Symantec Edge Secure Web Gateway (SWG) with SGOS v7.4

Common Criteria Administrative Guidance Document

Document Version: 0.8

Contact Information

Symantec, A Division of Broadcom
1320 Ridder Park Dr,
San Jose, CA 95131
www.broadcom.com

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Broadcom shall have no liability for any error or damages of any kind resulting from the use of this document.

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.

Revision History

Version	Date	Changes
Version 0.1	December 12, 2022	Initial Release
Version 0.2	February 6, 2023	Updated after Internal review
Version 0.3	February 13, 2023	Updated after Internal review
Version 0.4	March 28, 2023	Updated after Internal review
Version 0.5	April 5, 2023	Updated after Internal review
Version 0.6	May 26, 2023	Updated after Internal review
Version 0.7	June 23, 2023	Updated to add password complexity support
Version 0.8	August 4, 2023	Updated after version change

Table of Contents

1 Introduction

1.1 Overview

This Administrative Configuration Guide documents the administration of the Symantec Edge Secure Web Gateway (SWG) with SGOSv7.4 certified under Common Criteria (CC). The TOE may be referenced below as the ProxySG, Edge SWG, SGOS or TOE. This document is written for administrators configuring the TOE. This document is the Configuration Guide for the CC evaluation, fulfilling the AGD security assurance requirement. The purpose of this document is to highlight the administrator functions and interfaces necessary to configure and maintain the TOE in the evaluated configuration.

This document assumes that the administrator is a trusted individual, and the various operating systems run within the network. The administrator configuring the TOE must thoroughly use this Administrative Configuration Guide and refer to the documents identified in Section 7.

Note: The administrator using this Administrative Guidance is required to generate his own trusted certificate chain and upload to the TOE by deleting the default one's (Refer to sections 2.10 and 2.12).

1.2 Supported Hardware Platforms

The following Symantec Appliances are supported:

Model	Firmware Version
SSP-S410-20 with ISG using Intel Xeon Silver 4210 Processor (Cascade Lake)	7.4.1.1
VMware ESXi 6.5 Hypervisor hosted on Dell Power Edge R440, with Intel Xeon Silver 4216 Processor (Cascade Lake)	7.4.1.1

Table 1 - Hardware/Firmware Versions

1.3 TOE Environment

Component	Required	Usage/Purpose Description for TOE performance
Remote Management Workstation (GUI).	No	This includes any IT Environment Management workstation with a web browser installed that is used by the TOE administrator to support TOE administration through HTTPS and TLS protected channels.
Remote Management Workstation (CLI).	Yes	This includes any IT Environment Management workstation with an SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels.
Local Management Workstation (CLI).	Yes	This includes any IT Environment Management workstation with a local CLI support that is used by the TOE administrator to support TOE administration through a direct connection.
NTP Server	Yes	NTP server supporting SHA-1 integrity verification.
Audit Server	Yes	The audit server is used for remote storage of audit records that have been generated by and pulled from the TOE.
CA/OCSP Server	Yes	A server with a certification authority and certificate revocation list used by the TOE for validating the X.509 certificates used for TLS connection establishment.

Table 2 - Hardware/Firmware Versions

1.4 Cryptographic Support

The following section describes the evaluated cryptographic services provided by the TOE.

1.4.1 Algorithm Certificates

The table below lists the algorithm certificates issued by the CAVP.

Cryptographic Method	Use within the TOE	CAVP Certificate #
AES	• TLS Traffic Encryption/Decryption • SSH Traffic Encryption/Decryption	A2936
RSA	• TLS Session Establishment • SSH Session Establishment • Software Upgrade	A2936
SP800-90A	• TLS Session Establishment • SSH Session Establishment	A2936
SHS	• Used to provide TLS traffic integrity verification • Used to provide SSH traffic integrity verification	A2936
HMAC-SHS	• Used to provide TLS traffic integrity verification • Used to provide SSH traffic integrity verification	A2936
SP800-56A	• TLS Session Establishment • SSH Session Establishment	A2936
SP800-135rev1	• TLS Session Key Derivation • SSH Session Key Derivation	A2936

Table 3 - Cryptographic Algorithms

2 Accessing the TOE Using the CLI Console

The Management Console is a Serial interface that allows you to manage, configure, monitor, and upgrade the appliance from any location. After FIPS mode has been enabled on an appliance per the instructions, you must use SSH from a server or desktop that has the proper ciphers.

```
Management Console started
Press "enter" three times to activate the serial console.
Welcome to serial console of SGOS

Do you accept the terms and conditions [Yes/No]: Yes
       Welcome to the Blue Coat SG-VA Series Appliance Serial Console

Version: SGOS 7.4.0.0, Release id: 279238 64-bit, gdb, unoptimized

------------------------ MENU ----------------------------

1) Command Line Interface
2) Setup Console

------------------------------------------------------------

Enter option:
Welcome to the Blue Coat SG-VA Series Appliance command line interface

Type "exit" at the main prompt to quit

Username: admin
Password:

Symantec Edge Secure Web Gateway (SWG) with SGOS
10.1.5.122 - Blue Coat SG-VA Series>en
Enable Password:
10.1.5.122 - Blue Coat SG-VA Series#
```

Figure 1 - TOE CLI Console

2.1 Ways to Access the CLI Console

We can Access the management console as shown in Figure 2. Follow the steps below:

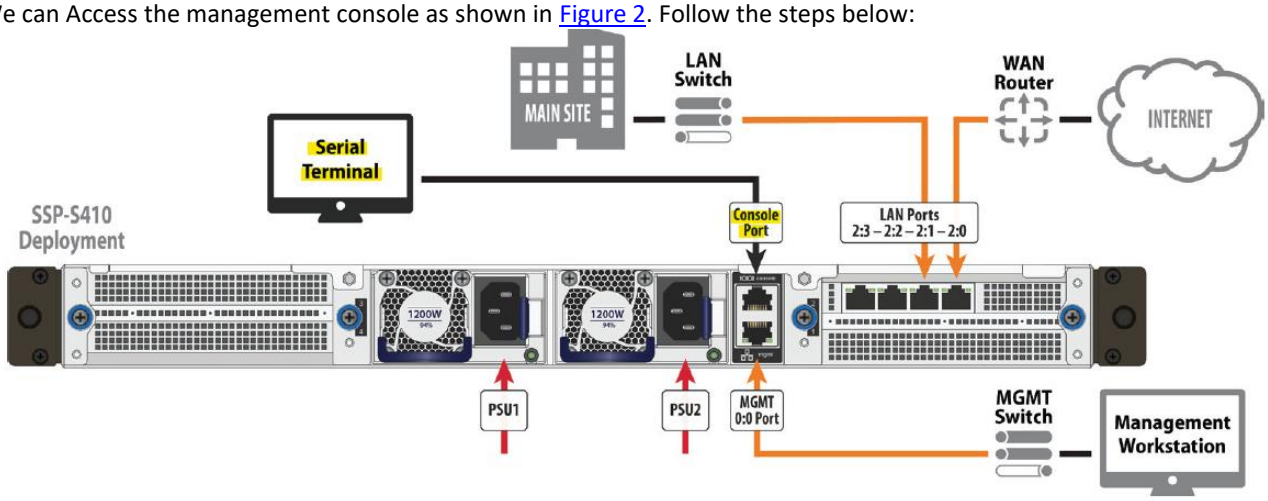

Figure 2 - SSP-S410-20 Deployment

2.1.1 Delivery of the TOE

- Customers with an active account may download the TOE securely from: https://support.broadcom.com/group/ecx/downloads?

Software Version	Image Name	Hash
SGOS 7.4.1.1 SWG Edition Release ID: 287291	EdgeSWG7.4.1_build287291.bcsi	**MD5** - 143fbc9d80d2323f9861ac667b8d329e **SHA256** - 37c08e40cad8644435d343a4ac7598dbe08b14f2fda2f273e4c8084f749248cb

Table 4 – Evaluated Software Image

- The TOE build maintains integrity throughout the delivery process by limiting access to current customers, supporting downloads over TLS, and providing an MD5 and SHA-256 hash for TOE verification post download.
- The following sections explains the installation process of the TOE.

2.1.2 Install ProxySG Application Image

Before you create and start an application, load the application image onto the ISG. ISG is the platform on which an application runs.

1. To install application images on the host appliance, perform the following steps:
   ```
   localhost# config
   localhost(config)# images
   localhost(config-images)# load <application_location_URL>
   ```
2. View all downloaded images:
   ```
   (config-images)# view
   ```
3. View a specific image:
   ```
   (config-images)# view image_id
   ```
4. View all ProxySG images:
   ```
   (config-images)# view sg
   ```
5. Remove the image:
   ```
   (config-images)# delete image_id
   ```

Note: Application images have a .bcsi extension and are available for download from: support.broadcom.com/security

2.1.3 Load the Management Console in ISG

1. Power on the Appliance and verify LEDs.
2. Confirm the appliance's Console port is connected to a serial terminal or workstation with terminal emulation software.
3. Open a terminal emulation program, such as HyperTerminal®, PuTTY, or Tera Term and configure it to use the following settings:

Baud Rate	9600bps
Data bits	8
Flow Control	None
Parity	None
Stop bit	1

4. On the Menu screen, press 2 to open the Setup Console.
5. Enter the IP address, IP subnet mask, IP gateway, and DNS server for the appliance's network ports.
6. Specify the console password and enable password. Press any key to activate and return to the serial console.
7. To access privileged commands on the host appliance, enter:
   ```
   localhost# enable
   ```

```
Enable Password: <host-enable-password>
```

8. To install a license on the host appliance, perform the following steps.On the Menu screen, press 1 to open CLI.
    - For node-locked licenses, enter: **licensing load id** <serial-number>
    - For ISG application licenses, enter: **licensing load id** <isg-license>

Note: You can locate your serial number or ISG license in the e*Fulfillment Letter* you received from Symantec at the time of purchase.

2.1.4 Create a ProxySG Application

1. Confirm the appliance's Console port is still connected to the serial terminal and that the terminal software is configured as in Section 2.1.1 step 3.
2. On the Menu screen, press 1 to open CLI.
3. To access privileged commands on the host appliance, enter:
```
localhost# enable
Enable Password: <host-enable-password>
```
4. To add an ISG application, enter:
```
localhost# config
localhost(config)# applications
localhost(config-applications)# create sg sg_name model model_name license-id license_id image-id image_id
ok
```

Note: To add a ProxySG application, enter sg as the app-type. Verify the number and type of ISG applications and models that can be installed on your host appliance.

5. To start an ISG application, enter:
```
localhost(config-applications)# start application_name
localhost(config-applications)# attach-console application_name
```
The following is an example output of the command:
```
config-applications)# attach-console SG1
Connected to domain sgos
Escape character is ^]
System starting up...
In MP mode; two processors active
Executing image: Version: SGOS 6.7.5.3, Release id: 249936 64-bit, gdb, optimized
Manufacturing MBR on directory-3 - Slot 3 (KVM VirtIO Disk N/A N/A)
This is a new system.
```
6. When prompted, press **Enter** three times.
7. Use the Configuration Wizard to initialize the ISG application.
    a. Press **a** to proceed with Manual Setup.
    b. Enter the **IP address, IP subnet mask, IP gateway, and DNS server** for the appliance's network ports.
    c. Specify the **console password** and **enable password**.

```
Press "enter" three times to activate the serial console
****************** CONFIGURATION ALERT ******************
System entering configuration wizard for the following reasons:
- Cannot find a network adapter configured with an IP address and subnet.
- The console password or 'enable' password is not set.
****************** CONFIGURATION ALERT ******************
--------------- CONFIGURATION START ------------------
Welcome to the Blue Coat SG-VA Series configuration wizard.
This appliance's serial number: 0000990007
-------------------------------------------------------------------
You can get field help by entering a question mark ? in the fields.
You can move backwards through the steps by pressing the UP arrow.
You can exit the wizard without saving your entries by pressing ESC.
-------------------------------------------------------------------
Step 1: How do you plan to configure this appliance?
a) Through a manual setup
```

```
b) Through a Director-managed setup
Your choice: []
```

8. To add additional ISG applications, enter the following commands to disconnect from the new application, then repeat the previous steps.

```
localhost - Blue Coat SG-VA Series> Press Ctrl+]
telnet> send escape
```

9. To view application information, such as license IDs, image IDs, and other properties that are associated with your applications, use the applications view command (in either enable or configuration mode). For example:

```
(config-applications)# view
NAME TYPE VCPU MEMORY MODEL STATUS LICENSE ID IMAGE ID
------------------------------------------------------------------------
SG1 SG 2 20 GB C2S Running 000090007 sg-7.3.8.1-273266
SG2 SG 2 20 GB C2S Running 000090007 sg-7.4.0.0-278577
SG3 SG 2 20 GB C2S Running 000090007 sg-7.4.0.0-280944

(config-applications)# view SG1
NAME TYPE VCPU MEMORY MODEL STATUS LICENSE ID IMAGE ID
------------------------------------------------------------------------
SG1 SG 2 20 GB C2S Running 000090007 sg-7.3.8.1-273266
```

2.1.5 Perform a Graceful Shutdown

1. Confirm the appliance's **Console** port is still connected to the serial terminal and that the terminal software is configured as in **Section 2.1.1 Step 3**.
2. On the **Menu** screen, press **1** to open the CLI.
3. To access privileged commands on the host appliance, enter:

```
localhost# enable
Enable Password: <host-enable-password>
```

4. To close any open ISG applications, enter:

```
localhost# configure
localhost(config)# applications
localhost(config-applications)# attach-console <app-name>
localhost - Blue Coat SG-VA Series> enable
Enable Password: <app-enable-password>
localhost - Blue Coat SG-VA Series> shutdown
If prompted... "shut down the appliance?" [no,yes] yes
Wait for... "It is now safe to power off the system."
localhost - Blue Coat SG-VA Series> Press Ctrl+]
telnet> send escape
localhost(config-applications)# stop <app-name>
localhost(config-applications)# exit
localhost(config)# exit
```

5. To power off the host appliance, enter:

```
localhost# shutdown
When prompted... "shut down the appliance?" [no,yes] yes
```

2.2 Accessing the TOE Using SSH

- You can connect to the ProxySG appliance command line interface via Secure Shell (SSH) using the IP address, username, password that you defined during initial configuration.
- The SSH management console service is configured and enabled to use SSHv2 and a default SSH host key by default.
- If you wish to access the CLI, you can use SSHv2 to connect to the ProxySG appliance.
- An SSH host key for SSHv2 and an SSH management service are configured by default.
- To log in to the CLI, you must have:

- the account name that has been established on the appliance
- the IP address of the appliance
- the port number (22 is the default port number)

- SGOS supports different levels of command security:
  - **Standard, or unprivileged, mode is read-only**. You can see but not change system settings and configurations. This is the level you enter when you first access the CLI.
  - **Enabled, or privileged, mode is read-write**. You can make immediate but not permanent changes to the appliance, such as restarting the system. This is the level you enter when you first access the Management Console.
  - **Configuration mode** allows you to make permanent changes to the appliance configuration. To access Configuration mode, you must be in Enabled mode.
- If you use the SSH, you must enter each level separately;

```
Username: admin
Password:
> enable
Enable Password:
#configure terminal
Enter Configuration commands, one per line. End with CTRL-Z.
```

- To log off from the TOE using SSH or Console use the following command:

```
#exit
```

2.3 Changing the Login Parameters

You can change the console username and password, the console realm name which displays when you log in to the appliance, and the auto-logout time. The default value for the auto-logout time is 900 seconds.
The Management Console requires a valid administrator username and password to have full read-write access; you do not need to enter a privileged-mode password as you do when using the CLI. A privileged-mode password, however, must already be set.

2.3.1 Changing the Administrator Account Credentials

During the initial configuration of your ProxySG appliance, a console administrator username and password was created. This is a special account that can always be used to administer the appliance from either the web-based Management Console or the Command Line Interface. You can change the username and the password of this administrator account.

Note: In order to meet NDcPP compliance, the password must contain at least at least 15 characters long with the following complexity:

- At least one uppercase letter
- At least one lowercase letter
- At least one numbers
- At least one of the following special characters: [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“’”, “+”, “-“, “=”, “.”, “/”, “\”, “:”, “;”, “<”, “>”, “[“, “]”, “_”, “{“, “}”, “|”, “~” “`”]]
- To change the username and password of the console use the following commands:

```
# (config) security  username <user name>
# (config) security password ?
 <Enter>
 <password>
```

This is the password required to enter enable mode from the CLI when using console credentials, the serial console, or RSA SSH.

- Example:

```
#(config)security password
Enter password: ********
Confirm password: ********
Ok
```

- To change the enable-mode password use the following commands:

```
#(config)security enable-password
Enter password:
Confirm password:
```

- Example:

```
#(config)security enable-password
Enter password: ********
Confirm password: ********
Ok
```

2.4 Time Settings

To manage objects, the ProxySG appliance must know the current Coordinated Universal Time (UTC), which is the international time standard and is based on a 24-hour clock. The user is required to logout of the session and close the browser, then reauthenticate after making changes to the local time.

2.4.1 Synchronizing to the Network Time Protocol

- The Network Time Protocol (NTP) is used to synchronize the time of a computer client or server to another server or reference time source, such as a radio or satellite receiver or modem. There are more than 230 primary time servers, synchronized by radio, satellite and modem.
- The ProxySG appliance ships with a list of NTP servers available on the Internet, and attempts to connect to them in the order they appear in the NTP server list on the NTP tab. You can add others, delete NTP servers, and reorder the NTP server list to give a specific NTP server priority over others.
- The ProxySG appliance uses NTP and the Coordinated Universal Time (UTC) to keep the system time accurate.
- You can add and reorder the list of NTP servers the appliance uses for acquiring the time. (The reorder feature is not available through the CLI.)
- You can specify NTP servers that support authentication where the time messages will be authenticated using symmetric-key encryption.
- Use this command to set NTP parameters:

  `#(config) ntp clear`

  Removes all entries from the NTP server list.

  `#(config) ntp {enable | disable}`

  Enables or disables NTP.

  `#(config) ntp encrypted-server {domain_name | IP_address} key_id key_type encrypted_key`

  Add a server to the server list, where encrypted_key is a key in an encrypted format.

  `#(config) ntp server`

  `#(config) ntp interval minutes`

  Specifies how often to perform NTP server queries.

  `#(config) ntp no server {domain_name | IP_address}`

  Removes the specified NTP server from the NTP server list.

  `#(config) ntp server {domain_name | IP_address} [key_id key_type [key]]`

  Add a server to the NTP server list, using either the domain name of an NTP server that resolves to an IPv4 or IPv6 address, or the IPv4 or IPv6 address of an NTP server.

  If the server supports authentication, you can specify the authentication key information provided to you by the NTP server authority:

  - *key_id* is a value from 1 to 65534

- *key_type* is the string *sha1*
- *key* is the plaintext shared secret from the NTP authority

- Use this command to see NTP parameters:
  > show ntp
  Displays NTP servers status and information.

2.4.2 Changing the TOE Timeout

- The timeout is the length of time a CLI or SSH session persists before you are logged out. The default timeout for these options is as follows:
  Enforce CLI auto-logout—15 minutes (900 seconds)
- To change the CLI or SSH use the following commands:
  #(config) **security management cli-timeout** <*minutes*>
  e.g. #(config)**security management cli-timeout** 20
- Acceptable values are between 1 and 1440 minutes (60 seconds to 86400 seconds).
- To disable the automatic session logout for CLI sessions use the following commands:
  #(config) **security management no cli-timeout**

2.5 Logging Out

The administrator can explicitly log out of the local or remote sessions using the following command:

#exit

2.6 Management Services (SSH Access)

Management services are used to manage the appliance. The appliance provides administrative access to the appliance through SSH or console.
The TOE forces a rekey before reaching 1 hour or 2^28 bytes (which is less than aggregate of one gigabyte of data), whichever occurs first.

2.6.1 Managing SSH Console

When managing the SSH console, you can:

- Generate or re-generate SSH host keys.
- Specify the SSHv2 algorithm.
- Create or remove client keys and Director keys.
- Specify a welcome message for clients accessing the appliance using SSHv2.

2.6.2 Managing SSH Host Key Pairs

- The SSH console service allows to you to securely connect to the Command Line Interface. By default, SSHv2 is enabled and assigned to port 22.
- You do not need to create a new host key unless you want to change the existing configuration.
- To manage new host keypairs or global settings for all SSH console services, use the #(config) ssh-console command.
- To create a host key pair use the following command:
  #(config ssh-console) **create host-keypair** {**ecdsa** | **ed25519** | **rsa** | **<Enter>**}

- To delete a host key pair use the following command:
  #(config ssh-console) **delete host-keypair** {**ecdsa** | **ed25519** | **rsa** | **<Enter>**}

- To view the created host key pair use the following command:
  ```
  #(config ssh-console)view host-public-key {ecdsa | ed25519 | rsa | <Enter>}
  ```

2.6.3 Managing SSH Client Keys

- You can import multiple RSA client keys on the appliance to provide public key authentication, an alternative to using password authentication.
- An RSA client key can only be created by an SSH client and then imported onto the appliance.
- Many SSH clients are commercially available for UNIX and Windows.
- After you create an RSA client key following the instructions of your SSH client, you can import the key onto the appliance using the CLI.
- The user ID for each key must be unique.

About the OpenSSH.pub Format

- The ProxySG appliance consumes the client key in the OpenSSH.pub format.
- The end of the OpenSSH.pub format has a space followed by the username and machine in the form username@machine, as shown below:
  ```
  ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEAwFI78MKyvL8DrFgcVxpNRHMFKJrBMeBn
  2PKcv5oAJ2qz+uZ7hiv7Zn43A6hXwY+DekhtNLOk3HCWmgsrDBE/NOOEnDpLQjBC6t/
  T3cSQKZjh3NmBbpE4U49rPduiiufvWkuoEiHUb5ylzRGdXRSNJHxxmg5LiGEiKaoELJfsD
  Mc= user@machine
  ```
- username@machine is the username and the machine the client will connect from, and it is referred to as the key_id on the ProxySG CLI.
- Each key_id must be unique in the ProxySG configuration.

Notes:

- If you have created the key on Linux using the ssh-keygen -t rsa command, the key is likely already in the format.
- 4096 bits is the maximum supported key size.
- An ssh-rsa prefix must be present.
- When importing the client key, remove trailing newlines.

Import RSA client keys using the CLI:

1. Log in to the ProxySG CLI and enter configuration mode.
2. Type the following commands:
   ```
   #(config) ssh-console
   #(config ssh-console) inline client-key <username> <eof marker>
   <contents_of_file_~/.ssh/id_rsa.pub_from_clipboard>
   <eof_marker>
   ```
3. Display the fingerprint (a unique ID) of the imported key:
   ```
   #(config ssh-console) view client-key <username> [<keyID>
   ```

2.6.4 Configuring Ciphers

- Manage SSH ciphers on the appliance. You can add, remove, reset, and view ciphers. Fewer ciphers are available when the appliance is in FIPS mode.
- After an upgrade or downgrade, the current list of ciphers —as shown in `view` subcommand output— may change. If you modify the current list using the `add`, `remove`, and `set` subcommands, the changes persist after system upgrades, downgrades, and reboots; however, the current list will not be identical to the list prior to upgrade/downgrade if the system must consider deprecated ciphers. To understand the behavior after upgrade/downgrade:

- Ciphers that were previously added explicitly (using the `add` subcommand) are added to the current list if they are supported after upgrade/downgrade. They are not added if they are deprecated in the current version.
- Ciphers that were previously removed explicitly (using the `remove` subcommand) are removed from the current list even if they are supported in the current version.
- Ciphers that were neither added nor removed explicitly are added to the current list if supported in the current version and removed from the list if deprecated.
- If you upgrade to a release that supports only ciphers that you previously removed, resulting in an empty current list, the appliance warns you that the list is empty and event-logs the occurrence.

- For example, if you upgrade to a version of SGOS in which an added cipher is deprecated, the cipher is removed from the current list. Downgrading to the previous SGOS version adds the cipher back to the current list.

Note: The event log indicates when any ciphers are added or removed.

- **To add SSH ciphersuites use the following commands:**

```
#(config) ssh-console
#(config ssh-console) ciphers add <cipher-name>
```

Adds a new SSH cipher to the current list. The *cipher-name* must be one of the names listed under *choices* in the `ciphers view` output.

- **To remove SSH ciphersuites use the following commands:**

```
#(config) ssh-console
#(config ssh-console) ciphers remove <cipher-name>
```

Removes an SSH cipher from the current list. The *cipher-name* must be one of the names listed under *choices* in the `ciphers view` output.

- **To reset SSH ciphersuites use the following commands:**

```
#(config ssh-console) ciphers reset
```

Resets the current SSH ciphers selection to the default set of ciphers; use the `ciphers view` command to see the default cipher list.

- **To set a SSH ciphersuites use the following commands:**

```
#(config ssh-console) ciphers set <cipher-list>
```

Sets the list of SSH ciphers in the specified order, where *cipher-list* is a comma-separated list. Names in the *cipher-list* must be one of the names listed under choices in the `ciphers view` output. The ciphers you set here replace the current list.

- **To view SSH ciphersuites use the following commands:**

```
#(config ssh-console) ciphers view
```

Displays the currently selected SSH ciphers, the default set of ciphers, and the available choices of ciphers. Fewer ciphers are available or selected if the appliance is in FIPS mode.

- Example:

```
#(config ssh-console)ciphers view
current:      aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-
ctr,aes128-ctr
default:      aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-
ctr,aes128-ctr
choices:      aes256-gcm@openssh.com,aes128-gcm@openssh.com,aes256-ctr,aes192-
ctr,aes128-ctr,aes256-cbc,aes128-cbc
```

2.6.5 Configuring HMACs

- Manage SSH HMAC (Hash-based Message Authentication Code) algorithms on the appliance. You can `add`, `remove`, `reset`, and `view` HMAC algorithms. Fewer HMAC algorithms are available when the appliance is in FIPS mode.
- After an upgrade or downgrade, the current list of HMACs —as shown in view subcommand output— may change. If you modify the current list using the add, remove, and set subcommands, the changes persist after system upgrades, downgrades, and reboots; however, the current list will not be identical to the list prior to upgrade/downgrade if the system must consider deprecated HMACs. To understand the behavior after upgrade/downgrade:
  - HMACs that were previously added explicitly (using the `add` subcommand) are added to the current list if they are supported after upgrade/downgrade. They are not added if they are deprecated in the current version.
  - HMACs that were previously removed explicitly (using the `remove` subcommand) are removed from the current list even if they are supported in the current version.
  - HMACs that were neither added nor removed explicitly are added to the current list if supported in the current version and removed from the list if deprecated.
  - If you upgrade to a release that supports only HMACs that you previously removed, resulting in an empty current list, the appliance warns you that the list is empty and event-logs the occurrence.
- For example, if you upgrade to a version of SGOS in which an added HMAC is deprecated, the HMAC is removed from the current list. Downgrading to the previous SGOS version adds the HMAC back to the current list.

Note: The event log indicates when any HMACs are added or removed.

- **To add SSH hmacs use the following commands:**

```
#(config) ssh-console
#(config ssh-console) hmacs add <hmac-name>
```

Adds a new SSH HMAC algorithm to the current list. The *hmac-name* must be one of the names listed under *choices* in the `hmacs view` output.

- **To remove SSH hmacs use the following commands:**

```
#(config) ssh-console
#(config ssh-console) hmacs remove <hmac-name>
```

Removes an SSH HMAC algorithm from the current list. The *hmac-name* must be one of the names listed under *choices* in the `hmacs view` output.

- **To reset SSH hmacs use the following commands:**

```
#(config ssh-console) hmacs reset
```

Resets the current SSH HMAC list to the default set of HMAC algorithms; use the `hmacs view` command to see the default HMAC list.

- **To set a SSH hmacs use the following commands:**

```
#(config ssh-console) hmacs set <hmac-list>
```

Sets the list of SSH HMACs in the specified order, where *hmac-list* is a comma-separated list. Names in the *hmac-list* must be one of the names listed under choices in the `hmacs view` output.

- **To view SSH hmacs use the following commands:**

```
#(config ssh-console) hmacs view
```

Displays the currently selected SSH HMAC algorithm, the default set of HMAC algorithm, and the available choices of HMAC algorithm. Fewer HMAC algorithms are available or selected if the appliance is in FIPS mode.

- Example:

```
#(config ssh-console)hmacs view
current:      hmac-sha2-512,hmac-sha2-256,hmac-sha1,hmac-sha1-96
default:      hmac-sha2-512,hmac-sha2-256,hmac-sha1,hmac-sha1-96
```

```
choices:        hmac-sha2-512,hmac-sha2-256,hmac-sha1,hmac-sha1-96
```

2.6.6 Configuring Key Exchange Algorithm

- The TOE supports diffie-hellman-group14-sha1, ecdh-sha2-nistp256, ecdh-sha2-nistp384, and ecdh-sha2-nistp521 as the only allowed key exchange methods.
- You can configure the key exchange akgorithms using the following commands:
- **To add Key exchange, use the following commands:**

```
#(config) ssh-console
#(config ssh-console) kex-algs add <kex-name>
```

Adds a new key exchange algorithm to the current list. The *ssh kexs* must be one of the names listed under *choices* in the `kex-algs view` output.

- **To remove Key exchange, use the following commands:**

```
#(config) ssh-console
#(config ssh-console) kex-algs remove <kex-name>
```

Removes an key exchange algorithm from the current list. The *ssh kexs* must be one of the names listed under *choices* in the `kes-algs view` output.

- **To reset Key exchange, use the following commands:**

```
#(config ssh-console) kex-algs reset
```

Resets the current key exchange list to the default set of SSH KEXs algorithms; use the `kex-algs view` command to see the default SSH KEXs list.

- **To set a Key exchange, use the following commands:**

```
#(config ssh-console) kex-algs set <list-of-comma-separated-kex-names>
```

Sets the list of Key exchange in the specified order, where *ssh kexs-list* is a comma-separated list. Names in the *ssh kexs-list* must be one of the names listed under choices in the `ssh kexs view` output.

- **To view Key exchange, use the following commands:**

```
#(config ssh-console) kex-algs view
```

Displays the currently selected SSH kex-algs algorithm, the default set of key exchange algorithm, and the available choices of key exchange algorithm. Fewer key exchange algorithms are available or selected if the appliance is in FIPS mode.

2.7 Random Bit Generation

- The TOE produces all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using CTR_DRBG (AES).

2.8 Event Logging

- You can configure the appliance to log system events as they occur.
- Event logging allows you to specify the types of system events logged, the size of the event log, and to configure Syslog monitoring.
- The ProxySG appliance does not send e-mail notifications by default for logged events.
- You can enable e-mail notification when certain types of events occur.
- **To view the event logs, use the following command:**

```
# show event-log
```

- **To filter the event logs, use the following command:**

```
# show event-log [start "[YYYY-mm-dd] [HH:MM:SS]"] [end "[YYYY-mm-dd] [HH:MM:SS]"]
[substring <string> | regex <expression>]] | tail [<count>]
```

Example:

```
show event-log start "2022-12-08 07:28:00" end "2022-12-30 08:53:07"
```

- **To view the event logs configuration, use the following command:**

```
#show event-log configuration
```

2.8.1 Syslog Event Monitoring

- Configure the SGOS appliance to log system events as they occur.
- Event logging allows you to specify the types of system events logged, the size of the event log, and to configure Syslog monitoring.
- The appliance can also notify you by e-mail if an event is logged.
- When configured to use an audit server the SGOS appliance transmits audit events to the audit server at the same time logs are written locally.
- If the connection fails, the SGOS continues to store audit records locally and will transmit any stored contents when connectivity to the syslog server is restored.
- The reference identifier for the remote audit server is configured by the administrator using the CLI.
- When the TLS client receives an X.509 certificate from the server, the client will compare the reference identifier with the established Subject Alternative Names (SANs) in the certificate. If a SAN is available and does not match the reference identifier, then the verification fails, and the channel is terminated. If there are no SANs of the correct type (IP address or DNS name) in the certificate, then the TOE will compare the reference identifier to the CN (IP address or DNS name) in the certificate Subject. If there is no CN, then the verification fails, and the channel is terminated. If the CN exists and does not match, then the verification fails, and the channel is terminated. Otherwise, the reference identifier verification passes, and additional verification actions can proceed. The TOE supports wildcards for DNS names in the CN and SAN.
- When the reference identifier is an IP address, the TOE converts the IP address to a binary representation in network byte order. IPv4 addresses are converted directly from decimal to binary, IPv6 addresses are converted as specified in RFC 5952. The TOE compares the binary IP address against all the IP Address entries in the Subject Alternative Name extension. If there is not an exact binary match, then the verification fails.

Warning: The above-mentioned reference identifier matching rules should be taken into consideration while connecting to peers or IT entities using certificates that have DNS or IP Address.

Note: When configuring Syslog monitoring, make sure that the transport protocol (UDP, TCP, or TLS) is enabled on the Syslog loghost server. Symantec recommends that you use TLS for best security instead of UDP or TCP. Before configuring a Syslog host using TLS, create client and server certificates and import them to the respective systems

- It is recommended to set the **logging level to verbose** while in an NDcPP mode of operation.

Event Logging Level	Description
Severe errors	Displays only severe error messages in results.
Configuration events	Displays severe and configuration change error messages in results.
Policy messages	Displays severe, configuration change, and policy event error messages in results.
Informational	Displays severe, configuration change, policy event, and information error messages in results.
Verbose	Displays all error messages in results.

Table 5 - Event Logging Levels

- **To change the syslog event levels, use the following command:**

```
#(config event-log) notifications
#(config event-log notifications) default syslog level {configuration | informational | policy | severe | verbose | trace}
```

- To manage syslog hosts, use the `#(config) event-log` command.
- **To add syslog loghost with UDP or TCP use the following command:**

```
#(config event-log) syslog add [tcp| udp] {host_name | ip_address} [port]
```

Enter the IPv4 or IPv6 address of your loghost server or specify a domain name that resolves to an IPv4 or IPv6 address. If you do not specify a port number, port 514 or 6514 respectively is used by default.

- **To add syslog loghost with TLS use the following command:**

```
#(config event-log) syslog add tls {host_name | ip_address} [port]
[ssl_device_profile_name]
```

Enter the IPv4 or IPv6 address of your loghost server or specify a domain name that resolves to an IPv4 or IPv6 address. If you do not specify a port number, port 6514 is used by default. Specify an existing SSL device profile to secure the appliance's communication with the Syslog server.

- **To removes all Syslog loghosts from system logging notification use the following command:**

```
#(config event-log) syslog clear
```

- **To disable or enable Syslog logging notifications use the following command:**

```
#(config event-log) notifications
#(config event-log notifications) default syslog {enable | disable}
```

- **To specifiy the syslog facility use the following command:**

```
# (config event-log) syslog facility {auth | daemon | kernel | local0 | local1 |
local2 | local3 | local4 | local5 | local6 | local7 | lpr | mail | news | syslog |
user | uucp}
```

- Removes the specified UDP Syslog loghost.

```
#(config event-log) syslog remove udp {host_name | ip_address}
```

- Removes the specified TCP Syslog loghost.

```
# (config event-log) syslog remove tcp {host_name | ip_address}
```

- Removes the specified UDP Syslog loghost.

```
# (config event-log) syslog remove tls {host_name | ip_address}
```

2.8.2 Log Size

- You can limit the size of the appliances' event log and specify what occurs when the log size limit is reached.
- **To set the log-size limit use the following command:**

```
#(config event-log)log-size <megabytes>
```

- Acceptable value is between 50 and 2047.
- Specifies what should happen to the event log when the maximum size has been reached. Overwrite overwrites the oldest information in a FIFO manner; stop disables event logging.

```
#(config event-log) when-full {overwrite | stop}
```

2.8.3 Local Storage Full

- You can make the logging buffer full using the following command:

```
#.event-log testfill <number>
```
(number of entries to be written)

2.8.4 Local Storage Reset

- You can reset the entire logging buffer full using the following command:

```
#.event-log reset
```

2.9 Managing X.509 Certificates

The following steps describe how the validity of certificates is checked in ProxySG:

- ProxySG checks the certificate chain to ensure that it is complete and valid. This involves verifying that the certificate was issued by a trusted root CA and that all intermediate certificates in the chain are valid and signed by a trusted CA.
- ProxySG checks the revocation status of the certificate. This involves verifying that the certificate has not been revoked by the issuing CA or the web server owner.
- ProxySG checks the hostname in the TLS certificate matches the hostname set on the TOE.
- ProxySG can also perform additional checks, such as verifying that the TLS version and cipher suite used are secure and in compliance with organizational policies.

- If any of these checks fail, ProxySG may either block the TLS connection or present the user with a warning message. This helps to ensure that the TLS connections are secure and that users are protected from potential security threats.

2.10 Importing CA certificate to the Device

- The appliance is preinstalled with and trusts all root CA certificates trusted by Internet Explorer and Firefox. This certificate list is updated periodically to be in sync with the latest versions of IE and Firefox.
- You can also import non-standard third-party CA certificates into the appliance CA certificate store, including root and intermediate CA certificates.
- By adding CA certificates to the CA certificate store, these will be available for use by the CA certificate lists (CCL) for validating the security of connections.
- The hostname is extracted from the X.509 certificate returned by the server while establishing an TLS connection.

To import the CA certificate to the appliance using CLI:

1. Install the loghost certificate to the appliance, as in the following example:
   ```
   #(config ssl)inline ca-certificate rsyslog_ca certificate_contents EOF
   ```
   where **rsyslog_ca** is the name of the Syslog certificate
2. Create a CA Certificate List (CCL) (Refer Section 2.7.1 for CCL details) for the certificate:
   ```
   #(config ssl)create ccl rsyslog_ccl
   ```
   where **rsyslog_ccl** is the name of the CCL
3. Add the certificate to the CCL:
   ```
   #(config ssl)edit ccl rsyslog_ccl
   #(config ssl ccl rsyslog_ccl)add rsyslog_ca
   ```
4. Create an SSL device profile:
   ```
   #(config ssl)create ssl-device-profile rsyslog
   ```
   where **rsyslog** is the name of the device profile
5. The device profile you create here is the one you reference in the
   ```
   #(config event-log) syslog add tls
   ```
   command, as in the following example:
   ```
   #(config event-log) syslog add tls company.com rsyslog
   ```
6. Specify the CCL for the device profile:
   ```
   #(config device-profile rsyslog)ccl rsyslog_ccl
   ```

2.10.1 CA Certificate List (CCL)

A CA certificate list (CCL), which contains some of the CA Certificates available on the appliance, allows the administrator to control the set of CA certificates trusted for a particular set of SSL connections. A CCL contains a subset of the available CA certificates on the appliance and restricts trust to those certificates. The CCL referenced by the profile or service configuration is used when an SSL connection is established to that service or using that profile.

Three CCLs are created by default on the appliance:

1. `appliance-ccl`: This CCL is used for authenticating connections among devices manufactured by Symantec. By default, it contains the Symantec ABRCA root certificate (ABRCA_root). This list is used by default in the **bluecoat-appliance-certificate** SSL device profile. This CCL can be edited but not deleted.
2. `browser-trusted-fips`: This CCL includes most of the well-known CAs trusted by common browsers. This CCL can be edited but not deleted. You can manually add CAs to this list. In addition, the appliance automatically retrieves an updated browser-trusted CCL from Symantec every seven days.The browser-trusted-fips CCL is used by default during certificate verification by the SSL client and by the **default** SSL device profile.
3. `image-validation`: This CCL is used to validate signed SGOS images. You can customize the CCLs available on the appliance to ensure that the appliance has the CA certificates it needs to handle HTTPS requests. You can create your own CA certificate lists or modify the default CCLs by adding or removing trusted CAs.

To create and delete the CA certificate list (CCL) to the appliance using CLI Console:

- To create the CA certificate list (CCL) to the appliance using CLI Console use the following commands:

```
#(config ssl)
#(config ssl) create fips ccl <list_name>
```

- To delete a CCL list from the ProxySG appliance use the following command:

```
#(config ssl) delete ccl list_name
```

To update the CA certificate list (CCL) to the appliance using CLI Console:

- To add the CA certificate to CCL use the following commands:

```
#(config) ssl
#(config ssl) edit ccl list_name
#(config ssl ccl list_name) add ca_certificate_name
```

  Adds a CA certificate to this list. (The CA certificate must first be imported in configure ssl mode.)

- To delete the CA certificate to CCL use the following commands:

```
#(config ssl ccl list_name) remove ca_certificate_name
```

  Removes a CA certificate from the specified list.

- To view the CA certificate to CCL use th e following commands:

```
#(config ssl ccl list_name) view
```

  Shows a summary of CA certificates in this list.

2.10.2 Creating a Keyring

Keyrings are virtual containers. Each keyring holds a public/private key pair and a customized key length. You can associate certificates, certificate chains or certificate signing requests with keyrings. A default keyring is shipped with the system and is used for accessing the Management Console, although you can use others. You can also use the default keyring for other purposes. You can create other keyrings for each SSL service.
The appliance ships with several keyrings already created:

1. default: The default keyring contains a certificate and an automatically generated keyring and a self signed certificate which can be used for accessing the appliance through HTTPS. As demonstrated by the appliance Management Console.
2. configuration-passwords-key: The configuration-passwords-key keyring contains a keypair but does not contain a certificate. This keyring is used to encrypt passwords in the show config command and should not be used for other purposes.
3. appliance-key: The appliance-key keyring contains an internally generated keypair. If the appliance is authenticated (has obtained a certificate from the Symantec CA appliance-certificate server), that certificate is associated with this keyring, which is used to authenticate the device.
4. passive-attack-protection-only-key: The passive-attack-protection-only-key keyring allows data to be encrypted, but with no endpoint authentication. Although the traffic cannot be sniffed, it can be intercepted with a man-in-the-middle attack. The passive-attack-protection-only-key keyring is NOT considered secure; therefore, it should not be used on production networks.

To create a keyring using CLI Console:

- To create a fips keyring use the following commands:

```
#(config ssl) create fips keyring {show | show-director | no-show} keyring_id
[key_length]
```

Creates keyring, where:
`show`: Private keys associated with keyrings created with this attribute can be displayed in the CLI or included as part of a profile or overlay pushed by Director.
`show-director`: Keyrings created with this attribute are part of the `show configuration` output if the CLI connection is secure (SSH/RSA) and the command is issued from Director.
`no-show`: Keyrings created with this attribute are not displayed in the `show configuration` output and cannot be part of a profile. The `no-show` option is provided as additional security for environments where the keys will never be used outside of the particular ProxySG appliance.

To delete a keyring use the following command:

```
#(config ssl) delete keyring keyring_id
```

Deletes a keyring, with a keypair.

To view a keyring use the following command:

```
#(config ssl) view keyring [keyring_id | unreferenced | expiring-in <n>]
```

Displays the keyring, where:
`keyring_id`: Displays the certificate subject, serial number, issuer, and all keylists that the specified keyring is a member of.
`unreferenced`: Lists all the keyrings that are not referenced anywhere else in the configuration or in policy.
`expiring-in <n>`: Lists all keyrings with certificates expiring in a specified <n> days. To display all keyrings with expired certificates, use the following command:

```
#(config ssl) view keyring expiring-in 0
```

`<Enter>`: Show all keyrings

```
#(config ssl) view keyring [keyring_id]
```

Displays the keyring.

2.11 Intermediate Certificate Cache

- The appliance automatically stores unrecognized intermediate CA certificates that are included with validated CA certificate chains whenever an SSL connection is established.
- These intermediate CA certificates are stored within a separate cache on the appliance and are used to validate SSL connections when an incomplete certificate chain is encountered.
- For security purposes, OCSP and CRL validation checks are performed to confirm the safety of the certificate chain.
- As an additional layer of security, the intermediate CA certificates in the chain must end with a trusted root certificate from the CCL (CA certificate list) that is associated with the connection.
- If a compatible certificate is not found, the connection is considered insecure, and the user will be given a security warning.

2.11.1 Enable Caching

To enable Intermediate Certificate Caching use the following commands:

```
#(config ssl) intermediate-cert-cache
```

This changes the prompt to:

```
#(config ssl icc)
#(config ssl icc) enable
```

Enables the caching of intermediate CA certificates on the ProxySG appliance.

2.11.2 Turn off Caching

To turn off Intermediate Certificate Caching use the following commands:

```
#(config ssl icc) disable
```

Simultaneously disables the caching of intermediate CA certificates and clears the existing cache on the ProxySG appliance.

2.11.3 View Cached Certificates

To view Intermediate Certificate Caches, use the following commands:

```
#(config ssl icc) view status
```

Displays the current status of the intermediate certificate cache, including usage statistics and the number of stored intermediate CA certificates.

```
#(config ssl icc) view certificate {detail certificate_name | summary | summary certificate_name}
```

You can view various details about the certificates that have been cached on the appliance.

2.11.4 Clear Cached Certificates

Clearing the CA certificate cache removes all stored intermediate CA certificates.

To clear Intermediate Certificate Cached use the following commands:

```
#(config ssl icc) clear-cache
```

Clears the intermediate CA certificates that are currently stored on the appliance.

```
#(config ssl icc) exit
```

Exits the `config ssl icc` prompt and returns to the `config ssl` prompt.

Note: The appliance retains the list of cached intermediate CA certificates even after the appliance is shut down and restarted. The only way to delete the cache is to manually clear or turn off certificate caching.

2.12 Certificate Signing Request (CSR)

- Certificate signing requests (CSRs) are used to obtain a certificate signed by a Certificate Authority. You can also create CSRs off box.

2.12.1 Creating a CSR

To create a CSR:

```
#(config ssl) create signing-request <keyring_id> [<attribute> <value>]+
```

Creates a certificate signing request (CSR). The CLI prompts you to enter values for the following attributes:

- two-digit ISO country code
- two-letter state/province abbreviation
- city name/locality
- organization name
- organization unit
- common name
- email address
- challenge password
- company name
- digest type

Press ENTER to specify no value. Default values are in square brackets [].

Notes:

- You must associate the CSR with a keyring and a digest.
- You can create a CSR in one of two ways: interactively or non-interactively.
- The default digest is SHA256.
- Director uses non-interactive commands in profiles and overlays to create CSRs.

2.12.2 Viewing a CSR

After a CSR is created, you must submit it to a CA in the format the CA requires. You can view the output of a certificate signing request.

To view the output of a certificate signing request:

```
#(config ssl) view signing-request <keyring_id>
```

Displays the certificate signing request.

2.12.3 Deleting a CSR

To delete a CSR:

```
#(config ssl)delete signing-request <keyring id>
```

2.12.4 Uploading CSR on TOE

The following command can be used to load the CSR on the TOE:

```
#(config ssl)inline certificate <keyring id> [<ccl>|<"">] <eof marker>
e.g. #(config ssl)inline certificate test browser-trusted-fips end-xxx
```

2.13 SSL Device Profile

- An SSL device profile only needs to be created if you cannot use the built-in **bluecoat-appliance-certificate profile** without modification; note that the **bluecoat-appliance-certificate profile** cannot be deleted or edited.
- If you require different cipher suites than those provided by the **bluecoat-appliance-certificate profile**, you can create a new profile to meet your specific cipher suite requirements.
- The already-created SSL device profiles and their purposes are:
    - **bluecoat-appliance-certificate:** This profile, which cannot be edited or deleted, is used for device-to-device authentication, allowing Symantec devices on a network to identify other Symantec devices that can be trusted. You can select this device profile when setting up device authentication, or you can create a new device profile as described in Creating an SSL Device Profile for Device Authentication.
    - **passive-attack-detection-only:** This profile, which cannot be edited or deleted, optionally can be used in place of the **bluecoat-appliance-certificate profile**. The **passive-attack-detection-only profile** uses a self-signed certificate and disables the verify-peer option, so that no authentication is done on the endpoints of the connection. The traffic is encrypted but is vulnerable to active attacks.
    - **default:** This profile can be edited but not deleted. Only secured non-proxy traffic uses this profile.
- You can edit the existing default SSL device profile for the environment and create additional SSL device profiles with different settings.
- For example, if you require a different cipher setting from what the default profile uses, create a profile with the different cipher suite.
- An SSL device profile contains the information required for device authentication:

1. The name of the keyring that contains the private key and certificate this device uses to authenticate itself. The default keyring is appliance-key.
2. The name of the CA Certificate List (CCL) that contains the names of certificates of CAs trusted by this profile. If another device offers a valid certificate signed by an authority in this list, the certificate is accepted. The default is `appliance-ccl`.
3. Verification of the peer certificate.
    - When the appliance is participating in device authentication as an SSL client, the peer certificate verification option controls whether the server certificate is validated against the CCL. If verification is disabled, the CCL is ignored.
    - When the appliance is participating in device authentication as an SSL server, the peer certificate verification option controls whether to require a client certificate. If verification is disabled, no client certificate is obtained during the SSL handshake. The default is `verify-peer-certificate enabled`.
4. Specification of how the device ID authorization data is extracted from the certificate. The default is `$(subject.CN )`.
5. SSL cipher settings. The default is SHA256.

- Each Symantec appliance has an automatically constructed profile called **bluecoat-appliance-certificate** that can be used for device-to-device authentication. This profile cannot be deleted or edited.
- If you cannot use the built-in profile because, for example, you require a different cipher suite or you are using your own appliance certificates, you must create a different profile, and have that profile reference the keyring that contains your certificate.
- The TLS client will transmit the Supported Elliptic Curves extension in the Client Hello message by default with support for the following NIST curves: P-256, P-384, and P-521.

- The non-TOE server can choose to negotiate the elliptic curve from this set for any of the mutually negotiable elliptic curve Ciphersuites.

Note:

- If you do not want to use peer verification, you can use the built-in **passive-attack-detection-only** profile in place of the **bluecoat-appliance-certificate** profile.
- This profile uses a self-signed certificate and disables the `verify-peer` option, so that no authentication is done on the endpoints of the connection.
- The traffic is encrypted but is vulnerable to active attacks. This profile can be used only when there is no threat of an active man-in-the-middle attack.
- Like the **bluecoat-appliance certificate** profile, the **passive-attack-detection-only** profile cannot be edited or deleted**.**
- If you create your own profile, it must contain the same kind of information that is contained in the Symantec profile.
- Non-proxy traffic uses an SSL device profile. Proxy traffic uses the SSL client profile.

2.13.1 Creating an SSL Device Profile

To create your own profile, use the following commands:

```
#(config ssl)create fips ssl-device-profile <SSL device profile name>
```

2.13.2 Editing an SSL Device Profile

To edit ssl-device-profile use the following commands:

```
#(config ssl)edit ssl-device-profile profile_name
```

This changes the prompt to:

```
#(config device-profile profile_name)
```

Subcommands:

1. **Ciphersuites configuration:**

```
#(config device-profile profile_name) cipher-suite cipher-suite
```

Configures device authentication profile cipher suites. If you press `<enter>`, you can see the list of available ciphers.

The default is to use all cipher suites. If you want to change the default, you have two choices:

- interactive mode
- non-interactive mode

Director uses non-interactive commands in profiles and overlays to create cipher suites.

The optional `cipher-suite` refers to the cipher-suites you want to use, space separated, such `as ECDHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-SHA`. If you want to use the interactive mode, do not specify a cipher suite. You may specify more than one cipher suite.

2. **CCL configuration:**

```
#(config device-profile profile_name) ccl ccl_name
```

Configures the device authentication profile CCL.

3. **Device ID configuration:**

```
#(config device-profile profile_name) device-id device_ID
```

Configure device authentication profile of the specific device ID.

4. **Keyring configuration:**

```
#(config device-profile profile_name) keyring-id keyring_ID
```

Configures the device authentication profile in the specified keyring.

```
#(config device-profile profile_name) no keyring-id keyring_ID
```

Clears the SSL device profile keyring ID.

5. **Protocol configuration:**

```
#(config device-profile profile_name) protocol {tlsv1 | tlsv1.1 | tlsv1.2}
```

Specifies the protocol or protocols to use.

6. **Peer verification configuration:**

```
#(config device-profile profile_name) verify-peer {enable | disable}
```

Enables or disables device authentication peer verification.

```
#(config device-profile profile_name) view
```

To view the SSL-device-profile configuration

```
#(config device-profile profile_name) exit
```

Returns to the `# (config ssl)` prompt.

2.14 Configuring OCSP

- OCSP (RFC 2560) allows you to obtain the revocation status of an X.509 digital certificate. OCSP provides the same revocation functionality as the local Certificate Revocation List (CRL) configured on the appliance.
- Managing large CRLs poses scalability challenges. This is due to high memory consumption on the appliance associated with storing revocation lists.
- OCSP overcomes these limitations by checking certificate status in real time using off-box OCSP responders.
- OCSP certificates presented for OCSP responses must have the 'ocspSigning' extendedKeyUsage purpose.
- Certificate revocation checking for the above scenarios is performed by querying with an OCSP Responder.
- OCSP-based revocation checks are performed on server certificates.
- In this section, this server certificates are referred to as subject certificates.
- The TOE acts as an OCSP client and sends OCSP queries to an OCSP responder for the given certificate.
- An OCSP responder is a server for OCSP request processing and response building functions.
- The OCSP responder sends status of the certificate back to the TOE (OCSP client). Status can be good, revoked, or unknown.
- *Good* means that the certificate is not revoked and valid at the time of the query. *Revoked* means that the certificate has been revoked either permanently or temporarily. *Unknown* means that the responder does not know about the revocation status of the certificate being requested.
- The appliance can also cache OCSP responses and can respect, override or ignore the timestamps related to cache ability in the OCSP response.
- If the certificate status is valid, the TLS connection is successful. If the status is revoked, an error is flagged, and the TOE denies access to the server. If status is unknown, the appliance can treat it as an error or ignore it based on the administrator's discretion.
- To configure OCSP use the following commands:

```
#(config) ssl
```

This changes the prompt to:

```
#(config ssl)
#(config ssl)ocsp
```

This changes the prompt to:

```
#(config ssl ocsp)
```

- **To create a OCSP responder use the following command:**

```
#(config ssl ocsp)create responder_name
```

Creates a responder.

- **To make a OCSP responder as default use the following command:**

```
#(config ssl ocsp)default responder_name
```

Sets a responder to the default responder.

- **To delete a OCSP responder use the following command:**

`#(config ssl ocsp)` **delete** *responder_name*

Deletes the specified responder.

`#(config ssl ocsp)` **exit**

Exits the config ssl ocsp prompt and returns to the config ssl prompt.

- **To edit a OCSP responder use the following command:**

`#(config ssl ocsp)` **edit** *responder_name*

Configure this *responder_name*. Changes the prompt to: `#(config ocsp` *responder_name*`)`

2.14.1 Subcommands

`#(config ocsp responder_name)` **exit**

Exits the `config ocsp` *responder_name* prompt and returns to the `config ssl ocsp` prompt.

`#(config ocsp responder_name)` **extension nonce {disable | enable}**

Enables or disables use of a nonce control in an OCSP request. When enabled, a nonce (unique digits sequence) is included as one of the requestExtensions in each OCSP request. Default is disable.

`#(config ocsp responder_name)` **extension request-signing-keyring** *<keyring-id>*

Configures the OCSP request to contain a signature along with certificates to help the OCSP responder verify this signature. The keyring must already exist and have a certificate.

`#(config ocsp responder_name)` **ignore expired-responder {enable | disable}**

Specifies whether the OCSP request must contain a signature along with certificates to help the OCSP responder verify this signature. The keyring must already exist and ave a certificate. By default, invalid responder certificate dates cause the subject certificate verification to fail.

`#(config ocsp responder_name)` **ignore ocsp-signing-purpose {enable | disable}**

Specifies whether to ignore the enforcement of purpose field in the responder certificate. Default is enable.

`#(config ocsp responder_name)` **ignore request-failure {enable | disable}**

Specifies whether to ignore connection failures and timeouts to the OCSP server. Default is disable.

`#(config ocsp responder_name)` **ignore unknown-status {enable | disable}**

Specifies whether to treat “unknown” revocation status for a certificate as an error. By default, unknown status is an error and causes subject certification verification to fail.

`#(config ocsp responder_name)` **ignore untrusted-responder {enable | disable}**

Specifies whether to bypass, during responder certificate verification, any untrusted certificate errors. For example, a missing issuer certificate or a missing self-signed certificate. By default, any untrusted certificate failure is an error and causes the subject certificate verification to fail.

`#(config ocsp responder_name)` **issuer-ccl** {*CCL Name | all | none*}

Sets the name of the CCL. This is the list of CA names which is associated with the certificate to be checked for revocation. It may either be a server or client certificate, or a certificate that is used for verifying system images.

`#(config ocsp responder_name)` **no extension request-signing-keyring**

Resets the request signing keyring.

`#(config ocsp responder_name)` **response-ccl** {*Response CCL Name | all*}

Sets the name of the CCL.

`#(config ocsp responder_name)` **ssl-device-profile** *SSL device-profile name*

Sets the SSL device profile. The device profile is a unique set of SSL cipher-suites, protocols and keyrings used when the ProxySG appliance makes HTTPS connections with an OCSP responder. The default value is the pre-created device profile named “default”.

```
#(config ocsp responder_name)ttl {auto | number_ of_days}
```

Configures the time to live (TTL) value. This value determines how long a response remains in the cache. The auto option indicates that the response is cached until nextUpdate. If nextUpdate is not present the response is not cached. The number_ of_days variable indicates that the nextUpdate field in the response is to be overridden and that the response is to be cached for the indicated number of days. Default is auto.

```
#(config ocsp responder_name)url <OCSP server url> | from-certificate
```

URL Indicates the location of the OCSP responder. The appliance needs this URL to locate the responder. This location can be obtained from the certificate’s Authority Information Access (AIA) extension or from a user-defined configuration. The default is to use the URL from the certificate.
Use URL from certificate—Select this option if you want the appliance to look up the OCSP server location from the subject certificate’s AIA extension.
Use URL—Select this option if the location of the designated OCSP responder is known to you. Enter a specific responder HTTP or HTTPS URL.

```
#(config ocsp responder_name) use-forwarding {disable | enable}
```

Sets the OCSP requests to use forwarding.

```
#(config ocsp responder_name) view
```

Displays the responder configurations.

2.15 Software Status and Upgrade

When the user has logged into the TOE through the CLI Console, the software version can be seen with the command line.

2.15.1 Checking the Software Version

- **To check the Software version, use the following command:**

```
> show version
```

Displays ProxySG appliance hardware and software version and release information and backplane PIC status.

```
10.1.5.121 - Edge SWG#show version
Version: SGOS 7.4.1.1 SWG Edition
Release id: 287291 64-bit, gdb, unoptimized
Serial number: 0070990142
Appliance identifier: d59dc1f242f06c59
NIC 0 MAC: 00D083D00241
System is in FIPS mode; cryptographic module algorithm version: 5.1.1
```

Figure 3 - TOE Version

- To install the SGOS software release, you must obtain the image, and then install it on your appliance. You must reboot after the installing the release. You can upgrade directly to 7.4.x, download.

- The TOE is the Symantec Edge SWG running SGOS software version 7.4. The Symantec Edge SWG is not tied to any specific hardware.

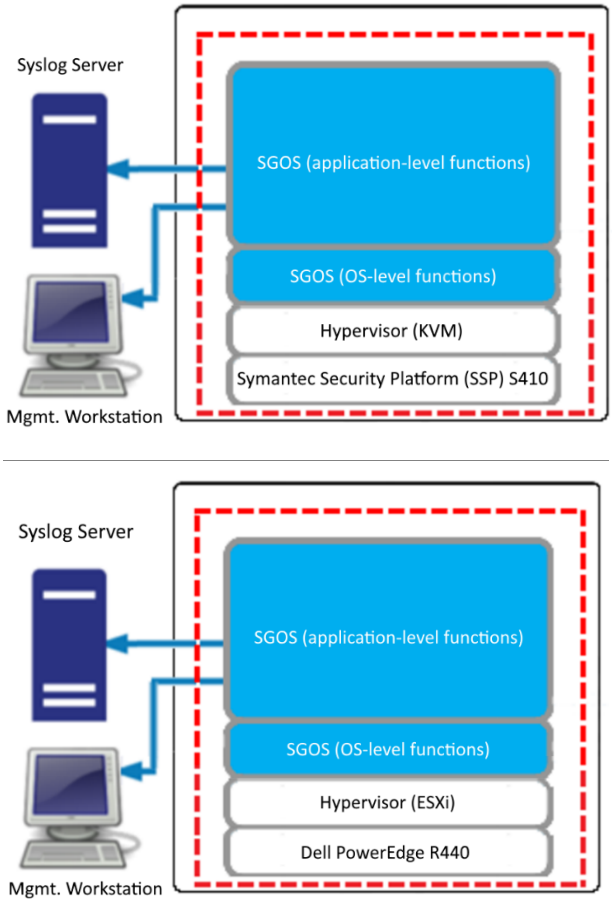

Figure 4 - TOE Boundaries

- The Integrated Secure Gateway (ISG) is the software on the Symantec Security Platform (SSP) appliance used to deploy applications.
- Use the ISG command line interface (CLI) to perform the following tasks:
  - Connect the SSP appliance to your network
  - Connect to the ISG serial console
  - Create and run one or more applications
  - License applications
- The SSP is not a licensed product and only the applications it runs require licenses.
- The administrator downloads the proxysg_7.4.X.X-######.bcsi file and makes note of the published hashes (SHA-256 and MD5). They then must, using a local tool of their own, (e.g., hash tool) compute the SHA-256 hash of the .bcsi file. Once the output from the hash tool is computed, they can then visually verify that the 2 hashes are the same. If they differ, then the downloaded file is not valid and should not be used to upgrade.
- Builds can be downloaded from: https://support.broadcom.com/group/ecx/downloads?. You'd have to be a customer with a provisioned account to login to the URL.
- Every build (.bcsi file outlined in red box) posted on our secure downloads portal will have both an MD5 and SHA-2 hash published with it.

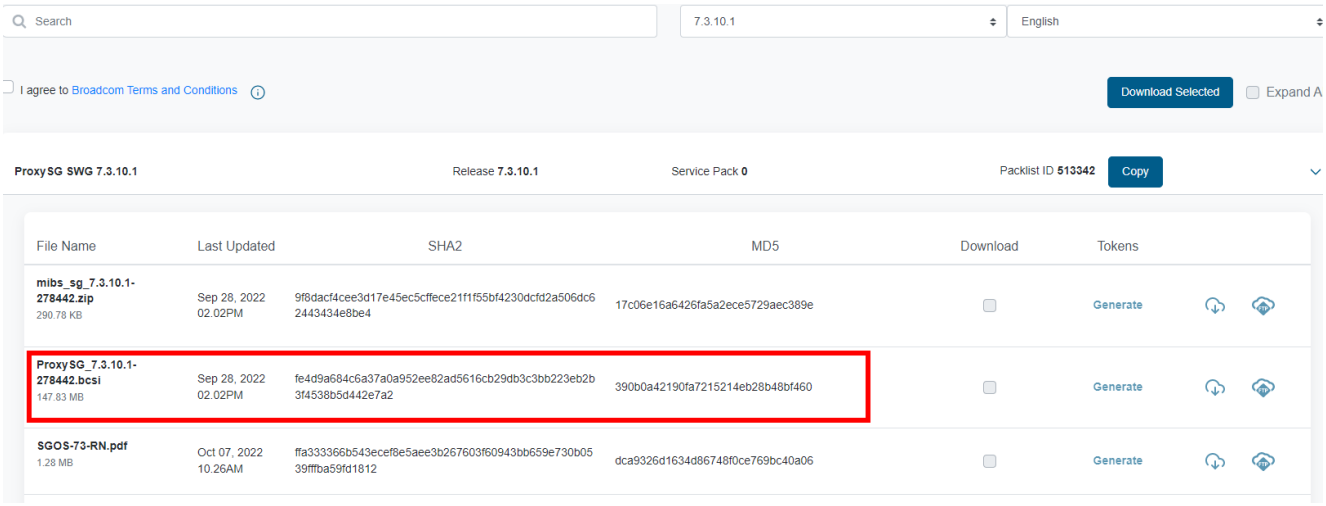

Figure 5 - Example of published hash on Portal

2.15.2 Upgrading the SGOS on SSP Hardware

- To upgrade the SGOS on SSP hardware we need to follow the following steps:

1. **Install an Application Image on ISG:**

- Before you create and start an application, load the application image onto the ISG. ISG is the platform on which an application runs.
  - From the appliance serial console, enter configuration mode:
    ```
    # config
    ```
  - Load the application image:
    ```
    (config)# images
    (config-images)# load <application_location_URL>
    ```

2. **Create Applications on ISG:**

- Create the application with the following commands:
  ```
  (config)# applications
  (config-applications)# create sg sg_name model model_name license-id license_id
  image-id image_id
  ok
  ```

3. **Stop and Start the application:**

- Stop the previous running application (if any) and start the newly created application with the following commands:
  ```
  (config-applications)# stop application_name
  (config-applications)# start application_name
  ```

4. **View Application Information:**

- To view application information, such as license IDs, image IDs, and other properties that are associated with your applications, use the applications view command (in either enable or configuration mode). For example:
  ```
  (config-applications)# view
  NAME TYPE VCPU MEMORY MODEL STATUS LICENSE ID IMAGE ID
  ------------------------------------------------------------------------------
  SG1 SG 2 20 GB C2L Running 000090007 sg-6.7.5.6-252532
  SG2 SG 2 20 GB C2L Running 000090007 sg-6.7.5.6-252532
  SG3 SG 2 20 GB C2L Running 000090007 sg-6.7.5.6-252532

  (config-applications)# view SG1
  NAME TYPE VCPU MEMORY MODEL STATUS LICENSE ID IMAGE ID
  ------------------------------------------------------------------------------
  SG1 SG 2 20 GB C2L Running 000090007 sg-6.7.5.6-252532
  ```

5. **Connect to the Application Serial Console:**

- From an application serial console, you can access the application's command line to perform tasks, such as initial configuration.

```
(config-applications)# attach-console <application_name>
```

- The following is an example output of the command:

```
(config-applications)# attach-console SG1
Connected to domain sgos
Escape character is ^]
System starting up...
In MP mode; two processors active
Executing image: Version: SGOS 6.7.5.3, Release id: 249936 64-bit, gdb, optimized
Manufacturing MBR on directory-3 - Slot 3 (KVM VirtIO Disk N/A N/A)
This is a new system.
Press "enter" three times to activate the serial console
****************** CONFIGURATION ALERT ******************
System entering configuration wizard for the following reasons:
- Cannot find a network adapter configured with an IP address and subnet.
- The console password or 'enable' password is not set.
****************** CONFIGURATION ALERT ******************
--------------- CONFIGURATION START ------------------
Welcome to the Blue Coat SG-VA Series configuration wizard.
This appliance's serial number: 0000990007
-------------------------------------------------------------------
You can get field help by entering a question mark ? in the fields.
You can move backwards through the steps by pressing the UP arrow.
You can exit the wizard without saving your entries by pressing ESC.
-------------------------------------------------------------------
Step 1: How do you plan to configure this appliance?
a) Through a manual setup
b) Through a Director-managed setup
Your choice: []
```

- There is no delayed activation of the software version. When an application is created for a particular software version, and that application is started, that software image is activated.
- To enable FIPS mode use the following command:

```
#fips-mode enable
```

To edit applications use the following command:

- Stop the application that you want to edit:

```
(config-applications)# stop <application_name>
```

NOTE: To edit an existing application, your application must be in a Created or Stopped state.

- Edit the application:

```
(config-applications)# edit <application_name model_type | image-id image_id>
```

- The following example shows how to view the application configuration, stop the application, and change the model from a C2L to a C2S:

```
(config-applications) view SG1
NAME TYPE VCPU MEMORY MODEL STATUS LICENSE ID IMAGE ID
-------------------------------------------------------------------------
SG1 SG 2 20 GB C2L Running 000090007 sg-6.7.5.6-252532
(config-applications)# stop SG1
ok
(config-applications)# edit SG1 model C2S
Ok
```

To remove applications use the following command:

```
(config-applications)# delete <application_name>
```

To view image information use the following command:

- View all downloaded images:

```
(config-images)# view
```

- View a specific image:
  `(config-images)# view <image_id>`
- – View all ProxySG images:
  `(config-images)# view sg`

To remove image use the following command:

`(config-images)# delete <image_id>`

2.15.3 Upgrading the SGOS on EXSi Hardware

- To upgrade the SGOS on EXSi hardware we need to follow the following steps:

1. **Install an Image on SGOS:**

- The first step is to load the application image onto the SGOS. Here SGOS runs on the EXSi platform.
  - From the SGOS serial console, enter configuration mode:
    `# configure terminal`
  - Load the SGOS image:
    `(config)# images`
    `(config-images)# upgrade-path <application_location_URL>`

2. **Download new system image:**

- To download and install the new installed image use the following command:
  `#load upgrade <Enter> | ignore-warnings` Ignore any upgrade warnings

3. **Restart the SGOS with newly insalled image:**

- To restart the new installed image, use the following command:
  `#restart upgrade <Enter> | keep-sgos7-config` Preserve existing 7.x configuration on upgrade
- There is no delayed activation of the software version. When the restart upgrade command is initiated, the TOE is rebooted with the freshly loaded image.

4. **Enable FIPS mode:**

- To enable FIPS mode use the following command:
  `#fips-mode enable`

The TOE uses public hash to verify the integrity of the update. If the computed hash matches the published hash the image will be installed, or else unsuccessful message will be delivered to the user.

2.15.4 Restoring System Defaults

You can restore some or all of the system defaults. Use these commands with caution. The `restore-defaults` command deletes most, but not all, system defaults:

- The `restore-defaults` command with the `factory-defaults` option reinitializes the appliance to the original settings it had when it was shipped from the factory. You must use the CLI to perform this action.
- The `restore-defaults` command with the `keep-console` option restores the default settings without losing all IP addresses on the system. This action is available in the Management Console and the CLI.
- The following sections describe the three possible operations:

1. Restore-Defaults:

   Settings that are deleted when you use the restore-defaults command include:
   - All IP addresses (these must be restored before you can access the Management Console again).
   - DNS server addresses (these must be restored through the CLI before you can access the Management Console
   - again).
   - Installable lists.
   - All customized configurations.
   - Symantec trusted certificates.
   - Original SSH (v1 and v2) host keys (new host keys are regenerated).

   You can use the `force` option to restore defaults without confirmation.

2. Keep-Console:

Settings that are retained when you use the restore-defaults command with the keep-console option include:

- IP interface settings, including VLAN configuration.
- Default gateway and static routing configuration.
- Virtual IP address configuration.
- Bridging settings.
- Failover group settings.

Using the keep-console option retains the settings for all consoles (Telnet, SSH, HTTP, and HTTPS), whether they are enabled, disabled, or deleted. Administrative access settings retained using the restore-defaults command with the keep-console option include:

- Console username and password.
- Front panel pin number.
- Console enable password.
- SSH (v1 and v2) host keys.
- Keyrings used by secure console services.
- RIP configurations.

You can also use the `force` option to restore defaults without confirmation.

- **To perform a restore-default keep-console action using the CLI:**
  `#restore-defaults keep-console`

3. Factory-Defaults:

All system settings are deleted when you use the restore-defaults command with the factory-defaults option. The only settings that are retained are:

- Trial period information
- The last five installed appliance systems, from which you can pick one for rebooting

The Serial Console password is also deleted if you use `restore-defaults factory-defaults`.

You can also use the `force` option to restore defaults without confirmation.

- **To restore the system to the factory defaults using the CLI enter the following command:**
  `#restore-defaults factory-default`

2.16 Username and Passwords

- The user will use these instructions to properly configure the username and password to operate in a compliant manner.
- A compliant password must be at least 8 characters long with the following complexity:
  - At least one uppercase letter
  - At least one lowercase letter
  - At least one numbers
  - At least one of the following special characters:
  - [“!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, “)”, [“”, “+”, “-“, “=”, “.”, “/”, “\”, “:”, “;”, “<”, “>”, “[“, “]”, “_”, “{“, “}”, “|”, “~” “`”]]
  - Administrators are trusted to ensure passwords/credentials have sufficient strength and entropy and to lack malicious intent when administering the device.
- During the initial configuration of your ProxySG appliance, a console administrator username and password was created.
- This is a special account that can always be used to administer the appliance from Command Line Interface.
- You can change the username and the password of this administrator account.
- The console password and privileged-mode password were defined during initial configuration of the system.
- The console password can be changed at any time.
- The privileged-mode, or enabled-mode, password can only be changed through the CLI or the serial console.
- For better security, the appliance hashes and encrypts passwords for various accounts and services.
- The appliance hashes passwords used for authentication on the appliance itself. It is not possible to reverse the hash to recover the cleartext passwords.

2.16.1 Setting Password Length and Complexity

- **To define the password length, use the following command:**

```
#(config) security password-min-len <length>
```

Acceptable values are between 8 and 64 when device in FIPS-mode.

- **To define the password complexity, use the following commands:**

```
#(config)security password-policy min-digits <1-64>
```

Set the minimum number of digits in passwords

```
#(config)security password-policy min-lowercase <1-64>
```

Set the minimum number of lowercase letters in passwords

```
#(config)security password-policy min-special <1-64>
```

Set the minimum number of special characters in passwords

```
#(config)security password-policy min-uppercase <1-64>
```

Set the minimum number of uppercase letters in passwords

```
#(config)security password-policy prohibit-common-words
```

Reject passwords matching common words

```
#(config)security password-policy prohibit-whitespace
```

Reject passwords containing whitespace

```
#show security password-policy
```

See the set Password policy parameters

2.16.2 Setting the Console Username

- **To set the console username use the following command:**

```
#(config) security username <name>
```

- Example:

```
#(config) security username Test
```

2.16.3 Setting the Console Password

- **To set the console password use the following command:**

```
#(config) security password <Enter> | <password>
```

- Example:

```
#(config)security password
Enter password: ********
Confirm password: ********
ok
```

- To set the encrypted console password use the following command:

```
#(config) security encrypted-password <encrypted password>
```

- To set the hashed console password use the following command:

```
#(config) security hashed-password <hashed password>
```

2.16.4 Setting the Enable Password

- To set the enable password use the following command:

```
#(config) security enable-password <Enter> | <password>
```

- Example:

```
#(config)security enable-password
Enter password: ********
Confirm password: ********
ok
```

- To set the encrypted enable password use the following command:

```
#(config) security encrypted-enable-password <encrypted password>
```

- To set the hashed enable password use the following command:

```
#(config) security hashed-enable-password <hashed password>
```

2.17 User Roles

- The Administrator has the ability to control all aspects of the TOE access configuration including: user management, information flow policy management, audit management and system start-up and shutdown.
- When the Administrator adds a new user, the Administrator defines the user role.
- SGOS supports different levels of command security:
  - Standard, or unprivileged, mode is read-only. You can see but not change system settings and configurations. This is the level you enter when you first access the CLI.
  - Enabled, or privileged, mode is read-write. You can make immediate but not permanent changes to the ProxySG appliance, such as restarting the system. This is the level you enter when you first access the Management Console.
- Configuration mode allows you to make permanent changes to the ProxySG appliance configuration. To access Configuration mode, you must be in Enabled mode.
- The Security Administrators (a.k.a Authorized Administrators) user can connect to the TOE from a protected network via remote CLI over SSHv2 or at the local console to perform these functions.
- The specific management capabilities available from the TOE include:
  - Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI, as described above,
  - Ability to configure the access banner,
  - Ability to configure the session inactivity time before session termination or locking,
  - Ability to update the TOE, and to verify the updates using hash comparison capability prior to installing those updates,
  - Ability to configure the authentication failure parameters;
  - Ability to configure audit behavior, in particularly, changes to the size of the audit space, ;
  - Ability to configure the cryptographic functionality. The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide protection of the data, such as generating RSA keys;
  - Ability to re-enable an Administrator account;
  - Ability to configure NTP for time;
  - Ability to configure the reference identifier for the peer (SAN-IP address and SAN-DNS hostname);
  - Import and delete X.509v3 certificates;
  - Generate and delete cryptographic keys. A security administrator can generate and delete the cryptographic keys associated with CSRs.

Management Functions	Interface	User Roles	
		Read Only	Read/Wirte
System Time and time zones	Local Console/Remote CLI i.e. SSH	View	View/Modify
Configuration- system configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
License	Local Console/Remote CLI i.e. SSH	View	View/Modify
Certificate Trust store (CCL- CA certificate lists)	Local Console/Remote CLI i.e. SSH	View	View/Modify
CA certificate configuration	Local Console/Remote CLI i.e. SSH	NA	View/Modify
Event logs configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
Event logs	Local Console/Remote CLI i.e. SSH	NA	View
Version	Local Console/Remote CLI i.e. SSH	View	View/Modify

Sessions- Information about CLI connections	Local Console/Remote CLI i.e. SSH	View	View
Active-sesssion- Active sessions statistics	Local Console/Remote CLI i.e. SSH	View	View
User-info	Local Console/Remote CLI i.e. SSH	View	View
Syslog configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
SSH Server Configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
Policies	Local Console/Remote CLI i.e. SSH	View	View/Modify
Appliance-name	Local Console/Remote CLI i.e. SSH	View	View/Modify
Appliance-identifier	Local Console/Remote CLI i.e. SSH	View	View
Diagnostics configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
Interface- Interface status and configuration	Local Console/Remote CLI i.e. SSH	View	View/Modify
Ping	Local Console/Remote CLI i.e. SSH	View	View
Traceroute	Local Console/Remote CLI i.e. SSH	View	View
Signing-requests	Local Console/Remote CLI i.e. SSH	NA	View/Modify
OCSP responder settings	Local Console/Remote CLI i.e. SSH	NA	View/Modify
SSL device profile settings	Local Console/Remote CLI i.e. SSH	NA	View/Modify

Table 6 – User Roles

Read-only users:

- View system status and logs.
- View policy configuration and reports.
- View user information and statistics.
- View proxy server and cache settings.
- View network topology and connectivity information.

Read/Write users:

- Create, edit, and delete policies.
- Configure and modify proxy settings.
- Manage user accounts and groups.
- Configure and manage authentication and authorization settings.
- View and modify system logs and alerts.
- Configure and manage network and SSL settings.

- Any user with either read-only or read/write user-role can use the local/remote interface securely using appropriate authentication to perform their tasks without compromising the security of the TOE.

2.17.1 User Creation

- The following steps are involved in user creation:
  1. Create a local authentication realm.
  2. Create a list that includes usernames and passwords for members whom you wish to provide read-only access in the Management Console.
  3. Connect the list to the local realm.
  4. Create policy to enforce read-only access to members included in the list.
- To see the user information, use the following command:

  `#show user-info`

  This will show the user information of the current logged in account.

2.17.2 Realms

- Using a Local realm is appropriate when the network topology does not include external authentication or when you want to add users and administrators to be used by the ProxySG appliance only.
- The Local realm (you can create up to 40) uses a Local User List, a collection of users and groups stored locally.
- You can create up to 50 different Local User Lists. Multiple Local realms can reference the same list at the same time, although each realm can only reference one list at a time. The default list used by the realm can be changed at any time.
- Local realm authentication can be used to authenticate administrative users to the appliance management console and is highly recommended. Because the user details are stored on the appliance, local authentication realms are always available.
- **To create realm, use the following command:**

  `#(config) security local create-realm <realm_name>`

  Creates the specified local realm.
- **To delete realm, use the following command:**

  `#(config) security local delete-realm <realm_name>`

  Deletes the specified local realm.
- **To edit realm, use the following command:**

  `#(config) security local edit-realm <realm_name>`

  Changes the prompt. See Submodes for details.
- **To view realm, use the following command:**

  `#(config) security local view <realm_name>`

  Displays the configuration of all local realms or just the configuration for realm_name if specified.
- **To rename realm use the following command:**

  `#(config local realm_name) rename <new realm name>`

  Renames this realm to *new_realm_name*
- **To configure authorization for user use the following command:**

  `#(config local realm_name) validate-authorized-user {disable | enable}`

  When `validate-authorized-user` is enabled, an authorization (not authentication) request verifies that the user exists in the local user list. If the user does not exist in the list, the authorization request fails (authentication requests always require the user to exist).

  When `validate-authorized-user` is disabled, no user existence check is made for an authorization request. If the user does not exist, the authorization request succeeds.

2.17.3 Defining the Local User List

- The user list `local_user_database` is created on a new system or after an upgrade. It is empty on a new system.
- If a password file existed on the appliance before an upgrade, then the list contains all users and groups from the password file; the initial default user list is `local_user_database`.
- If a new user list is created, the default can be changed to point to it instead by invoking the `security local-user-list default list list_name` command.
- The local console account is not subject to the lockout mechanism. This account should not be used for day-to-day administrator.
- You can create up to 50 new lists with 10,000 users each.
- Lists can be uploaded, or you can directly edit lists through the CLI.
- **To create local user list use the following command:**
  `#(config) security local-user-list create <list_name>`
- **To delete local user list use the following command:**
  `#(config) security local-user-list delete <list_name>`
- **To view local user list use the following command:**
  `#(config) security local-user-list view <list_name>`
- **To edit local user list use the following command:**
  `#(config) security local-user-list edit <list_name>`

2.17.4 Subcommands:

- **To create User use the following command:**
  `#(config local-user-list list_name) user create` *<user_name>*
  Creates the specified user in the local user list.
- **To delete User use the following command:**
  `#(config local-user-list list_name) user delete` *<user_name>*
  Deletes the specified user in the local user list.
- **To edit User use the following command:**
  `#(config local-user-list list_name) user edit` *<user_name>*
  Changes the prompt to `#(config local-user-list list_name user_name)`
  Edits the specified user in the local user list.
- **To set password to User use the following command:**
  `#(config local-user-list list_name user_name)` **`password`** *<Enter> | <password>*
  Specifies the user's password.
- **To enable/disable a User use the following command:**
  `#(config local-user-list list_name user_name)` **`{disable | enable}`**
  Disables/enables the user account.
- **To set max-failed attempts for a User use the following command:**
  `#(config local-user-list list_name)` **`max-failed-attempts`** *<number>*
  The number of failed attempts to login to an ProxySG appliance before the user account is locked. The default is 60 attempts.
- **To set lockout-duration for a User use the following command:**
  `#(config local-user-list list_name)` **`no lockout-duration`**
  The length of time a user account is locked out after too many failed password attempts. The default is 3600 (one hour). If `no` command is used, the account does not automatically re-enable, but instead remains locked until manually enabled.
- **To set reset-interval for a User use the following command:**
  `#(config local-user-list list_name)` **`no reset-interval`** *<time in seconds>*
  The length of seconds to wait after the last failed attempt before resetting the failed counter to zero. If `no` command is used, the failed password count resets only when the account is enabled or when its password is changed. The default is 7200 seconds (two hours).
- **To disable any of these settings:**

```
#(config local-user-list list_name) no [max-failed-attempts]
```
Disables the settings for this user list.

2.17.5 Add the local-list to realm

- **To add the created local-list to realm use the following command:**
  ```
  #(config local realm_name) local-user-list list_name
  ```
  Specifies the local user list to for this realm.

2.17.6 Local Policy Creation

- **To create a local policy use the following command:**
  ```
  #(config)inline policy local <eof marker>
  ```
  Specify to enforce read-only or full access to members included in the list.
- **Example:**
  ```
  inline policy local end-xxx
  <Admin>

  authenticate(pqr)

  <Admin>

  ALLOW user=test1
  ALLOW user=test2 admin.access=READ

  Deny

  end-xxx
  ```

2.18 Configuring the Banner

2.18.1 Configuring Console Banner

- **Use this command to configure Console banner:**

```
#(config) serial-console
#(config-serial-console) inline pre-authentication-terms <eof marker>
```

Example:

```
#(config serial-console)inline pre-authentication-terms end-xxx
Welcome to Console session of SGOS
end-xxx
ok
```

- **Use this command to remove Console banner:**

```
#(config serial-console)no pre-authentication-terms <Enter>
```

- **Use this command to login console and SSH banner:**

```
#(config)banner login <string>
```

2.18.2 Configuring SSH Banner

- **Use this command to configure SSH banner:**

```
#(config) ssh-console
#(config ssh-console) inline welcome-banner <eof marker>
```

Example:

```
#(config ssh-console)inline welcome-banner end-xxx
Welcome to SSH session of SGOS
end-xxx
ok
```

- **Use this command to remove SSH banner:**

```
#(config ssh-console)no welcome-banner <Enter>
```

2.19 Configuring DNS

During initial configuration of the appliance, you configured the IP address of a single primary DNS server. You can add one or more alternate DNS servers, as well as define custom DNS service groups.

- **Use the following commands to create, delete, and edit Dns-forwarding groups for the appliance.**
  Syntax:
  `#(config) dns-forwarding`
  This changes the prompt to:
  `#(config dns-forwarding)`

2.19.1 Subcommands

`#(config dns-forwarding) create group-alias [host-ip]`
Creates a Dns-forwarding group.
`#(config dns-forwarding) delete group-alias`
Deletes a Dns-forwarding group.
`#(config dns-forwarding) edit {primary | alternate | group-alias}`
Edit a Dns-forwarding group.
`#(config dns-forwarding) exit`
Exits `#(config dns-forwarding)` mode and returns to `#(config)` mode.
`#(config dns-forwarding) view`
Displays snapshot status and configuration.
`#(config dns-forwarding group_name) add {domain domain | server server ip}`
Add domains or DNS servers to this group. IP addresses can be IPv4 or IPv6.
`#(config dns-forwarding group_name) clear {domain | server}`
Clear the domain or server list for this group.
`#(config dns-forwarding group_name) demote server_ip[slots]`
Demote the specified server IP address.
`#(config dns-forwarding group_name) exit`
Return to the #(config dns-forwarding) prompt.
`#(config dns-forwarding group_name) promote server_ip[slots]`
Promote the specified server IP address in the DNS server list the number of places indicated. Must be a positive number. If the number is greater than the number of servers in the list, the server is promoted to the first entry in the list.
`#(config dns-forwarding group_name) remove {domain | server}`
Remove a domain or server from the list.
`#(config dns-forwarding group_name) routing-domain <routing domain name>`
Associate a Dns-forwarding group with a configured Routing Domain.
`#(config dns-forwarding group_name) view`
View the Dns-forwarding configuration for this group.
Example:

```
SGOS#(config dns-forwarding) edit primary
SGOS#(config dns-forwarding primary) add server 1.1.1.1
ok
SGOS#(config dns-forwarding primary) view
Group: primary
Servers:
1.1.1.1
1.2.1.1
Domains:
  *
    Group: alternate
      Servers:
      Domains:
        *
SGOS#(config dns-forwarding) create testgroup 1.1.1.1
ok
SGOS#(config dns-forwarding) delete testgroup
```

```
ok
SGOS#(config dns-forwarding) edit primary
SGOS#(config dns-forwarding primary) exit
SGOS#(config dns-forwarding) view
Dns-forwarding configuration:
Group: testgroup
Servers:
1.1.1.1
Domains:
Group: primary
Servers:
 10.1.5.227
Domains:
  *
Group: alternate
Servers:
Domains:
  *
SGOS#(config dns-forwarding) exit
SGOS#(config)
```

3 Audit Record Examples

The TOE generates audit records for various events. The format of the audit records are as follows:

```
<Date/Time> <Process> <User> <Audit Message Content>
```

The following defines the fields included in an audit record:

- Date/Time: This is the date/time of when the auditable event occurred.
- Process: This identifies the actual process within the TOE which generated the auditable event.
- User: This is the name of the user that triggered the auditable event.
- Audit Message text: Descriptive text of the event that occurred.

The following are examples of audit records generated by ProxySG.

3.1 Start-up and shut-down of audit functions

```
2023-01-23 11:42:10-00:00UTC  "Syslog (10.1.5.227 6514): Connection established"  0
1C0001:96  syslogsocket.cpp:287

2023-01-23 11:42:26-00:00UTC  "Syslog (10.1.5.227 6514): Connection closed"  20
1C0001:96  syslogsocket.cpp:287
```

3.2 Administrative login and logout

```
2022-12-10 14:29:58-00:00UTC  "Administrator login, user 'admin', from secure serial
port"  0 250047:96  authconsole.cpp:1002

2022-12-10 14:30:02-00:00UTC  "Read/write mode entered from Serial for user 'admin'"
0 25001F:96  authconsole.cpp:785

2022-12-10 14:41:09-00:00UTC  "Administrator logout, user 'admin', from 'secure
serial port'"  0 250043:96  authconsole.cpp:909

2022-12-10  14:44:16-00:00UTC  "Administrator  login,  user  'admin',  from
192.168.254.24"  0 250047:96  authconsole.cpp:1002

2022-12-10 14:44:16-00:00UTC  "SSH: Accepted, login-authentication "password", user
"admin", realm "local", from 192.168.254.24, port "2847", protocol "ssh2" "  0
45000C:96  sgos_log.cpp:150

2022-12-10 14:44:22-00:00UTC  "Read/write mode entered from 192.168.254.24 for user
'admin'"  0 25001F:96  authconsole.cpp:785

2022-12-10  14:45:26-00:00UTC  "SSH:  Connection  closed  by  192.168.254.24"  0
45000C:96  sgos_log.cpp:150

2023-02-15 06:12:28-00:00UTC  "SSH: Closing connection to 192.168.228.49 port 51496
(user="admin")"  0 45000C:96  sgos_log.cpp:150
```

3.3 SSL device profile configuration changes

```
2022-12-10 14:53:25-00:00UTC  "Serial admin 'admin', SSL device profile: default
protocol changed to tlsv1.2"  0 140002:7D  cli_parse.hpp:315
```

```
2023-02-14 11:08:18-00:00UTC  "SSH admin at 192.168.228.49 'admin', SSL device profile: default cipher suite changed to ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256"  0 140002:7D cli_parse.hpp:315

2023-02-14 11:11:42-00:00UTC  "SSH admin at 192.168.228.49 'admin', SSL device profile: default verify-peer enabled"  0 140002:7D  cli_parse.hpp:315

2023-02-14 11:06:23-00:00UTC  "SSH admin at 192.168.228.49 'admin', SSL device profile:  default  CCL  changed  to  browser-trusted-fips"  0  140002:7D cli_parse.hpp:315
```

3.4 Deleting and creating key pairs

```
2022-12-10 15:02:03-00:00UTC "SSH: sshd: Deleted RSA host key pair"  0 45000C:96 admin.cpp:145


2022-12-10 15:02:07-00:00UTC  "SSH: Created key for sshv2 host"  0 45000C:96 sgos_log.cpp:150
```

3.5 Resetting passwords

```
2022-12-10 16:03:34-00:00UTC  "Serial admin 'admin', changed console password"  0 140002:7D  cli_parse.hpp:315

2023-02-14 13:26:51-00:00UTC  "SSH admin at 192.168.228.49 'admin', changed password for user 'unpriv' in local user list 'abc'"  0 140002:7D  cli_parse.hpp:315
```

3.6 Incorrect password

```
2022-12-10 16:26:01-00:00UTC  "Authentication failed from 10.1.5.227: user 'test', realm='xyz'"  0 250017:96  authutility.cpp:114

2022-12-10 16:26:01-00:00UTC  "SSH: Failed, login-authentication "password", user "test", realm "", from 10.1.5.227, port "48498", protocol "ssh2" "  0 45000C:96 sgos_log.cpp:150
```

3.7 Password length and complexity

```
2023-06-19 14:47:42-00:00UTC  "SSH admin at 10.1.5.227 'admin', the minimum number of digits in a password is set to 1"  0 140002:7D  cli_parse.hpp:315

2023-06-19 14:47:49-00:00UTC  "SSH admin at 10.1.5.227 'admin', password lengths now at 8 character minimum"  0 140002:7D  cli_parse.hpp:315

2023-06-19 14:47:56-00:00UTC  "SSH admin at 10.1.5.227 'admin', the minimum number of lowercase letters in a password is set to 1"  0 140002:7D  cli_parse.hpp:315

2023-06-19 15:15:40-00:00UTC  "SSH admin at 10.1.5.227 'admin', the minimum number of special characters in a password is set to 1"  0 140002:7D  cli_parse.hpp:315

2023-06-19 15:15:46-00:00UTC  "SSH admin at 10.1.5.227 'admin', the minimum number of uppercase letters in a password is set to 1"  0 140002:7D  cli_parse.hpp:315
```

```
2023-06-19 15:16:37-00:00UTC  "SSH admin at 10.1.5.227 'admin', reject common words
password is set to true"  0 140002:7D  cli_parse.hpp:315

2023-06-19 15:16:44-00:00UTC  "SSH admin at 10.1.5.227 'admin', reject password
containing whitespace is set to true"  0 140002:7D  cli_parse.hpp:315
```

3.8 Non-existent user

```
2022-12-10 16:28:52-00:00UTC  "Administrator login from secure serial port, user
'admiu', denied: Authentication failure, request='cli access'"  250017 250017:96
authconsole.cpp:1093

2022-12-10 16:28:52-00:00UTC  "Local: Authentication failed from 127.0.0.1: no such
user 'admiu' in realm 'xyz'"  1 250001:96  authutility.cpp:114
```

3.9 Account locked out

```
2022-12-10 16:26:01-00:00UTC  "Authentication failed from 10.1.5.227: user 'test',
realm='xyz'"  0 250017:96  authutility.cpp:114

2022-12-10 16:26:01-00:00UTC  "SSH: Failed, login-authentication "password", user
"test", realm "", from 10.1.5.227, port "48498", protocol "ssh2" "  0 45000C:96
sgos_log.cpp:150

2022-12-10 16:26:01-00:00UTC  "SSH: maximum authentication attempts exceeded for
test from 10.1.5.227 port 48498 ssh2"  0 45000B:1  sgos_log.cpp:15
```

3.10 SSH Connection

```
2022-12-10 16:39:45-00:00UTC  "SSH: Unable to negotiate with 10.1.5.227 port 48372:
no matching host key type found. Their offer: ssh-dss"  0 45000B:1  sgos_log.cpp:150

2022-12-10 16:39:24-00:00UTC  "SSH: Success: session established, protocol ssh-2"
0 45000C:96  sgos_log.cpp:150
```

3.11 SSH, login

```
2022-12-10 16:39:28-00:00UTC  "Administrator login, user 'test', realm 'xyz', from
10.1.5.227"  0 250047:96  authconsole.cpp:1002

2022-12-10 16:39:28-00:00UTC  "SSH: Accepted, login-authentication "password", user
"test", realm "xyz", from 10.1.5.227, port "37412", protocol "ssh2" "  0 45000C:96
sgos_log.cpp:150
```

3.12 SSH, wrong username

```
2022-12-10 16:52:06-00:00UTC  "Local: Authentication failed from 192.168.254.24: no
such user 'abcd' in realm 'xyz'"  9 250001:96  authutility.cpp:114
```

3.13 SSH, wrong password

```
2022-12-10 16:54:06-00:00UTC  "SSH: Failed, login-authentication "none", user
"test", realm "", from 192.168.254.24, port "2070", protocol "ssh2" "  0 45000C:96
sgos_log.cpp:150
```

```
2022-12-10 16:54:09-00:00UTC  "Authentication failed from 192.168.254.24: user
'test', realm='xyz'"  0 250017:96  authutility.cpp:114

2022-12-10 16:54:09-00:00UTC  "SSH: Failed, login-authentication "password", user
"test", realm "", from 192.168.254.24, port "2070", protocol "ssh2" "  0 45000C:96
sgos_log.cpp:150
```

3.14 Serial CLI, login

```
2023-02-21 06:33:52-00:00UTC  "Administrator login, user 'admin', from secure serial
port"  0 250047:96  authconsole.cpp:1002

2023-02-21 06:33:57-00:00UTC  "Read/write mode entered from Serial for user 'admin'"
0 25001F:96  authconsole.cpp:785
```

3.15 Serial CLI, wrong username

```
2022-12-10 16:57:00-00:00UTC  "Administrator login from secure serial port, user
'admin1', denied: Authentication failure, request='cli access'"  250017 250017:96
authconsole.cpp:1093

2022-12-10 16:57:00-00:00UTC  "Local: Authentication failed from 127.0.0.1: no such
user 'admin1' in realm 'xyz'"  D 250001:96  authutility.cpp:114
```

3.16 Serial CLI, wrong password

```
2022-12-10 16:58:14-00:00UTC  "Administrator login from secure serial port, user 'admin',
denied: Authentication failure, request='cli access'"  250017 250017:96  authconsole.cpp:1093
```

3.17 Configure the access banner

```
2022-12-10 17:01:39-00:00UTC  "SSH admin at 192.168.254.24 'test', realm='xyz',
changed login banner from "Welcome to Serial Console of SGOS  " to "Welcome to SGOS
""  0 140002:7D  cli_parse.hpp:315

2022-12-10 17:04:43-00:00UTC  "SSH admin at 192.168.254.24 'test', realm='xyz',
changed sshv2 pre-authentication banner message to 'Welcome '"  0 140002:7D
cli_parse.hpp:315

2022-12-10 17:07:43-00:00UTC  "SSH admin at 192.168.254.24 'admin', changed
config:serial_console to 'Welcome to serial console of SGOS '"  0 140002:7D
cli_parse.hpp:31
```

3.18 Configure the session inactivity time

```
2022-12-10 17:09:24-00:00UTC  "Serial admin 'admin', set the CLI console timeout to
2 minutes"  0 140002:7D  cli_parse.hpp:315

2023-02-01 10:48:16-00:00UTC  "SSH admin at 192.168.228.38 'admin', CLI session timed out.
Connection closing."  0 140002:7D  cli_parse.hpp:315
```

3.19 Configure syslog behaviour

```
2022-12-10 17:12:41-00:00UTC  "Serial admin 'admin', Eventlog config: added hostname
to syslog tls list, hostname='10.1.5.227' port=6514 device_profile='default'"  0
140002:7D  cli_parse.hpp:315

2023-02-21 07:33:51-00:00UTC  "Serial admin 'admin', Eventlog: enabled syslog"  0
140002:7D  cli_parse.hpp:315
```

3.20 Configure audit behaviour

```
2022-12-10 17:11:49-00:00UTC  "Serial admin 'admin', changed event log maximum size
from 6400 blocks to 12800 blocks (from 50 MB to 100 MB)"  0 140002:7D
cli_parse.hpp:315

2022-12-10 17:13:35-00:00UTC  "Serial admin 'admin', Eventlog: changed event log
overflow plan from "overwrite earlier events" to "stop logging events""  0 140002:7D
cli_parse.hpp:315
```

3.21 Configure the cryptographic functionality

```
2022-12-10 15:56:31-00:00UTC  "SSH admin at 10.1.5.227 'admin', SSL device profile:
default cipher suite changed to
TLS_AES_256_GCM_SHA384:TLS_AES_128_GCM_SHA256:TLS_AES_128_CCM_8_SHA256:TLS_AES_128
_CCM_SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-RSA-
AES256-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA:ECDHE-RSA-AES128-
SHA:DHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-SHA:DHE-
RSA-AES128-SHA"  0 140002:7D  cli_parse.hpp:315

2023-01-31 09:18:39-00:00UTC  "Created signing request for keyring 'appliance-key'"
0 140002:7D  cm.cpp:5316
```

3.22 NTP Configuration

```
2023-03-31 14:04:02-00:00UTC  “SSH admin at 192.168.228.60 ‘admin’, added NTP server
“10.1.5.227””  0 140002:7D  cli_parse.hpp:315

2023-03-28 14:06:35-00:00UTC  "SSH admin at 192.168.228.60 'admin', added NTP server
with auth info for "10.1.3.78""  0 140002:7D  cli_parse.hpp:315

2023-03-31 14:04:07-00:00UTC  "SSH admin at 192.168.228.60 'admin', changed NTP
update interval to 1 and issued an NTP update"  0 140002:7D  cli_parse.hpp:315

2023-03-31 14:04:07-00:00UTC  "NTP: Queried server 10.1.5.227 per user request,
system clock is 259201 seconds 247 ms fast compared to NTP time. Updated system
clock."  0 90000:1  ntp.cpp:1073

2023-03-28 14:07:06-00:00UTC  "NTP: Periodic query of server 10.1.5.227, time
matches system clock."  0 90000:96  ntp.cpp:1112
```

3.23 Max-failed attempts configuration

```
2023-02-21 07:44:24-00:00UTC  "Serial admin 'admin', set the maximum failed login
attempts before automatic user lockout for database 'abc' to 3"  0 140002:7D
cli_parse.hpp:315

2023-02-21 07:44:31-00:00UTC  "Serial admin 'admin', set the user lockout duration
for database 'abc' to 0"  0 140002:7D  cli_parse.hpp:315
```

```
2023-02-21 07:44:38-00:00UTC  "Serial admin 'admin', set the user lockout reset interval for database 'abc' to 0"  0 140002:7D  cli_parse.hpp:315
```

3.24 User lockout and enable

```
2023-02-21 10:17:33-00:00UTC  "Local: User 'test' in realm 'xyz' exceeded maximum failed password attempts; user locked out"  0 25002A:96  authutility.cpp:114

2023-02-21 10:53:29-00:00UTC  "SSH admin at 192.168.228.44 'admin', enabled user 'test' in local user list 'abc'"  0 140002:7D  cli_parse.hpp:315
```

3.25 Trust store configuration

```
2023-02-10 20:21:49-00:00UTC  "Imported ca-certificate 'ca'"  0 140002:7D  cm.cpp:902

2023-02-10 20:22:04-00:00UTC  "SSH admin at 192.168.254.24 'admin', added ca-certificate "ca" to ccl "browser-trusted-fips" "  0 140002:7D  cli_parse.hpp:315

2023-02-10 20:02:25-00:00UTC  "SSH admin at 192.168.254.24 'admin', deleted ca-certificate "ca" from ccl "browser-trusted-fips" "  0 140002:7D  cli_parse.hpp:315

2023-02-10 20:02:45-00:00UTC  "Deleted ca-certificate 'ica'"  0 140002:7D  cm.cpp:918
```

3.26 External Certificate configuration

```
2023-02-08 12:37:16-00:00UTC  "SSH admin at 192.168.228.242 'admin', Imported external certificate "test""  0 140002:7D  cli_parse.hpp:315

2023-02-08 12:37:31-00:00UTC  "SSH admin at 192.168.228.242 'admin', added external certificate "test" to ecl "abc" "  0 140002:7D  cli_parse.hpp:315

2023-02-08 12:37:36-00:00UTC  "SSH admin at 192.168.228.242 'admin', deleted external certificate "test" from ecl "abc" "  0 140002:7D  cli_parse.hpp:315
```

3.27 Initiation of update

```
2023-01-31 09:18:24-00:00UTC  "System startup"  0 7FFF0009:96  event_logger.cpp:508

2023-01-31 09:18:24-00:00UTC  "Executing image: Version: SGOS 7.4.0.0, Release id: 280944 64-bit, gdb, unoptimized"  0 7FFF0009:1  event_logger.cpp:538

2023-01-31 09:18:24-00:00UTC  "This system is in FIPS mode"  0 7E0000:96  cf_main.cpp:1049
```

4 Objectives for the Environment

Security objectives for the operational environment assist the TOE in correctly providing its security functionality. These objectives, which are found in the table below, track with the assumptions about the TOE operational environment.

ID	Objectives for the Operational Environment
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE. Note: For vNDs the TOE includes only the contents of the its own VM, and does not include other VMs or the VS.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner. For vNDs, this includes the VS Administrator responsible for configuring the VMs that implement ND functionality. For TOEs supporting X.509v3 certificate-based authentication, the Security Administrator(s) are assumed to monitor the revocation status of all certificates in the TOE's trust store and to remove any certificate from the TOE's trust store in case such certificate can no longer be trusted.
OE.UPDATES	The TOE firmware and software is updated by an Administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The Administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment. For vNDs, this applies when the physical platform on which the VM runs is removed from its operational environment.

ID	Objectives for the Operational Environment
OE.VM_CONFIGURATION	For vNDs, the Security Administrator ensures that the VS and VMs are configured to • Reduce the attack surface of VMs as much as possible while supporting ND functionality (e.g., remove unnecessary virtual hardware, turn off unused inter-VM communications mechanisms), and • Correctly implement ND functionality (e.g., ensure virtual networking is properly configured to support network traffic, management channels, and audit reporting).

Table 7 – Security Objectives for the Operational Environment

5 Self-Test Error

If the device has rebooted successfully, it can be inferred that all claimed self-tests (AES Known Answer Test, HMAC Known Answer Test, RNG /DRBG Known Answer Test, SHA Known Answer Test, RSA Signature Known Answer Test (both signature/verification), DH Known Answer Test, ECDH Known Answer Test) have been completed and passed without any issues.

Indication of self-test failures are printed on the local console and the boot process is terminated. Any indication of power on self-test failures, users should contact Symantec customer support for instructions on how to proceed.

If the module fails the POST Integrity Test, the following error is printed to the CLI (when being accessed via the serial port):

```
PKCS7 Signature verification failed, signature does not match.
```

If any other self-tests fail, the following error is printed to the CLI (when being accessed via the serial port):

```
**********SYSTEMERROR********** The SG Appliance has
failed the FIPS Self test.

System startup cannot continue.



**********SYSTEM STARTUP HALTED**********

E)xit FIPS mode and reinitialize system

R)estart and retry FIPS self-test Selection:
```

When either of these errors occurs, the modules halt operation and provide no functionality. The only way to clear the error and resume normal operation is for the Crypto-Officer to reboot the modules. The status output provided above is shown only over the CLI (when being accessed via the serial port).

5.1 Power-Up Self-Tests

The module performs the following self-tests using the UEFI OS Loader:

- Known Answer Tests
  - SHA KAT using each of SHA-1 and SHA-256;
  - HMAC KAT using each of SHA-1; and
  - RSA Sign/Verify KAT with SHA-256.
- Firmware integrity check

The module performs the following self-tests using the SGOS Cryptographic Library software.
implementation at power-up:

- Known Answer Tests
  - AES KAT for encryption and decryption
  - AES-GCM KAT for decryption and decryption
  - TDES KAT for encryption and decryption
  - SHA KAT using each of SHA-1, SHA-224, SHA-256, SHA-384, SHA-512
  - HMAC KAT using each of SHA-1, SHA-224, SHA-256, SHA-384, SHA-512
  - RSA Sign/Verify KAT with SHA-256
  - RSA wrap/unwrap KAT

- SP800-90A DRBG KAT
- DH “Primitive Z” KAT
- ECDH “Primitive Z” KAT

No data output occurs via the data output interface until all power-up self-tests have completed.

5.2 Conditional Self-Tests

The module performs the following conditional self-tests found in its SGOS Cryptographic Library only.

- Firmware Load Test using RSA Signature Verification
- RSA pairwise consistency check upon generation of an RSA keypair
- Continuous RNG test (CRNGT) for the SP800-90A DRBG
- Continuous RNG test (CRNGT) for the Non-Deterministic Random Number Generator (NDRNG)

5.3 Critical Function Tests

The ProxySG performs the following critical function tests:

- DRBG Instantiate Critical Function Test
- DRBG Reseed Critical Function Test
- DRBG Generate Critical Function Test
- DRBG Uninstantiate Critical Function Test

The module also performs a validity check on the installed license. If the license is not valid, the module will not operate.

6 Cryptographic Key Destruction

The table below describes the key zeroization provided by the TOE and as referenced in FCS_CKM.4.

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
Master Encryption Key (MEK)	Encrypting Crypto-Officer password, RSA private key	Stored in plaintext on non-volatile memory	By disabling the FIPS-Approved mode of operation
Integrity Test Public Key	Verifying the integrity of the system image during upgrade or downgrade	Stored in plaintext on non-volatile memory	Overwritten after upgrade by the key in the newly signed image
RSA Public Keys	Negotiating TLS or SSH sessions	Stored in encrypted form on non-volatile memory	Module's public key is deleted by command
RSA Public Key	Negotiating TLS or SSH sessions	Other entities' public keys reside on volatile memory	Other entities' public keys are cleared by power cycle
RSA Private Keys	Negotiating TLS or SSH sessions	Stored in encrypted form on non-volatile memory	Inaccessible by zeroizing encrypting MEK
DH public key	Negotiating TLS or SSH sessions	Stored in plaintext on volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
DH private key	Negotiating TLS or SSH sessions	Stored in plaintext on volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
ECDH private key	Negotiating TLS or SSH sessions	Stored in plaintext on volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
ECDH public key	Negotiating TLS or SSH sessions	Stored in plaintext on volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
TLS or SSH Session key	Encrypting TLS or SSH data	Stored in plaintext on volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules

Keys/CSPs	Purpose	Storage Location	Method of Zeroization
			Removing power
TLS or SSH Session Authentication key	Data authentication for TLS or SSH sessions	Resides in volatile memory in plaintext	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
Crypto Officer Password User Password	Locally authenticating a CO or User for Management Console or CLI	Stored in encrypted form on non-volatile memory	Inaccessible by zeroizing the encrypted MEK
“Enabled” mode password	Used by the CO to enter the “privileged” or “enabled” mode when using the CLI	Stored in encrypted form on non-volatile memory	Inaccessible by zeroizing the encrypting MEK
“Setup” Password	Used by the CO to secure access to the CLI when accessed over the serial port	Stored in encrypted form on non-volatile memory	Inaccessible by zeroizing the encrypting MEK
SP 800-90A CTR_DRBG Seed	Seeding material for the SP800-90A CTR_DRBG	Plaintext in volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power
SP 800-90A CTR_DRBG Entropy	Entropy material for the SP800-90A CTR_DRBG	Plaintext in volatile memory	Inaccessible by disabling FIPS-mode Rebooting the modules Removing power

Table 8 – Key Storage and Zeroization

7 References

1. ISG 2.1 Administration and Deployment Guide
2. Edge SWG 7.4.x Command Line Interface Reference
3. The security target document- Symantec Edge Secure Web Gateway (SWG) with SGOS v7.4 Security Target Document Version: 1.0