause**  CMM is not properly assembled

**Resolution**  Re-seat the CMM

**Validation**  Verify the Blue LED on the CMM turned off

**Problem**  Power Supplies are not monitored

**Cause**  Chassis type is not configured properly in the CMM

**Resolution:**
1. Login via serial console to the CMM and repeat the CMM installation process (install.sh).
2. Select correct Chassis type:
   - Telkor - 3 PSUs per 1U
   - Lambda - 5 PSUs per 1U
3. In case of dual CMM, perform it on each CMM individually.

**Validation**  Run asg stat -v and check the power supplies amount

**Problem**  Failed to install 2nd Chassis in dual Chassis setup

**Cause**  Chassis ID is identical on both Chassis

**Resolution**
1. Login via serial console to the CMM
2. Set the Chassis ID by editing the SHMM_CHASSID in /etc/shmm.cfg
3. Reboot the CMM

**Validation**  Check whether the installation on the 2nd Chassis works

**Problem**  CMM firmware is different after CMM failover

**Cause**  CMM firmware mismatch in a dual CMM

**Resolution**  Upgrade the faulty CMM individually and reseat the other CMM

**Validation**  Verify version by invoking asg_version

**Problem**  No connectivity to the CMM through one of the CIN interfaces

**Cause**  CMM interface is set to the front panel instead of the backplane

**Resolution:**
1. Remove CMM
2. Change JP4 jumpers' position to 2-3
3. Plug in the CMM

**Validation**  Run asg stat -v and check the CMM amount

**Problem**  Alarm LED is on
**Cause**  High temperature in Chassis surroundings  

**Resolution:**
1. Login via serial console to the CMM and reset LED by running: clia alarm 0  
2. Make sure that all open slots (missing SGMs/CMMs/PSUs) are covered with blanks  
3. Make sure that all fans are properly attached  
4. Make sure there is proper cooling in Chassis surroundings  

**Validation**  Alarm LED should remain off

---

**Security Switch Module (SSM)**

**Problem**  Blue LED on the SSM  
**Cause**  SSM is not properly assembled  

**Resolution**
1. Reseat the SSM  
2. If SSM cannot be attached due to broken latch - RMA  

**Validation**  Verify the Blue LED on the SSM turned off

---

**Problem**  asg dxl dist_mode verify failed and there are traffic issues on pseudo interfaces  
**Cause**  SSM distribution configuration is not set properly  

**Resolution**
1. reset distribution mode and verify it  
2. In case it didn't solve, login into the appropriate SSM and invoke load-balance apply  
3. As a last resort, invoke system reload manufacturing-defaults  

**Validation**  Run asg dxl dist_mode verify

---

**Problem**  Connectivity issues between SGMs  
**Cause**  Invalid SSM configuration on Sync interfaces  

**Resolution**  Login into the appropriate SSM and run system reload manufacturing-defaults  

**Validation**  Verify connectivity between the SGMs by invoking asg monitor

---

**Problem**  Connectivity issues between SGMs on different Chassis  
**Cause**  Sync ports are connected through 1G link/transceivers are not compatible with distance  

**Resolution:**
Connect the Sync ports to 10G links, using LC fiber optic. Make sure to use SR/LR transceivers, according to distance  

**Validation**  Verify connectivity between SGMs on different Chassis by invoking asg monitor

---

**Problem**  No link on SSM traffic port  
**Cause**  Uncertified transceivers or incorrect port speed  

**Resolution:**
1. Check if the transceivers are certified by invoking the command asg diag verify and check whether the "Media Details" Passed  
2. Run asg_chassis_ctrl get_port_admin_speed <ssm_id> <port_id> to verify the fan speed  
3. If necessary, run asg_chassis_ctrl set_port_speed <ssm_id> <port_id> <speed>  

**Validation**  Verify link on the port and connectivity (if possible)
Problem: No link on SSM management port
Cause: Uncertified transceivers or incorrect port speed

Resolution:
1. Check if the transceivers are certified by connecting them to one of the traffic ports and invoke transceiver_verifier
2. Verify the port speed
3. Login to the relevant SSM
4. Run show port 1/5/<management port index, 1-4>
5. Reset port speed
6. Login to the relevant SSM
7. Enter conf t
8. Set port speed to different value:
   port 1/5/<management port index> speed 100
9. Set port speed back to the desired value:
   port 1/5/<management port index> speed 1000

Validation: Verify link on the port and connectivity (if possible)

Problem: Silent installation on other SGMs does not work after FTW finished
Cause: SSM version is incorrect or has an invalid configuration

Resolution:
1. Login into the appropriate SSM and verify version by invoking show version
2. In case the version is incorrect, please upgrade the SSM
3. Otherwise, run system reload manufacturing-defaults

Validation: Verify that silent installation on other SGMs completed after 5-10 minutes

Fans
Problem: Blue LED on fan
Cause: Fan is not properly assembled

Resolution: Extract and insert the fan again. Lock the captive screw (where applicable).

Validation: Verify the blue LED on the fan is turned off and all fans rotate at normal speed.

Problem: Rotation speed is too high, fans are extremely noisy
Cause: High temperature in Chassis surroundings

Resolution:
- Make sure that all open slots (missing SGMs/CMMs/PSUs) are covered with blanks.
- Verify that all fans are properly attached.
- Make sure there the temperature is sufficiently cool around the Chassis.

Validation: asg hw_monitor indicates that fans rotation speed is normal and the threshold is not crossed

Power Supply Unit (PSU)
Problem: Blue LED on PSU
Cause: PSU is not properly assembled

Resolution: Make sure all PSUs are properly attached to the Chassis by pushing the insertion latch

Validation: Run: asg stat -v and check PSUs amount
Problem: After Chassis RMA: rightmost Chassis components (SGMs, PSUs) are not monitored
Cause: DC PEMs are missing
Resolution: Move DC PEMs from the old to the new Chassis.
Validation: Verify that all Chassis components are monitored

Problem: asg diag reports that power unit is misplaced
Cause: Telkor PSUs should be placed from upper right to bottom left
Resolution: Place 5 Telkor PSUs as follows: 3 in upper tray, 2 in bottom tray.
Leaving the leftmost bay in bottom tray empty and covered with blank
Validation: asg diag should not warn about PSU misplacement

## Debug files

These are the debug files that relate to the 61000/41000 Security System:

<table>
<thead>
<tr>
<th>Feature</th>
<th>Debug File</th>
</tr>
</thead>
<tbody>
<tr>
<td>FWK</td>
<td>$FWDIR/log/fwk.elg.*</td>
</tr>
<tr>
<td>Policy</td>
<td>$FWDIR/log/cpha_policy.log.*</td>
</tr>
<tr>
<td>SGM Configuration / Pull Configuration</td>
<td>$FWDIR/log/blade_config.*</td>
</tr>
<tr>
<td>Alerts</td>
<td>/var/log/send_alert.*</td>
</tr>
<tr>
<td>Distribution</td>
<td>$FWDIR/log/dist_mode.log.*</td>
</tr>
<tr>
<td>Installation – OS</td>
<td>/var/log/anaconda</td>
</tr>
<tr>
<td>Installation – 61000/41000 Security System</td>
<td>/var/log/start_mbs.log</td>
</tr>
<tr>
<td>Installation – 61000/41000 Security System</td>
<td>/var/log/mbs.log</td>
</tr>
<tr>
<td>Dynamic Routing</td>
<td>/var/log/routed.log</td>
</tr>
<tr>
<td>CPD</td>
<td>$CPDIR/log/cpd.elg</td>
</tr>
<tr>
<td>FWD</td>
<td>$FWDIR/log/fwd.elg</td>
</tr>
<tr>
<td>General</td>
<td>/var/log/messages*</td>
</tr>
<tr>
<td>SMD</td>
<td>/var/log/smd_smo.log</td>
</tr>
<tr>
<td>SMD</td>
<td>/var/log/smd.log</td>
</tr>
<tr>
<td>Log servers</td>
<td>/var/log/log_servers*</td>
</tr>
<tr>
<td>Pingable hosts</td>
<td>/var/log/pingable_hosts*</td>
</tr>
<tr>
<td>Clish auditing</td>
<td>/var/log/auditlog*</td>
</tr>
<tr>
<td>Command auditing</td>
<td>/var/log/asgaudit.log*</td>
</tr>
<tr>
<td>VPND</td>
<td>$FWDIR/log/vpnnd.elg*</td>
</tr>
<tr>
<td>Reboot logs</td>
<td>/var/log/blade_reboot_log</td>
</tr>
</tbody>
</table>

---

61000/41000 Security System Administration Guide R76SP | 245