Common Criteria

KlasOS Keel 5.4.0 Operational User Guidance

V0.4

June 2024

Klas Group
1101 30th NW Street, Suite 500
Washington, DC 20007
Phone: 202-625-8315
Email: sales@klasgroup.com

Common Criteria KlasOS Keel Operational User Guidance	
Published June 2024	

Contents

Common Criteria KlasOS Keel Operational User Guidance	KLAS TELECOM
Published June 2024	

1. Introduction

The Klas Voyager Keel Family Operational User Guidance documents the administration of the Voyager Keel Family of devices as it was certified under Common Criteria. The ToE was evaluated against the collaborative Protection Profile for Network Devices (cPPND) v2.2E and the PP-Module for Stateful Traffic Filter Firewalls, Version 1.4e (MOD_CPP_FW_V1.4E).

1.1. Audience

This document is written for administrators of the ToE. This document assumes that the administrator is familiar with the basic concepts and terminologies used in networking, that the administrator is a trusted individual and is trained and knowledgeable in the configuration of network devices.

The administrator must ensure the guidance items within this document are enforced.

1.2. Purpose

This document is the Common Criteria Operational User Guidance for administrators of the ToE. The administrator must ensure that all guidelines in this document are implemented if the device is to operate in a Common Criteria evaluated mode of operation. It was written to highlight the specific ToE configuration and administrator functions and interfaces that are necessary to configure and maintain the ToE in the evaluated configuration.

1.3. Supported Hardware and Software

The Klas Keel Family Common Criteria evaluated module is a component of the following Klas products:

TOE Model	Specifications
VoyagerVMm (i3) and VoyagerVMm (i5) 	5th Gen Intel® Dual Core i3-5010U (1.8 GHz) Broadwell-U, 8 GB DDR3 RAM Network Ports: 1 x console, 4 x Gb Ethernet Storage: Samsung 850 EVO 256 GB mSATA SSD or Samsung 1TB mSATA SSD
	5th Gen Intel® Quad Core i5-5350U (1.8 GHz) Broadwell-U, 32 GB DDR3 RAM Network Ports: 1 x console, 4 x Gb Ethernet

TOE Model	Specifications
	Storage: Samsung 850 EVO 256 GB mSATA SSD or Samsung 1TB mSATA SSD
TRX R2 (4-core) and TRX R2 (8 core) 	Atom™/Denverton C3508 Intel® Atom™ Denverton C3508 4-Core processor with 1.6 GHz clock. 8 GB RAM (upgradeable to 32 GB) Network Ports: 2 x 1 Gb Ethernet 4G/LTE Modems Sierra Wireless EM7455 LTE Cat-6 (B1, B2, B3, B4, B5, B7, B12, B13, B20, B25, B26, B29, B30, B41) Sierra Wireless EM7511 LTE Cat-12 (B1, B2, B3, B4, B5, B7, B8, B9, B12, B13, B14, B18, B19, B20, B26, B29, B30, B32, B41, B42, B43, B46, B48, B66) IEEE802.11 ac/b/g/n 3×3 MIMO Wi-Fi modem with data rates of 1.3 Gb/s downlink in 2.4/5 Ghz bands
	Atom™/Denverton C3708 Intel® Atom™ Denverton C3708 4-Core processor with 1.7 GHz clock. 8 GB RAM (upgradeable to 32 GB) Network Ports: 2 x 1 Gb Ethernet 4G/LTE Modems Sierra Wireless EM7455 LTE Cat-6 (B1, B2, B3, B4, B5, B7, B12, B13, B20, B25, B26, B29, B30, B41)

TOE Model	Specifications
	Sierra Wireless EM7511 LTE Cat-12 (B1, B2, B3, B4, B5, B7, B8, B9, B12, B13, B14, B18, B19, B20, B26, B29, B30, B32, B41, B42, B43, B46, B48, B66) IEEE802.11 ac/b/g/n 3×3 MIMO Wi-Fi modem with data rates of 1.3 Gb/s downlink in 2.4/5 Ghz bands
VoyagerVM 3.0 	Xeon D-1539 Intel® Xeon Processor D1539 16-Core with 48 or 96 GB RAM Network Ports: 1 x console, 2 x 10 GB SFP, 2 x 1GB ethernet Storage: removable SATA dual SSDs, removable NVMe Voyager Ignition Key (VIK+)
	Xeon D-1559 Intel® Xeon Processor D1559 12-Core with 48 or 96 GB RAM Network Ports: 1 x console, 2 x 10 GB SFP, 2 x 1GB ethernet Storage: removable SATA dual SSDs, removable NVMe Voyager Ignition Key (VIK+)
	Xeon D-1577 Intel® Xeon Processor D1577 16-Core with 48 or 96 GB RAM

TOE Model	Specifications
	Network Ports: 1 x console, 2 x 10 GB SFP, 2 x 1GB ethernet Storage: removable SATA dual SSDs, removable NVMe Voyager Ignition Key (VIK+)

The following software version is the Common Criteria validated software:

- KlasOS Keel 5.4.0

Only this software can be used to operate in the validated Common Criteria mode of operation.

1.4. Operational Environment and Secure Acceptance

Your KlasOS Keel device will be delivered via commercial carrier. Perform the following checks when receiving the hardware device:

- Verify that the correct hardware device has been received
- Inspect all packaging contents to ensure there has been no damage or tampering

If either of the checks above fails, please reach out to the support team.

1.4.1. Technical Requirements

The Voyager Keel Family Hardware Guides provide a clear and detailed description of the ports and interfaces available on the device. These documents along with software downloads and other documentation can be found on the Klas Telecom Forum website:

- https://helpdesk.klasgov.com/

User registration is required to access the forum and this access can be requested by submitting a request through the web page. Once access is granted, the administrator should get familiar with this site and the documentation on the site.

The full URL to the Voyager Keel Family Hardware Guides are located here:

- VoaygerVMm - https://helpdesk.klasgov.com/s/article/voyagervmm-hardware-guide
- TRX R2 - https://helpdesk.klasgov.com/s/article/TRX-R2
- VoyagerVM 3.0 - https://helpdesk.klasgov.com/s/article/VoyagerVM-3-0-Hardware-Guide

Klas Voyager Keel Family devices are compatible with the following devices, servers and protocols:

- Syslog server
  - The syslog audit server is used for remote storage of audit records that have been generated by and transmitted from the ToE.
  - Required in a CC validated environment.
  - Uses an SSH tunnel to encrypt syslog traffic
- Local console
  - This includes any IT Environment Console that is directly connected to the ToE via the Serial Console Port and is used by the ToE administrator to support ToE administration
- Management workstation with SSH client
  - 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. Any SSH client that supports SSHv2 may be used.
  - The ToE SSH implementation is based on the following RFCs: 4251, 4252, 4253, 4254, 5656 and 6668.
- NTP Server
  - The TOE can sync with an external NTP server using NTPv3
  - The NTP server can be configured to have secure timestamps using the SHA1 message digest algorithm

Klas Voyager Keel device provides the following features which are outside the scope of the NIAP Common Criteria validation:

- SNMP
- Spanning-Tree
- Port Security
- TACACS+
- RADIUS

1.4.2. Procedural Requirements

To ensure that the device is operating in Common Criteria validated mode, the following steps must be followed by an Administrator:

- The Klas Voyager Keel device ToE is physically protected from unauthorized access. It is recommended to not connect the device to any external network until the validated firmware image has been installed and booted and the device configured by the administrator.
- The Administrator must ensure that the guidance specified in this document is adhered to.
- The Administrator must download the Common Criteria validated KlasOS firmware image (See section "Supported Hardware and Software) from the Klas Telecom recommended download location:
    - https://helpdesk.klasgov.com/s/article/KlasOS-Keel-Download
        - **NOTE:** User registration is required to download the firmware images.
        - The CC evaluated version will be clearly highlighted. See Section 2.1 'Software Installation' for details.
- Documentation for the Voyager Keel devices are in the Voyager Keel section located here:
    - https://helpdesk.klasgov.com/s/article/klasos-initial-configuration1
        - **NOTE:** User registration is required.

1.4.3. KlasOS

KlasOS is the firmware that runs on the ToE. All configuration of KlasOS is done using a Cisco-like command line interface. There are 3 CLI modes in KlasOS:

- User EXEC mode:
    - This is the initial prompt that a user gets after booting up the device. It will look like the this:
        - `KlasOS>`
    - This is a non-administrator mode and only basic show commands can be used in this mode.
- Privileged EXEC Mode:
    - The '**enable**' command is used to enter privileged exec (administrator level) mode. This is for administrators only and the administrator **MUST** set a password to avoid other users entering this mode. By default, there is no password set. The password for this mode can be set using the **'enable secret'** command in global configuration mode. See details on setting passwords in the section User Authentication. In this mode, an administrator can run all '`show`' commands on the system and run various administrative functions such as saving configuration files, setting the system clock and displaying logs.
      The enable mode prompt is the hostname followed by a hash symbol:

- **KlasOS#**

- Global Configuration Mode:
    - This mode is where most of the configuration of the ToE is performed. To enter Global Configuration mode, enter the command '**configure terminal**' from the Privileged EXEC mode prompt. The Global Configuration mode prompt looks like the following:
        - **KlasOS(config)#**

When a command is typed on the CLI, the administrator can see what the next available parameter for that command is by pressing either '**?**' or **<TAB>**.

A privileged EXEC mode command can be run from Global Configuration by preceding the command with the '**do**' command.

To return from Global Configuration mode to Privileged EXEC mode type '**exit**'. The '**exit**' command can also be used to exit out of Privileged EXEC mode. Typing '**exit**' from User EXEC mode will log the user out of the ToE.

To display the current running configuration, run the following command from Privileged EXEC mode:

- **show running-config**

To display the current startup/boot configuration, run the following command from Privileged EXEC mode:

- **show startup-config**

2. Software Installation and Initial Setup

2.1. Software Installation

Before powering on the ToE, connect a PC or laptop to the console port using an RJ-45 to RS-232 console cable and configure the terminal emulator with the following settings:

- Baud: 9600
- Data bits: 8
- Parity: None
- Stop bits: 1
- Flow control: none

Continue to power on the ToE until bootup completes and the following is seen on the terminal window:

```
Press RETURN to get started.
```

Pressing **<ENTER>** here will give a **KlasOS>** prompt.

Upgrade the ToE so that it is using the CC evaluated software as follows:

- Connect an Ethernet port of the PC or laptop to an ethernet port on the ToE. The below configurations assume ethernet 0/0 is used.
- Enable Privileged EXEC Mode:

```
KlasOS> enable
Mar 22 17:58:43 KlasOS : %SYS-5-PRIV_AUTH_SUCCESS: Authentication to privilege
level 15 succeeded by klas on ttyS0
KlasOS#
```

- Configure interface ethernet 0/0 with an IP address as follows::

```
KlasOS# configure terminal
KlasOS(config)# interface ethernet 0/0
KlasOS(config-if)# ip address 192.168.1.1 255.255.255.0
KlasOS(config-if)#
KlasOS(config-if)# end
KlasOS#
```

- Configure the connected PC or laptop with an IP address on the same network, e.g. 192.168.1.10/24.
- Copy the downloaded software (see previous section) to the PC/laptop and move it to the SCP server directory.
- If using SCP to copy the software do the following:

- **copy scp: flash:**
  - You will be required to enter the IP address and username for the SCP server and the name of the firmware image to be copied.
  - **IMPORTANT NOTE:** Ensure the firmware image name that you use is a different name to an image that is already installed. If you overwrite the previous image with the new image and it fails the digital signature verification, the image will be deleted. This will result in no valid image present on the ToE. If the ToE is then rebooted, it will not boot up as no valid image is present. No functions are affected during an update and only during the reboot following the update.

- Check the image is present using the 'show flash:' command:
  - **show flash:**

- Verify the signature on the firmware image:
  - The uploaded firmware image must be verified to check that the digital signature is correct before proceeding any further. See Section 2.2 Verifying the Firmware Image below for instructions on how to do this.

- Specify that this newly copied image is the image to be booted:
  - **boot system flash <name of image>**

Once this command is executed, the new image will now load when the system is rebooted. A log message will be generated in the audit log signifying that the firmware has been installed as follows:

```
 2024-04-30T20:06:38.685029+00:00 KlasOS %SIG_VER:  Verifying boot image file
 2024-04-30T20:06:42.764413+00:00 KlasOS %IMG_VER:   Good signature from "Klas
Telecom (Klas Signed Image Key) <trxsupport@klastelecom.com>" [unknown]
```

If the update failed because the image is not a valid image or the image name entered does not exist on flash: then the following message will be displayed:

```
2024-04-30T20:06:38.685029+00:00 KlasOS CLI[10476]: (admin) (klas) (KlasOS) boot
system flash KlasOS.keel.v5.4.0rc7.bin: Failure
```

The format of these messages are as follows:

```
[Date and Time] [IP address or Hostname of ToE] CLI[process id]: (admin) [user]
[Origin of CLI command] [CLI command] : [success|failure]
```

- Reboot the device:
  - **reload**

2.2. Verifying the Firmware Image

The firmware image must be verified against the digital signature before it can be installed. This is done using the following command:

- **verify /bootimgver flash: <name of image>**

The command to initiate the signature verification will be sent to the audit log as follows:

```
 2024-04-30T20:06:42.789521+00:00 KlasOS CLI[6396]:  (admin) (root) (ttyS0) verify
/bootimgver  KlasOS.keel.v5.4.0rc7.bin : Success
```

See Section 8 'Logging and Auditing' for details on the format of the audit log messages.

If the verify command is successful, the image can be installed using the 'boot system flash' command (See section 2.1). When the device is be rebooted, it will bootup using the installed Common Criteria verified image. A log message indicating success will be displayed on the console and logged to the system log as follows:

```
 2024-04-30T20:06:38.685029+00:00 KlasOS %SIG_VER:  Verifying boot image file
 2024-04-30T20:06:42.764413+00:00 KlasOS %IMG_VER:   Good signature from "Klas Telecom
(Klas Signed Image Key) <trxsupport@klastelecom.com>" [unknown]
```

See Section 8 'Logging and Auditing' for details on the format of the system log messages.

If the digital signature validation **fails,** the image will be automatically deleted from the device and cannot be used. An error message will be displayed on the console and also logged to the system log as follows:

```
 2024-04-30T20:20:16.430830+00:00 KlasOS %SIG_VER:  Verifying boot image file
 2024-04-30T20:20:20.562678+00:00 KlasOS %IMG_VER_FAIL:  The signature could not be
verified.
```

See Section 8 'Logging and Auditing' for details on the format of the system log messages.

The administrator must verify after bootup that they are currently running the Common Criteria validated image. This is done by entering the '**show version'** command in the CLI.

2.3. Initial Configuration

When the ToE is booted up for the first time there is no configuration applied, all ports are in bypass mode and **ALL** data can flow through the device.

Initial configuration of any of the ToE must be done through the serial console (See Section 2.1 'Software Installation' for details on connecting through serial console). It is recommended that the administrator fully configures the device before connecting to any external network.

For initial configuration, the administrator must enter Global Configuration mode and then proceed to configure the ToE in a Common Criteria compliant manner (See Section 2.6 'ToE CC Compliant Configuration' for details.)

2.4. TOE Interface

The RJ45 console port is used for connecting to the serial terminal and for initial configuration of the ToE (See Section 2.1 'Software Installation')

To retrieve the current configuration on a ToE interface, the administrator can execute any of the following commands from Privileged EXEC mode:

- **`show running-config`**
- **`show interfaces`**
- **`show ip interface brief`**

2.5. Saving Configuration

ToE configuration can be saved so that it is restored on device bootup.
To save the configuration to the startup configuration use either of the following commands from Privileged EXEC mode:

- **`copy running-config startup-config`**
  or
- **`write`**

The configuration can also be saved into a separate configuration file and stored in any of the following locations:

- flash
- SCP to external PC
- TFTP to external PC
- usbflash (VIK)

This is done using the '**copy running-config <destination>'** command.
The administrator can specify which configuration to load on bootup by using the '**boot config-file <flash>'** command.

NOTE: To be in a validated mode, the ToE must not be configured to boot from a configuration file located on the VIK. Verify the ToE is not configured to boot from a configuration file on the VIK by running the command **show boot** in Privileged EXEC mode and verifying the Config file is not located on usbflash. Typing no **boot config-file** in global configuration mode will set the ToE to the default behavior of booting from the startup-config in NVRAM.

2.6. ToE CC Compliant Configuration

To ensure the ToE is operating in a CC compliant configuration the following actions must be performed on the ToE after the CC firmware image has been loaded and verified.
This configuration **MUST** be completed before the ToE is connected to any network:

- Configure an Administrator password:
  - See section 3.1 below
- Configure username(s) and password(s):
  - See section 3.1 below

- Configure an Access Banner:
  - See section 3.2 below
- Configure the Inactivity Timer:
  - See section 3.3 below
- Configure Account Locking:
  - See section 3.4 below
- Configure Time:
  - See section 5.4 for manual clock configuration.
- Generate a public/private key-pair:
  - A public/private key-pair is required for SSH remote administration.
  - See section 6.1 for information on generating a public/private key-pair.
- Configure SSH for Remote Administration:
  - See section 7.
  - This is optional and only required if SSH is required for remote administration.
- Configure SSH Tunnel for trusted path:
  - See section 9.
  - This is required to send syslog messages over a trusted channel to a remote syslog server
- Configure a Syslog server:
  - See section 8.4.
  - **NOTE:** A trusted channel using SSH must be configured between the ToE and the syslog server.
- Configure the HTTPS server:
  - See section 4.1

NOTE: The ToE Random Number Generator does not need to be configured and is automatically functional when the ToE has completed boot up.

3. User Identification and Authentication

The ToE provides a password-based login mechanism. Many non-administrative users can be configured on the ToE but there is ONLY one administrator user which is the **'Privileged EXEC mode'** user (See Section 1.4.3 'KlasOS'). This is really a privilege escalation rather than a distinct user and the '**enable**' command is required to enter Privileged EXEC mode.

The ToE by default has no preconfigured usernames or passwords and the administrator password is not set, therefore, the administrator is required to add users and set the administrator password for the device, before it can be used on a network.

The ToE supports both local administration using the local console port and remote administration using SSH (See Section 7 'Remote Administration using SSH').

No configuration is required to ensure the following services are available for each login method before authenticating to the TOE:

- Display the warning banner
- Respond to ICMP requests

Authentication is performed by providing the username and password and all passwords are obscured during logon. Successful authentication will give the CLI prompt and a message saying authentication was successful. An unsuccessful authentication attempt will drop the user back to the login prompt and display a message saying that login failed. A failed login attempt would look like this:

```
KlasOS login: klas
Password:


Authentication failed

KlasOS login:
```

An error message will be logged to the system log as follows:

```
 2024-04-30T20:21:51.914014+00:00 KlasOS login:  %SYS-5-PRIV_AUTH_FAIL:
Authentication Failed by klas on ttyS0
```

3.1. Passwords

All account passwords must be chosen to include a combination of capital and lower-case letters, numbers, and special characters. Easily guessed and common language (dictionary) words should be avoided. Password strength is a function of length and complexity. Longer passwords provide more protection against brute-force attacks. Klas recommend using as long and as complex a password that can be remembered. No configuration is necessary for obscuring authentication data.

The minimum requirement is for a 15 character password which supports special characters from the set **!#$%&()*+,-./[]^_{|}~=<>@;:.** The minimum password length can be increased by an administrator to up to 128 characters. The "\" character is interpreted as an escape character and is silently stripped from the password if entered.

The minimum password length can be set using the following command in Global Configuration mode and must be set to a minimum of 15 characters:

- **security passwords min-length <minimum password length>**

The administrator password is configured on the ToE using the following command in Global Configuration mode.

- **enable secret <admin password>**
  - The password must be at least 15 characters and should use at least one of the characters specified above. The minimum password length of at least 15 characters should be set using the "security passwords min-length" command above.

Non-administrative usernames and passwords are configured using one of the following commands in Global Configuration mode:

- **username <username> secret <password>**
  - The password hashing algorithm used is SHA512. These passwords must also be at least 15 characters long.
- **username <username> algorithm-type sha512 secret <password>**
  - The password hashing algorithm used is SHA512. These passwords must also be at least 15 characters long.

The following commands are required so that a login prompt appears after bootup rather than dropping directly into the User EXEC mode shell:

- **aaa new-model**
- **aaa authentication login default local**

The usernames and passwords configured above will be used to login to the device.

3.2. Access Banner

The ToE can use the login banner to display an advisory notice and consent warning message regarding use of the ToE. This message is displayed before the login prompt is shown. To set the login banner do the following from Global Configuration mode:

- This command would set the login banner to "This is my login banner":
  - **banner login "This is my login banner"**

To add a banner with multiple lines, use "///" in the command above to add a carriage return/line feed (CR/LF).

- This command would set a multi-line login banner::
  - **banner login "This is my login banner///This is the second line of my login banner"**

Note: If an administrator desires to

3.3. Session Termination

A user can terminate their own interactive session by entering the **'exit'** command at the CLI prompt.
If the user is in Privileged EXEC mode and the exit command is entered, the administrator session will end and the user will be dropped back into User EXEC (non-administrator) mode. The user will need to enter the '**enable**' command and re-authenticate to return to the Privileged EXEC mode.
If the user is in User EXEC mode and the exit command is entered the user will be logged out completely and will have to enter the username and password to log back in.

A session inactivity timer can also be configured for both local console and remote SSH sessions. After this time-period expires, the session will close and the user will be logged out.
To configure the session inactivity timer for the local console, do the following from Global Configuration mode:

- **line console 0**
  - **exec-timeout <mins> <secs>**

A log message similar to the following will be displayed in the system log when the user is automatically logged out of the console session.

```
 2024-04-30T20:29:38.764467+00:00 KlasOS :  %SYS-5-PRIV_AUTH_TIMEOUT: Session for user klas timed out on ttyS0
```

To configure the session inactivity timer for a remote SSH session, do the following from Global configuration mode:

- **line vty 0 4**
  - **exec-timeout <mins> <secs>**

A log message similar to the following will be displayed in the system log when the user is automatically logged out of the SSH session.

```
 2024-04-30T20:31:44.176220+00:00 KlasOS :  %SYS-5-PRIV_AUTH_TIMEOUT: Session for user klas timed out on pts/3
```

3.4. Account Locking

The ToE can be configured so that a remote user will be locked out after a number of unsuccessful login attempts. The remote user will be locked out until a local administrator manually unlocks the account from a local console.

NOTE: The ToE will always allow a user to authenticate using the local console port, even if the user account is locked. This behavior is not configurable.

The following command can be used to configure a maximum number of authentication attempts by a user from global configuration mode:

- **aaa authentication attempts max-fail <number of failures>**

A log event similar the following will be displayed in the system log if a max-fail of 3 was configured:

```
2024-04-30T20:31:44.176220+00:00 KlasOS %SYS-5-AUTH_LIMIT_ENABLE: Remote authorisation
attempt limit enabled (Max-fail: 3) - CLI initiation
```

If an account becomes locked due to a number of unsuccessful login attempts equaling the number of failures configured above, a log event similar to the following will be displayed in the system log:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[4666]: error: maximum authentication
attempts exceeded for test from 192.168.1.9 port 48416 ssh2 [preauth]
```

The account can be unlocked by a local console administrator using this command from privileged exec mode:

- **`clear aaa remote user username <username>`**

A log event similar to the following will be displayed in the system log:

```
2024-04-30T20:31:44.176220+00:00 KlasOS %SYS-5-AUTH_LIMIT_RESET: Remote authorisation
attempts reset (user: admin) - CLI initiation
```

NOTE: This max-fail attempts number is for consecutive login attempts and is not affected by automatic SSH session termination after its default number of failures. For example, if the lockout failure number is set to 5 and SSH disconnects after 3 failed attempts, if the user then tries to SSH unsuccessfully 2 more times, then that user will be locked out.

To disable account locking just execute the command:

- **`no aaa authentication attempts max-fail`**

NOTE: When a remote user is locked out, the ToE SSH daemon must restart in order to successfully prevent the locked remote user from remotely accessing the ToE again. Any SSH sessions in progress at the time of the SSH daemon restarts will be disconnected.

4. Security Management

The security administrator of the ToE is the **Privileged EXEC mode** user (See Section 1.4.3 'KlasOS'). Only this administrator user can perform the following management functions on the ToE:

- Firmware updates and verification using a digital signature
- Modify the behavior of the TSF with regard to transmission of audit data to external syslog servers
- Modify, delete, generate/import of cryptographic keys
- Ability to configure the ToE locally and remotely
- Ability to configure the access banner
- Ability to configure the session inactivity time before session termination
- Ability to configure the cryptographic functionality
- Start and stop all services
- Ability to set the time which is used for time-stamps
- Ability to re-enable an Administrator account
- Ability to configure the authentication failure parameters
- Ability to configure NTP
- Ability to configure the reference identifier for the peer
- Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors
- Ability to import X.509v3 certificates to the TOE's trust store
- Ability to manage the trusted public keys database

4.1. Services

The following ToE services can be started and stopped by an administrator:

- DNS server
  - Run the following command in Global Configuration mode to enable DNS:
    - **`ip dns server`**
  - Run the following command in Global Configuration mode to disable DNS:
    - **`no ip dns server`**
- NTP server
  - NTP Server is disabled by default and must remain disabled to be in a CC validated state.
  - The following command in Global Configuration mode will also disable NTP if it is found enabled:
    - **`no ntp server <ntp server IP>`**
- HTTPS
  - In configuration mode, run following command:
    - **`(config)# ip http secure-server`**
  - To specify the certificate the trustpoint uses for HTTPS, run the following command:
    - **`(config)# ip http secure-server certmgr <trustpoint>`**
- SSH Client
  - To get an SECSH formatted public key from the TOE, run the following command in privileged EXEC mode:
    - **`Show ip ssh`**

- Remote Syslog
  - To configure the logs to be sent to a syslog server, use the following command in global configuration mode:
    - **`Logging host 127.0.0.1`**

NOTE: Session IDs are enabled by default for HTTPS when that service is turned on. No other configuration is necessary for HTTPS. The TOE only supports TLSv1.2 for all HTTPS connections. All other versions of TLS are rejected. The HTTPS server on the TOE only supports the following algorithms using an RSA key size of 2048, 3072 or 4096 bits:

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

See the Klas Voyager Forums and the Klas Voyager Configuration Guide for further details on configuration of these features (Registration is required for access):

- https://helpdesk.klasgov.com

These services were not evaluated under the Common Criteria process.

Configuration of CC evaluated services are described in this document:

- NTP Server
  - See section 17
- HTTPS Server
  - See section 4.1
- SSH Server
  - See section 7
- SSH Client
  - See section 9
- Remote Syslog
  - See section 8.4

5. Protection of the TSF

The ToE implements policies and checks to ensure that critical security data is protected. All passwords and pre-shared keys can be encrypted on the CLI so that they cannot be viewed in cleartext. Private keys are stored in secure micro-partitions of the flash and are not readable by any user or administrator. The ToE also performs self-tests on bootup. These self-tests verify the integrity of the firmware image and also perform known answer tests against all of the validated cryptographic algorithms. Trusted firmware upgrades using a digital signature are also supported by the ToE (See Section 2.1 'Software Installation') and the ToE provides reliable timestamps using a real-time clock. Traffic cannot traverse the ToE until all configuration has been loaded after booting up.

5.1. Password Protection

All user and administrator passwords are encrypted on the CLI in SHA-512 format (configurable). The number "7A" indicates SHA512 hashing. They are never shown in cleartext. When a username and password are configured (See Section 3.1 'Passwords'), the details are written to the running configuration in a pattern similar to this:

- **`username klas secret 5 $1$lMXgyU2f$GZZ19DKsfkuo3K5jXXo8F/`**

The administrator password would look similar to the following:

- **`enable secret 5 $1$Sm7kgWJq$VIGvygfsvQoxfPGSp1707.`**

Creation or resetting of passwords is logged to the audit log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for creating or resetting a password would look like the following for a non-administrator user:

- `2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2397]: (admin) (klas) (KlasOS) username klas secret : Success`

The log message for creating or resetting the administrator password would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2397]: (admin) (klas) (KlasOS) enable secret : Success`

5.2. Pre-shared Key Protection

Plaintext ephemeral keys are generated for each unique SSH session and are stored in RAM. Persistent keys that are in use are stored in plaintext in RAM. Private keys stored in non-volatile memory are stored in a temporary directory in plaintext. The keys are not accessible or visible to any users or administrators.

5.3. Self-Tests

The ToE performs the following self-tests:

- Integrity check of the firmware image (during bootup)
    - During system boot the ToE performs an integrity check of the installed firmware by comparing the RSA 4096 using SHA-256 digital signature of the firmware image. This happens before any configuration has been loaded or any interfaces are enabled. If signature verification fails, all SSH functionality is disabled and the following messages will be sent to the system log:

`Firmware image RSA signature verification: FAILURE !!! Please install a valid firmware image`

The format of this log message is as follows:

```
[DATE] local0.err %SELF-TEST-ERROR [Self-test Log Message]
```

WARNING: The above error is a result of installation of an invalid firmware image. If this error is seen, then all SSH functionality will be disabled. The administrator must get a valid Klas firmware image (See Section 1.4.2) and install it as per Section 2.1 'Software Installation'.

NOTE: Although all cryptographic operations will be disabled, the administrator will still be allowed to validate the new firmware image against a digital signature.

- FIPS module self-tests in accordance with the OpenSSL 3.0.8 FIPS 140-2 Policy (during bootup)
    - The TOE performs FIPS self-tests to test the integrity of the operational environment when the cryptographic module is first initialized during boot-up. This includes KAT and PCT on all supported algorithms. If any cryptographic self-test fails, the TOE will complete the boot process with all cryptographic functions disabled.

- Entropy self-tests (continuous and during bootup)
    - The entropy noise source health tests are performed during bootup as part of the self-tests. They also are run continuously during system runtime. If any of the entropy health tests fail, the system will reboot immediately and an error message will be displayed to the console.

IMPORTANT: If any cryptographic algorithm known-answer tests or entropy self-tests failures are observed, the user should no longer use the device for cryptographic operations with the current firmware image. The user should try the following:

1. Load a new firmware image and check if the issue still occurs.
2. If the problem continues to exist, please discontinue usage of the device and contact Klas Telecom for assistance.

5.4. Time

The ToE has a real-time clock that can be used as a reliable time source. The system clock can be set using the following command from Privileged EXEC mode:

- **clock set <HH:MM:SS> <MONTH> <DAY> <YEAR>**

Modification of the system time is logged to the system log.

Manual modification of the system clock as per the command above would look similar the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS %SYS-6-CLOCKUPDATE: System clock has been
updated from 12:02:37 UTC Fri May 11 2018 to 12:02:00 UTC Fri May 11 2018 user test at
ttyS0
```

where 'klas' is the current logged in username.

The format of the log message for manually changing the clock is as follows:

[Date and Time] [Hostname or IP address of ToE] [%SYS-6-CLOCKUPDATE]: [log message including old and new time] user <username> at <Source IP address>

6. Cryptographic Key Management

6.1. Cryptographic Key Generation

The ToE can support the generation of one (1) EC/RSA cryptographic keypair as follows in Common Criteria evaluated mode. This keypair is used by both the SSH Server on the ToE for the SSH Host Key (see Section 6.1.1 - SSH Host Key), and the SSH client on the ToE for establishing an SSH tunnel to a remote server (see Section 9 - SSH Tunnel for Trusted Channel):

- EC keys of size 256 or 384
- RSA keys of size 2048, 3072, or 4096

Before keys can be generated, a domain name must be configured on the ToE with the following command entered in global configuration mode:

- **`ip domain-name klas.cc.test`**

Each private key generated is stored on the system flash and each key can be zeroized securely as per Common Criteria requirements.

Generating, importing, modifying or zeroizing cryptographic keys is logged to the audit log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The audit log message for generating a crypto key would look like the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2833]: (admin) (klas) (ttyS0)
crypto key generate rsa general-keys modulus 2048 label rsa_key : Success
```

6.1.1. SSH Host Key

The SSH host key is obtained from the generated keypair and is used for SSH remote administration of the ToE. To see details of existing keys generated, enter the '**show crypto key mypubkey all'** (see Section 6.1.4**)** from privileged EXEC mode. The output will show the key names. To generate a new SSH host key, you need to firstly zeroize any existing keypairs. This can be done using either of the following methods:

- Zeroize the individual key stored in flash:
  - **`crypto key zeroize <ec|rsa> <label name>`**
    - where the <label name> matches the Key name in flash.
- Zeroize all existing keys:
  - **`crypto key zeroize`**

See Section 6.2 for more details on key zeroization. Once the key has been zeroized, a new keypair and SSH host key can be generated as per Section 6.1.2 below.

Deletion of cryptographic keys will also disable the SSH server daemon. This is logged in the system log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for the SSH server disabling would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS %SSH-5-DISABLED: SSH 2.0 has been disabled`

6.1.2. ECDSA Keypair

To generate an ECDSA keypair do the following in global configuration mode:

- **`crypto key generate ec keysize <256|384> label <label name>`**

The <label name> is a unique identifier for the key.

6.1.3. RSA Keypair

To generate an RSA keypair do the following in global configuration mode:

- **`crypto key generate rsa general-keys modulus <2048|3072|4096> label <label name>`**

The <label name> is a unique identifier for the key.

Running the same command again with the same label name will overwrite the existing key with that label name.
The '**show crypto mypubkey all**' command will display details of all existing keys on the ToE. See section 6.1.3 below for details.

Note: The above keys can also be used for DTLS certificate generation. To learn more about how DTLS is used on the TOE, refer to sections 10 & 11.

6.1.4 Information on Crypto Key Generation

Crypto keys are generated in pairs: one private and one public key. If the user repeats the command using the same label name then the old keypair will be zeroized and a new keypair created.

Key-pairs are not stored in the configuration and private keys are not visible by a user or administrator. The administrator can see what keys have been generated by executing the command from privileged EXEC mode:

- **`show crypto key mypubkey <all | rsa | ec>`**
    - This will output details of the generated keys and the public component. The private key will not be displayed.
    - The Key name field corresponds to the label name. The Key type field displays if the key is RSA or ECDSA. The Key storage field displays the secure partition the key is stored in.

- Use **'all'** to display all keypairs. Use **'ec'** to display EC keypairs. Use **'rsa'** to display RSA keys

Keypairs are persistent across reboots as they are stored in secure flash partitions.
If the error **"% Please define a domain-name first.**" is displayed after trying to run the crypto key generate command then the administrator is required to configure a domain-name using the **'ip domain-name <domain-name>'** command from global configuration mode.

Generation of cryptographic keys will also enable the SSH server daemon. This is logged in the system log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for the SSH server enabling would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS %SSH-5-ENABLED: SSH 2.0 has been enabled`

6.2. Cryptographic Key Zeroization

Cryptographic keys can be zeroized using the following methods:

- Using the crypto key zeroize command from global configuration mode:
  - **`crypto key zeroize <rsa | ec>`**
    - Type 'crypto key zeroize' to zeroize all keypairs. Use the 'ec' option to just zeroize EC keypairs. Use the 'rsa' option to just zeroize RSA keys.
- Generating a new key (See section 6.1.2 for ECDSA keys and section 6.1.3 for RSA keys) will overwrite and erase any existing keys

Note: There are no circumstances that may not strictly conform to the key destruction requirements or situations where key destruction may be delayed at the physical layer

Deletion of cryptographic keys is logged to the audit log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for deleting a cryptographic key would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS CLI[5214]: (admin) (test) (ttyS0) crypto key zeroize rsa rsa-key-3072 : Success`

When all keys are zeroized, a message is logged to the system log to uniquely identify the key that was deleted. The log message would look similar to the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS %KEY-6-INFO: KEY SSH has been removed`

Deletion of cryptographic keys will also disable the SSH server daemon. This is logged in the system log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for the SSH server disabling would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS %SSH-5-DISABLED: SSH 2.0 has been disabled`

The SSH server daemon can also be manually disabled and enabled by executing the following commands from global configuration mode:

- `line vty 0 4`
- `no transport input`
  - These commands will disable the SSH server daemon
- `line vty 0 4`
- `transport input ssh`
  - These commands will enable the SSH server daemon

Note: There are no configurations or circumstances that do not strictly conform to the key destruction requirements found in FCS_CKM.4. There are also no situations where the key destruction may be delayed at the physical layer.

7. Remote Administration Using SSH

Remote administration of the device is allowable using SSH. The ToE can be configured to use public-key authentication or password authentication. The default setting is to attempt public-key authentication first and if no SSH public-key is found it will fall back to password authentication.

SSH server on KlasOS supports SSH version 2 only. SSH version 1 is not supported.

SSH server on the ToE is restricted to the following algorithms:

- Encryption using AES-CTR-128, AES-CTR-256, AES-CBC-256 or AES-CBC-128
- Public key user and host authentication using SSH-RSA, ECDSA-SHA2-NISTP256 or ECDSA-SHA2-NISTP384
- Integrity using HMAC-SHA1, HMAC-SHA2-256, or HMAC-SHA2-512
- Key exchange using DIFFIE-HELLMAN-GROUP14-SHA1, ECDH over NIST P256 with SHA2 or ECDH over NIST P384 with SHA2.

NOTE: These algorithms are not configurable on the ToE by an administrator. The algorithm used will depend on the algorithms the SSH client is using and the type of key generated on the ToE and is restricted to the algorithms outlined above. The use of any other cryptographic engines other than those listed above were not evaluated or tested during the CC evaluation of the ToE. The TOE does not require configuration for key size(s) and mode(s) for data encryption/decryption, since it is pre-configured and fixed.

NOTE: The ToE in a validated configuration will only allow an SSH client to use the algorithms specified above. The SSH client on the remote device should be configured so that it ONLY uses the algorithms specified above. On a Linux device, the SSH client settings are configured using the file **ssh_config**. This is usually stored in the **/etc** or **/etc/ssh/** directory. To check what algorithms are used by the SSH connection, the SSH client can be run in debug mode by using the following command on a Linux PC:

- `ssh -v username@a.b.c.d`
    - where username is the username configured on the ToE and a.b.c.d is the IP address of the ToE.

On the PuTTY SSH client, the algorithm settings can be configured using the Connection -> SSH settings.

To enable SSH server on the ToE, do the following:

- Generate a host key. Section 6.1.1 SSH Host Key describes how to do this. **NOTE:** The key-pair stored in flash is the one that will be used as the SSH host key**.**

The following commands can be used to configure SSH authentication-retries (default is 3) and SSH time-out (default is 60 seconds) if required:

- **`ip ssh authentication-retries <number of retries>`**
- **`ip ssh time-out <number of seconds>`**

SSH authentication messages are sent to the ToE system log. See Section 8 'Logging and Auditing' for information on the system log and the format of the system log messages. The ToE will log whenever a user tries to SSH to the device using an algorithm that is not allowed. This is an example of a user trying to SSH to the ToE using the hmac-md5 mac:

- `2024-04-30T20:31:44.176220+00:00 KlasOS sshd[15292]: %SYS-5-SSH_AUTH_FAILURE: Failed to negotiate with 192.168.3.10: no matching MAC found. [preauth]`

The ToE will also log failure to establish an SSH session due to an incorrect password. This would look like the following:

- `2024-04-30T20:31:44.176220+00:00 KlasOS sshd[15292]: %SYS-5-SSH_AUTH_STATUS: Password Authentication Failed by klas on 192.168.3.10`

The format of audit records for failure to establish an SSH session is as follows:

- **[Date and Time] [Hostname or IP address of ToE] process[process id]: [tag]: [SSH failure log message]**

Termination of the SSH trusted path is also logged to the system log.
If the user manually exited the SSH session they would see a message like this:

- `2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-SSH_AUTH_LOGOUT: Session ended by klas on 192.168.3.10`

If the SSH session timed out, a message similar to the following would be displayed in the system log:

- `2024-04-30T20:31:44.176220+00:00 KlasOS [15655]: %SYS-5-SSH_AUTH_TERMINATED: Session Ended for klas on 192.168.3.10`

The ToE SSH server supports the following types of public keys for remote SSH authentication::

- EC keys of size 256 or 384
- RSA keys of size 2048, 3072, or 4096

7.1. Importing a Public Key

To import a public key into the ToE for SSH public key authentication by a remote administrator, first ensure a username is configured. See section 3.1 Passwords on how to configure a username with password.

Once a username has been configured, type the following commands from global configuration mode:

- **ip ssh pubkey-chain**
  - **username <username>**
    - **key-string <SSH public key>**

The <SSH public key> is the full string taken from the SSH client PC public key file.
Once the public key is imported a user can SSH to the ToE without entering a password.

The ToE ensures that a SSH rekey happens after no more than 1 GB of data has been received or after 1 hour, whichever is arrived at first. When a SSH rekey occurs the following message is displayed in the system log:

SSH Client Message:

```
2024-04-30T20:31:44.176220+00:00 KlasOS /usr/bin/ssh[2909]: %SYS-6-SSH_AUTH_REKEY: SSH
rekey completed with x.x.x.x
```

SSH Server Message:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[4615]: %SYS-6-SSH_AUTH_REKEY: SSH rekey
completed from x.x.x.x
```

where x.x.x.x is the IP address of the remote SSH client/server..

The format of the SSH rekey message is:

```
[TAG]: SSH rekey completed from [SSH client IP address]
[TAG]: SSH rekey completed with [SSH server IP address]
```

The administrator can see what SSH sessions are currently active by running the following command from Privileged EXEC mode:

- **`show ssh`**

8. Logging and Auditing

The ToE utilizes the following logs for debugging, auditing, packet filtering and troubleshooting:

- Audit log:
    - This logs every CLI command entered by a user or administrator including:
        - Security related changes
        - Generating/import of, modification or deletion of cryptographic keys
        - Resetting passwords
        - Starting and stopping services
    - Unless otherwise specified in this guidance document, all audit log messages are in the following format:
        - **[Date and Time] [IP address or Hostname of ToE] CLI[process id]: (admin) [user] [Origin of CLI command] [CLI command] : [success|failure]**
    - If a CLI command is entered in Privileged EXEC mode or Global Configuration mode (which are both administrator roles), then the log message has **'(admin)'** before the login username to signify that the action was performed at an administrator level:
        - `2024-04-30T20:31:44.176220+00:00 KlasOS CLI[10476]: (admin) (klas) (KlasOS) exit : Success`
    - If a CLI command was entered in User EXEC mode then the log message is displayed with just the login user name:
        - `2024-04-30T20:31:44.176220+00:00 KlasOS CLI[10457]: (klas) (KlasOS) show clock : Success`
    - The audit log cannot be stopped or disabled by an administrator. It is always on.
    - The locally stored audit log files cannot be modified or deleted by an unauthorized user
- System log:
    - Logs all general system and authentication messages including:
        - User and administrator authentication events for both local and remote sessions
        - Self-test firmware integrity pass/fail messages
        - Clock modification notifications
    - Unless otherwise specified all messages in the system log are in one of the following formats:
        - **[Date and Time] [Hostname or IP address of ToE]:[tag]: [log message] or**

- `[Date and Time] [Hostname or IP address of ToE] process[process id]: [tag]: [log message]`

- The system log cannot be stopped or disabled by an administrator. It is always on.

All ToE logs include the date and time of the event, type of event, subject identity and the outcome of the event. There is a log entry when the file system flash storage is 75% full.

Each log file is rotated at approximately 10MB in size but due to the lag between the appending to the log and the rotation of the log, the size may grow larger than this. Each log will never grow larger than 20MB in size. This is not a configurable option. To check the name and current size of the log files, run the following command in Privileged EXEC mode:

- `show log files`

All of the above logs can be sent to a remote syslog server but this MUST be done over a trusted channel using SSH. See Section 9 for instructions on configuring a SSH trusted channel.

NOTE: If a remote syslog server is configured, all contents of the Audit log and System log are **simultaneously** sent to both the local logs on the ToE and the audit/syslog server.

8.1. Audit Logging

The audit log on the ToE can be displayed using either of the following commands from Privileged EXEC mode on the CLI:

- `show logging audit`
  - This will display the most recent contents of the audit log and then return to CLI prompt

- `show logging audit continuous`
  - This will display the contents of the audit log in real-time and the user must enter Ctrl-c to return to the CLI prompt.
- `show logging audit full`
  - This will display the entire contents of the audit log and then return to the CLI prompt

8.1.1. Starting and Stopping Local Audit Logging

Displaying the audit logs on the local console is disabled by default but can be enabled by the administrator if required. The administrator can enable this using a single command. This is usually not required as the log of each CLI command always gets entered into the audit log which can be displayed by using the 'show logging audit' or 'show logging audit continuous' commands.

To start the audit log for the local console, do the following from Global configuration mode:

- **`logging audit local`**
  - The log event for this would look as follows:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[3330]: (admin) (admin) (ttyS0) logging
audit local : Success
```

To disable local CLI audit logging do the following from Global configuration mode:

- **`no logging audit local`**
  - The log event for this would look as follows:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[3330]: (admin) (admin) (ttyS0) no logging
audit local : Success
```

8.1.2. CLI Command Logging

Every CLI command entered on the ToE is logged in the audit logs. Therefore, all security related configuration changes such as password modification, pre-shared key modification, generation and deletion of cryptographic key-pairs, starting and stopping services etc will all be logged in the audit log.

NOTE: See beginning of Section 8 for details on the format of the CLI audit log messages.

This is an example of the log event generated by the administrator on the local console entering the command "**`show running-config`**":

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2659]: (admin) (klas) (KlasOS) show
running-config : Success
```

This is an example of the log event generated by the administrator from a remote SSH session entering the command "**`show running-config`**":

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2659]: (admin) (klas) (192.168.3.10) show
running-config : Success
```

8.2. System Log

The system log records all general system messages from various services. It also records all attempts by a user to authenticate to the ToE for both local console access and remote SSH access. The system log can be displayed by using the following command in privileged exec mode:

- **`show logging`**
  - This will display the most recent contents of the system log and then return to CLI prompt

- **`show logging continuous`**
  - This will display the contents of the system log in real-time and the user must enter Ctrl-c to return to the CLI prompt.

- **`show logging full`**
  - This will display the entire contents of the system log and then return to CLI prompt

All authentication events are displayed in the log in the following format:

- `[Date and Time] [IP address of ToE] [process]: [tag]: [authentication message detailing the user name and the location of the authentication attempt]`

The logging priority on the local console can be modified so that system log messages can be seen without opening the relevant log files. By default the logging priority is set to critical but can be changed by entering the following command in Global Configuration mode:

- **`logging console debugging`**

To disable logging on the console, enter the following command in Global Configuration mode:

- **`no logging console`**

Logs Generated by the TOE (with examples):

- FAU_GEN.1
  - Startup of audit functions

```
2024-03-05T09:26:30.153648+00:00 VM3.0 CLI[18018]:  (admin) (acumensec) (192.168.228.37) logging audit local : Success
```

  - Shutdown of audit functions

```
2024-03-05T09:27:02.766568+00:00 VM3.0 CLI[18018]:  (admin) (acumensec) (192.168.228.37) no logging audit local : Success
```

  - Administrative login

```
2023-11-03T09:54:04.478743+00:00 VM3.0 sshd[19326]:  Accepted password for acumensec from 10.1.5.106 port 51020 ssh2
2023-11-03T09:54:04.542707+00:00 VM3.0 acumensec:  %SYS-5-PRIV_AUTH_SUCCESS: Session started by acumensec on ttyS0
2023-11-03T09:54:04.587778+00:00 VM3.0 CLI[19404]:  (acumensec) (10.1.5.106) startup : Success
2023-11-03T09:54:08.994755+00:00 VM3.0 su:  %SYS-5-SSH_AUTH_SUCCESS: Authentication to privilege level 15 succeeded by acumensec on (null)
```

  - Administrative logout

```
2023-11-03T09:54:22.013977+00:00 VM3.0 acumensec:  %SYS-5-PRIV_AUTH_LOGOUT: Session ended by acumensec on ttyS0
2023-11-03T09:54:22.015352+00:00 VM3.0 CLI[19404]:  (acumensec) (10.1.5.106) exit : Success
2023-11-03T09:54:22.025139+00:00 VM3.0 sshd[19403]:  Received disconnect from 10.1.5.106 port 51020:11: disconnected by user
2023-11-03T09:54:22.025296+00:00 VM3.0 sshd[19403]:  Disconnected from user acumensec 10.1.5.106 port 51020
```

  - Changes to TSF date related to configuration changes

```
2023-11-03T10:07:14.287773+00:00 VM3.0 sshd[24954]:  Failed password for acumensec from 10.1.5.106 port 58306 ssh2
2023-11-03T10:07:17.786612+00:00 VM3.0 sshd[24954]:  Failed password for acumensec from 10.1.5.106 port 58306 ssh2
2023-11-03T10:07:18.250628+00:00 VM3.0 sshd[24954]:  error: maximum authentication attempts exceeded for acumensec from 10.1.5.106 port 58306 ssh2 [preauth]
2023-11-03T10:07:18.250700+00:00 VM3.0 sshd[24954]:  Disconnecting authenticating user acumensec 10.1.5.106 port 58306: Too many authentication failures [preauth]
2023-11-03T10:07:18.251511+00:00 VM3.0 sshd[24954]:  PAM 1 more authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=10.1.5.106  user=acumensec
```

  - Generating, import, changing or deletion of cryptographic keys

```
2024-03-05T12:14:44.278277+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) configure terminal : Success
2024-03-05T12:15:15.437184+00:00 VM3.0 %KEY-6-INFO:  KEY rsa2048 has been removed
2024-03-05T12:15:15.437335+00:00 VM3.0 %SSH-5-DISABLED:  SSH 2.0 has been disabled
2024-03-05T12:15:15.460613+00:00 VM3.0 sshd[7014]:  Received signal 15; terminating.
2024-03-05T12:15:15.460983+00:00 VM3.0 %SSH-5-DISABLED:  SSH 2.0 has been disabled
2024-03-05T12:15:15.513255+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) crypto key zeroize rsa rsa2048 : Success
2024-03-05T12:15:27.587238+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

```
2024-03-05T12:17:12.818672+00:00 VM3.0 sshd[6287]:  Server listening on :: port 22.
2024-03-05T12:17:12.869812+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) crypto key generate rsa general-keys modulus 2048 label RSA_2048 : Success
2024-03-05T12:17:14.850257+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

- Resetting Passwords

```
2023-11-03T10:41:40.864183+00:00 VM3.0 CLI[7207]:  (admin) (root123) (10.1.5.106) configure terminal : Success
2023-11-03T10:42:16.524321+00:00 VM3.0 CLI[7207]:  (admin) (root123) (10.1.5.106) username root123 algorithm-type sha512 secret : Success
2023-11-03T10:42:18.671902+00:00 VM3.0 CLI[7207]:  (admin) (root123) (10.1.5.106) exit : Success
```

- FAU_STG_EXT.3/LocSpace
  - Low storage space for audit events

2024-03-11T15:10:00.271762+00:00 VM3.0 CLI: File system flash 75% full

- FCS_DTLSC_EXT.1
  - Failure to establish a DTLS session

```
2023-07-27T12:19:14.217267+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T12:19:14.217351+00:00 VM3.0 %SDWAN0-PROC:  [INFO/config] SDWAN DTLS initialization success
2023-07-27T12:19:14.217436+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map0_Eth0/0 bind to 10.1.5.142
2023-07-27T12:19:14.217520+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map1_Eth0/0 bind to 10.1.5.142
2023-07-27T12:19:14.217605+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map2_vSwitch101 bind to 192.168.101.1
2023-07-27T12:19:14.217690+00:00 VM3.0 %SDWAN0-PROC:  [WARN/protocol] DTLS warning on map0_Eth0/0 - cannot connect to the peer - Connection refused(
111)
```

- FCS_DTLSS_EXT.1
  - Failure to establish a DTLS session

2024-02-05T21:55:26.777252+00:00 VM3.0 %SDWAN0-PROC: [WARN/protocol] DTLS protocol error on map1_Eth0/1 - no shared cipher
2024-02-05T21:55:26.777339+00:00 VM3.0 %SDWAN0-PROC: [INFO/protocol] map1_Eth0/1 closing read/write socket
2024-02-05T21:55:26.777422+00:00 VM3.0 %SDWAN0-PROC: [INFO/protocol] map1_Eth0/1 > tunnel down

  - Detected Replay Attacks

```
2024-02-22T07:59:58.040688+00:00 VM3.0 %SDWAN0-PROC:  [INFO] created interface `sdwan0'
2024-02-22T07:59:58.040762+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] DTLS device certificate:
2024-02-22T07:59:58.040836+00:00 VM3.0 %SDWAN0-PROC:  [WARN/dtls] valid ext key usage server auth
2024-02-22T07:59:58.040927+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] ExtKeyUsage check successful
2024-02-22T07:59:58.041005+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 0 s:/C=US/O=Acumen/OU=CC/CN=10.1.3.142
2024-02-22T07:59:58.041079+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=us/O=acumen/OU=cc/CN=DTLSS-ICA
2024-02-22T07:59:58.041153+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 1 s:/C=us/O=acumen/OU=cc/CN=DTLSS-ICA
2024-02-22T07:59:58.041228+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041302+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 2 s:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041375+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041449+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] DTLS Trusted certificates:
2024-02-22T07:59:58.041524+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 0 s:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041598+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041672+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 1 s:/C=us/O=acumen/OU=cc/CN=DTLSS-ICA
2024-02-22T07:59:58.041745+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=us/O=acumen/OU=cc/CN=DTLSS-CA
2024-02-22T07:59:58.041819+00:00 VM3.0 %SDWAN0-PROC:  [INFO/config] SDWAN DTLS initialization success
2024-02-22T07:59:58.041916+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map0_Eth0/1 bind to 10.1.3.142
2024-02-22T07:59:58.041993+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map1_Eth0/1 bind to 10.1.3.142
2024-02-22T07:59:58.793898+00:00 VM3.0 %SDWAN0:  INFO, sdwan0 process poll check completed
```

```
2024-02-22T08:00:01.627965+00:00 VM3.0 kernel: [139024.542767] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=66
OS=0x00 PREC=0x00 TTL=64 ID=29768 DF PROTO=UDP SPT=60145 DPT=5001 LEN=46
2024-02-22T08:00:01.628999+00:00 VM3.0 kernel: [139024.543810] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=67
OS=0x00 PREC=0x00 TTL=64 ID=29769 DF PROTO=UDP SPT=60145 DPT=5001 LEN=47
2024-02-22T08:00:01.629024+00:00 VM3.0 kernel: [139024.543846] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=233
TOS=0x00 PREC=0x00 TTL=64 ID=29762 DF PROTO=UDP SPT=60145 DPT=5001 LEN=213
2024-02-22T08:00:01.630275+00:00 VM3.0 kernel: [139024.545088] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=253
TOS=0x00 PREC=0x00 TTL=64 ID=29763 DF PROTO=UDP SPT=60145 DPT=5001 LEN=233
2024-02-22T08:00:01.637717+00:00 VM3.0 kernel: [139024.552528] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=153
 TOS=0x00 PREC=0x00 TTL=64 ID=29764 DF PROTO=UDP SPT=60145 DPT=5001 LEN=1480
2024-02-22T08:00:01.637743+00:00 VM3.0 kernel: [139024.552576] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=153
 TOS=0x00 PREC=0x00 TTL=64 ID=29765 DF PROTO=UDP SPT=60145 DPT=5001 LEN=1480
2024-02-22T08:00:01.640219+00:00 VM3.0 kernel: [139024.555033] "%SEC-5-ACL: 101 ACC dtlscin"IN=eth1 OUT= MAC=00:13:f2:0f:20:a1:00:50:56:8b:05:8c:08:00 SRC=10.1.3.179 DST=10.1.3.142 LEN=497
TOS=0x00 PREC=0x00 TTL=64 ID=29766 DF PROTO=UDP SPT=60145 DPT=5001 LEN=477
```

Note: A successful DTLS connection shows that TOE does not take action in response to retransmitted traffic.

- FCS_DTLSC_EXT.2
    - Detected replay attacks

<u>Refer to the logs found above**</u>**

- FCS_DTLSS_EXT.2
    - Failure to authenticate the client

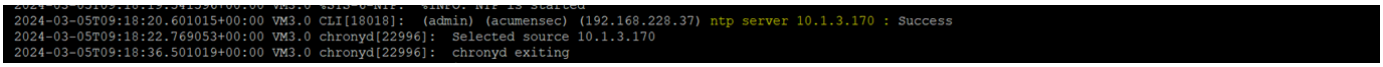
```
2023-07-27T12:19:14.217267+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T12:19:14.217351+00:00 VM3.0 %SDWAN0-PROC:  [INFO/config] SDWAN DTLS initialization success
2023-07-27T12:19:14.217436+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map0_Eth0/0 bind to 10.1.5.142
2023-07-27T12:19:14.217520+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map1_Eth0/0 bind to 10.1.5.142
2023-07-27T12:19:14.217605+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map2_vSwitch101 bind to 192.168.101.1
2023-07-27T12:19:14.217690+00:00 VM3.0 %SDWAN0-PROC:  [WARN/protocol] DTLS warning on map0_Eth0/0 - cannot connect to the peer - Connection refused(
111)
```

- FCS_HTTPS_EXT.1
    - Failure to establish a HTTPS session

```
er www.klasgroup.com:443)
 2023-10-05T14:11:51.340573+00:00 VM3.0 httpd:  info ssl [Thu Oct 05 14:11:51 2023] - SSL Library Error: error:0A0000C1:SSL routines::no shared cipher -- Could not con
ect, no matching algorithms
 2023-10-05T14:11:51.340588+00:00 VM3.0 httpd:  [10.1.5.206:54614] [10.1.5.142:443] info ssl [Thu Oct 05 14:11:51 2023] - AH01998: Connection closed to child 66 with ab
rtive shutdown (server www.klasgroup.com:443)
 2023-10-05T14:11:51.345331+00:00 VM3.0 httpd:  [10.1.5.206:54628] [10.1.5.142:443] info ssl [Thu Oct 05 14:11:51 2023] - AH01964: Connection to child 4 established (se
ver www.klasgroup.com:443)
 2023-10-05T14:11:51.345854+00:00 VM3.0 httpd:  [10.1.5.206:54628] [10.1.5.142:443] info ssl [Thu Oct 05 14:11:51 2023] - AH02008: SSL library error 1 in handshake (ser
er www.klasgroup.com:443)
 2023-10-05T14:11:51.345882+00:00 VM3.0 httpd:  info ssl [Thu Oct 05 14:11:51 2023] - SSL Library Error: error:0A0000C1:SSL routines::no shared cipher -- Could not con
ect, no matching algorithms
```

- FCS_NTP_EXT.1

- Configuration of a new time server



- Removal of configured time server

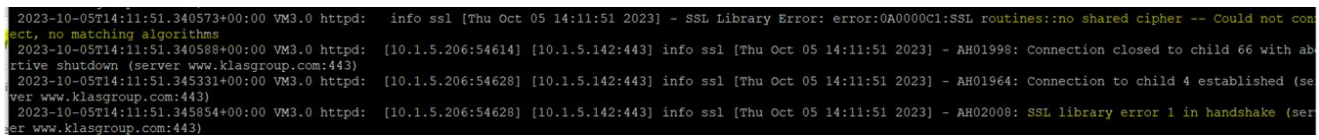
- FCS_SSHS_EXT.1
  - Failure to establish an SSH session

2023-06-08T22:30:18.504024+00:00 VM3.0 sshd[3111]: %SYS-5-SSH_AUTH_FAILURE: Failed to negotiate with 10.1.5.106: no matching MAC found. [preauth]
2023-06-08T22:30:18.504065+00:00 VM3.0 sshd[3111]: %SYS-5-SSH_AUTH_FAILURE: Ciphers the host 10.1.5.106 supports "hmac-md5" [preauth]
2023-06-08T22:30:18.504090+00:00 VM3.0 sshd[3111

- FCS_SSHC_EXT.1
  - Failure to establish an SSH session

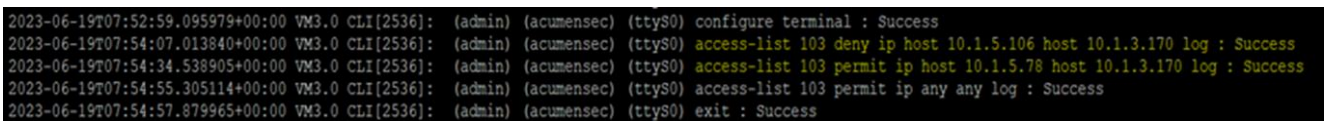
2024-02-23T22:42:14.876707+00:00 TRX-R2 ssh[18808]: Unable to negotiate with 10.1.3.170 port 22: no matching MAC found. Their offer: hmac-md5

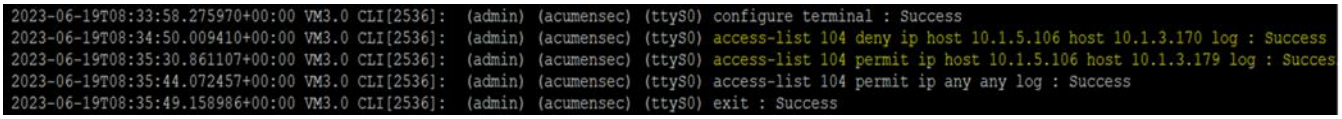
- FCS_TLSS_EXT.1
  - Failure to establish a TLS session



- FFW_RUL_EXT.1
  - Application of rules configured with the 'log' operation

Source of destination addresses





Source and Destination Ports

```
2023-06-19T12:47:07.682199+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 103 deny tcp host 10.1.5.106 eq 1200 host 10.1.3.170 log : Success
2023-06-19T12:47:19.090239+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 103 permit tcp host 10.1.5.106 eq 1500 host 10.1.3.170 log : Success
2023-06-19T12:47:37.648362+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 103 permit ip any any log : Success
2023-06-19T12:47:39.786983+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) exit : Success
 VM3.0#
```

```
2023-06-19T13:14:03.341290+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 104 deny tcp host 10.1.5.106 host 10.1.3.170 eq 1300 log : Success
2023-06-19T13:14:24.096056+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 104 permit tcp host 10.1.5.106 host 10.1.3.170 eq 1600 log : Success
2023-06-19T13:14:36.736970+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) access-list 104 permit ip any any log : Success
2023-06-19T13:14:40.730954+00:00 VM3.0 CLI[2536]:  (admin) (acumensec) (ttyS0) exit : Success
VM3.0#
```

Transport Layer Protocol

```
2024-03-06T14:02:23.821042+00:00 VM3.0 CLI[7378]:  (admin) (acumensec) (192.168.228.37) access-list 105 deny udp host 10.1.5.106 host 10.1.3.170 log : Success
2024-03-06T14:03:20.533576+00:00 VM3.0 CLI[7378]:  (admin) (acumensec) (192.168.228.37) access-list 105 permit tcp host 10.1.5.106 host 10.1.3.170 log : Success
2024-03-06T14:03:32.259151+00:00 VM3.0 CLI[7378]:  (admin) (acumensec) (192.168.228.37) access-list 105 permit ip any any log : Success
2024-03-06T14:03:33.757415+00:00 VM3.0 CLI[7378]:  (admin) (acumensec) (192.168.228.37) exit : Success
```

```
2023-10-23T15:18:30.306223+00:00 VM3.0 CLI[2681]:  (admin) (acumensec) (ttyS0) configure terminal : Success
2023-10-23T15:19:14.879025+00:00 VM3.0 CLI[2681]:  (admin) (acumensec) (ttyS0) access-list 103 deny ipv6 udp host 2022:10:1:5::106 host 2022:10:1:3::170 log : Success
2023-10-23T15:20:00.052925+00:00 VM3.0 CLI[2681]:  (admin) (acumensec) (ttyS0) access-list 103 permit ipv6 tcp host 2022:10:1:5::106 host 2022:10:1:3::170 log : Success
2023-10-23T15:20:18.514253+00:00 VM3.0 CLI[2681]:  (admin) (acumensec) (ttyS0) access-list 103 permit ipv6 ip any any log : Success
2023-10-23T15:20:20.649229+00:00 VM3.0 CLI[2681]:  (admin) (acumensec) (ttyS0) exit : Success
```

- FIA_AFL.1
    - Unsuccessful login attempts limit is met or exceeded

```
2022-07-12T15:59:00.981671+00:00 VM3.0 sshd[25495]:  Failed password for nonadmin from 10.1.3.169 port 41671 ssh2
2022-07-12T15:59:11.123537+00:00 VM3.0 sshd[25495]:  message repeated 2 times: [ Failed password for nonadmin from 10.1.3.169 por
t 41671 ssh2]
2022-07-12T15:59:11.560949+00:00 VM3.0 sshd[25495]:  error: maximum authentication attempts exceeded for nonadmin from 10.1.3.169
 port 41671 ssh2 [preauth]
2022-07-12T15:59:11.561052+00:00 VM3.0 sshd[25495]:  Disconnecting authenticating user nonadmin 10.1.3.169 port 41671: Too many a
uthentication failures [preauth]
```

- FIA_UIA_EXT.1
    - All use of identification and authentication mechanism

2023-11-03T17:14:53.897064+00:00 VM3.0 sshd[29242]: %SYS-6-SSH_AUTH_REKEY: SSH rekey completed with 10.1.3.169 [preauth]
2023-11-03T17:14:58.203957+00:00 VM3.0 sshd[29242]: Accepted password for acumensec from 10.1.3.169 port 53252 ssh2
2023-11-03T17:14:58.264878+00:00 VM3.0 acumensec: %SYS-5-PRIV_AUTH_SUCCESS: Session started by acumensec on ttyS0
2023-11-03T17:14:58.308931+00:00 VM3.0 CLI[29265]: (acumensec) (10.1.3.169) startup : Success

- FIA_UAU_EXT.2
    - All use of identification and authentication mechanism

2023-11-03T17:29:02.501427+00:00 VM3.0 sshd[2844]: %SYS-6-SSH_AUTH_REKEY: SSH rekey completed with 10.1.3.169 [preauth]
2023-11-03T17:29:07.838018+00:00 VM3.0 sshd[2844]: Failed password for acumensec from 10.1.3.169 port 33674 ssh2

- FIA_X509_EXT.1/Rev

- Unsuccessful attempt to validate a certificate

2023-07-22T02:22:20.388681+00:00 VM3.0 %CERTMGR-ERROR: Certificate has expired! Please update the Certificate VM3-expire.crt

- Any addition, replacement, or removal of trust anchors in the TOE's trust store

```
2024-03-05T12:10:57.946741+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr trustpoint DTLSS_ROOTCA RSA : Success
2024-03-05T12:11:07.494506+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) no ca-cert-chain flash: : Success
```

```
2024-03-05T12:10:57.946741+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr trustpoint DTLSS_ROOTCA RSA : Success
2024-03-05T12:11:07.494506+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) no ca-cert-chain flash: : Success
2024-03-05T12:11:23.296821+00:00 VM3.0 %CERTMGR:  Loading Certificate file: ICA_Server.pem
2024-03-05T12:11:23.339455+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: cert #0
2024-03-05T12:11:23.339559+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Version: 3 (0x2)
2024-03-05T12:11:23.339634+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T12:11:23.339708+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Issuer: CN=SharedICA
2024-03-05T12:11:23.339779+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Subject: CN=ICA.Server
2024-03-05T12:11:23.339864+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Validity: Not Before: Feb 13 13:18:00 2024 GMT  Not After : Feb 13 13:12:00 2034 GMT
2024-03-05T12:11:23.339947+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: CA:TRUE
2024-03-05T12:11:23.340020+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: X509v3 Key Usage:  Digital Signature, Non Repudiation, Key Encipherment, Certificate Sign, CRL Sign
2024-03-05T12:11:23.340316+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: X509v3 Extended Key Usage:  OCSP Signing
2024-03-05T12:11:23.340427+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: cert #1
2024-03-05T12:11:23.340669+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Version: 3 (0x2)
2024-03-05T12:11:23.340873+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T12:11:23.341054+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Issuer: CN=RootOCSP
2024-03-05T12:11:23.341131+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Subject: CN=SharedICA
2024-03-05T12:11:23.341202+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Validity: Not Before: Feb 13 13:14:00 2024 GMT  Not After : Feb 13 13:12:00 2034 GMT
2024-03-05T12:11:23.341279+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: CA:TRUE
2024-03-05T12:11:23.341349+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: X509v3 Key Usage:  Digital Signature, Non Repudiation, Key Encipherment, Certificate Sign, CRL Sign
2024-03-05T12:11:23.341419+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: X509v3 Extended Key Usage:  OCSP Signing
2024-03-05T12:11:23.341495+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: cert #2
2024-03-05T12:11:23.341568+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Version: 3 (0x2)
2024-03-05T12:11:23.341637+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T12:11:23.341707+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Issuer: CN=RootOCSP
2024-03-05T12:11:23.341776+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Subject: CN=RootOCSP
2024-03-05T12:11:23.341861+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: Validity: Not Before: Feb 13 13:12:00 2024 GMT  Not After : Feb 13 13:12:00 2034 GMT
2024-03-05T12:11:23.341941+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: CA:TRUE
2024-03-05T12:11:23.342013+00:00 VM3.0 %CERTMGR:  %INFO: ICA_Server.pem: X509v3 Key Usage:  Digital Signature, Non Repudiation, Key Agreement, Certificate Sign, CRL Sign
2024-03-05T12:11:23.405973+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) ca-cert-chain flash: ICA_Server.pem : Success
2024-03-05T12:11:25.974249+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

- FMT_MOF.1/ManualUpdate
  - Any attempt to initiate a manual update

2023-09-22T22:03:53.670419+00:00 VM3.0 CLI[3036]: (admin) (acumensec) (ttyS0) boot system flash KlasOS.keel.v5.4.0rc3.bin : Success

- FMT_SMF.1
  - All management activities of TSF data

Ability to administer the TOE remotely

2023-11-03T17:14:58.203957+00:00 VM3.0 sshd[29242]: Accepted password for acumensec from 10.1.3.169 port 53252 ssh2
2023-11-03T17:14:58.308931+00:00 VM3.0 CLI[29265]: (acumensec) (10.1.3.169) startup : Success

Ability to administer the TOE locally

2024-03-05T10:11:14.973527+00:00 VM3.0 acumensec: %SYS-5-PRIV_AUTH_SUCCESS: Session started by acumensec on ttyS0

Ability to configure the access banner

```
2022-07-20T18:09:06.592397+00:00 VM3.0 CLI[8035]:  (admin) (anotheruser) (10.1.5.106) configure terminal : Succes
s
2022-07-20T18:09:57.814072+00:00 VM3.0 CLI[8035]:  (admin) (anotheruser) (10.1.5.106) banner login "z This is the
 LOGIN banner. WARNING: Authorized users only. z" : Success
2022-07-20T18:10:03.485889+00:00 VM3.0 CLI[8035]:  (admin) (anotheruser) (10.1.5.106) do exit : Success
```

Ability to configure the session inactivity timer before session termination or locking

```
2022-07-15T20:11:03.327903+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) startup : Success
2022-07-15T20:11:23.085487+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) configure terminal : Success
2022-07-15T20:11:30.222910+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) line console 0 : Success
2022-07-15T20:11:40.767970+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) exec-timeout 1 0 : Success
2022-07-15T20:11:45.538457+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) exit : Success
2022-07-15T20:12:47.047541+00:00 VM3.0 :  %SYS-5-PRIV_AUTH_TIMEOUT: Session for user consoleu timed out on ttyS0
2022-07-15T20:11:49.298459+00:00 VM3.0 CLI[31272]:  (admin) (consoleuser) (ttyS0) exit : Success
```

Ability to update the TOE, and to verify the updates using digital signature capability prior to installing those updates

2023-09-22T20:46:08.212178+00:00 VM3.0 %SIG_VER: Verifying boot image file
2023-09-22T20:46:11.005208+00:00 VM3.0 %IMG_VER: Good signature from "Klas Telecom (Klas Signed Image Key) <trxsupport@klastelecom.com>" [unknown]
2023-09-22T20:46:11.038299+00:00 VM3.0 CLI[23687]: (admin) (acumensec) (ttyS0) verify /bootimgver KlasOS.keel.v5.4.0rc3.bin : Success
2023-10-06T15:35:21.129402+00:00 VM3.0 CLI[24903]: (admin) (acumensec) (ttyS0) boot system flash KlasOS.keel.v5.4.0rc3-changed.bin : Success

Ability to configure the authentication failure parameters for FIA_AFL.1;

```
2022-07-12T15:53:59.489920+00:00 VM3.0 CLI[24963]:  (admin) (nonadmin) (10.1.3.1
69) startup : Success
2022-07-12T15:54:09.948130+00:00 VM3.0 CLI[24963]:  (admin) (nonadmin) (10.1.3.1
69) configure terminal : Success
2022-07-12T15:54:22.342499+00:00 VM3.0 %SYS-5-AUTH_LIMIT_ENABLE:  Remote authori
sation attempt limit enabled (Max-fail: 3) - CLI initiation
2022-07-12T15:54:22.348267+00:00 VM3.0 CLI[24963]:  (admin) (nonadmin) (10.1.3.1
69) aaa authentication attempts max-fail 3 : Success
```

Ability to start and stop services;

```
2023-11-03T09:29:39.621792+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) startup : Success
2023-11-03T09:29:43.293216+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) configure terminal : Success
2023-11-03T09:29:54.528555+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) logging host 10.1.3.169 : Success
2023-11-03T09:29:57.076441+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) exit : Success
```

```
2023-11-03T09:30:00.570786+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) configure terminal : Success
2023-11-03T09:30:09.571280+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) no logging host 10.1.3.169 : Success
2023-11-03T09:30:12.233084+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) exit : Success
```

Ability to modify the behaviour of the transmission of audit data to an external IT entity;

```
2023-11-03T09:29:39.621792+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) startup : Success
2023-11-03T09:29:43.293216+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) configure terminal : Success
2023-11-03T09:29:54.528555+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) logging host 10.1.3.169 : Success
2023-11-03T09:29:57.076441+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) exit : Success
```

```
2023-11-03T09:30:00.570786+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) configure terminal : Success
2023-11-03T09:30:09.571280+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) no logging host 10.1.3.169 : Success
2023-11-03T09:30:12.233084+00:00 VM3.0 CLI[9032]:  (admin) (acumensec) (10.1.5.106) exit : Success
```

Ability to manage the cryptographic keys;

```
2022-07-14T20:37:54.926945+00:00 VM3.0 CLI[7452]:  (admin) (anotheruser) (ttyS0) crypto key gene
rate rsa general-keys modulus 2048 label vm3 : Success
```

Ability to configure the cryptographic functionality;

```
2024-03-05T10:40:12.828607+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr generate CSR trustpoint DTLSS_ROOTCA  10.1.3.142_2048.pem : Success
```

Ability to re-enable an Administrator account;

```
2022-07-20T17:34:38.400150+00:00 VM3.0 %SYS-5-AUTH_LIMIT_RESET:  Remote authorisation attempts reset (user: anotheruser) - CLI initiation
2022-07-20T17:34:38.415319+00:00 VM3.0 CLI[4035]:  (admin) (anotheruser) (ttyS0) clear aaa remote user username anotheruser : Success
```

Ability to set the time which is used for time-stamps;

2023-09-06T23:50:00.023110+00:00 VM3.0 %SYS-6-CLOCKUPDATE: System clock has been updated from 22:50:01 UTC Wed Sep 06 2023 to 23:50:00 UTC Wed Sep 06 2023 user acumensec at ttyS0
2023-09-06T22:47:36.258368+00:00 VM3.0 CLI[7757]: (admin) (acumensec) (ttyS0) show clock : Success
2023-09-06T23:50:00.088550+00:00 VM3.0 CLI[7757]: (admin) (acumensec) (ttyS0) clock set 18:50:00 9 6 2023 : Success

Ability to configure NTP;

2021-09-06T15:00:55.032945+00:00 VM3.0 %SYS-6-NTP: %INFO: NTP is started
2021-09-06T15:00:56.123397+00:00 VM3.0 CLI[2261]: (admin) (acumensec) (ttyS0) ntp server 10.1.3.169 : Success
2021-09-06T15:00:58.242434+00:00 VM3.0 chronyd[13520]: Selected source 10.1.3.169
2021-09-06T15:00:58.242489+00:00 VM3.0 chronyd[13520]: System clock wrong by 63171931.884058 seconds, adjustment started

Ability to configure the reference identifier for the peer;

```
2024-03-05T10:21:29.111251+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) configure terminal : Success
2024-03-05T10:21:55.257309+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr trustpoint DTLSS_ROOTCA RSA : Success
2024-03-05T10:22:37.079361+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) reference-id CN 10.1.3.170 : Success
2024-03-05T10:22:52.062271+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

Ability to manage the TOE's trust store and designate X509.v3 certificates as trust anchors;

```
2024-03-05T10:44:16.197243+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) configure terminal : Success
2024-03-05T10:44:17.063448+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr trustpoint DTLSS_ROOTCA RSA : Success
2024-03-05T10:44:46.702206+00:00 VM3.0 %CERTMGR:  Loading Certificate file: DTLSS-CAICA.pem
2024-03-05T10:44:46.742027+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: cert #0
2024-03-05T10:44:46.742131+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Version: 3 (0x2)
2024-03-05T10:44:46.742208+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T10:44:46.742284+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Issuer: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.742358+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Subject: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.742429+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Validity: Not Before: Feb 13 11:33:00 2024 GMT  Not After : Feb 13 11:33:00 2025 GMT
2024-03-05T10:44:46.742501+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: CA:TRUE
2024-03-05T10:44:46.742572+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign
2024-03-05T10:44:46.742644+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:44:46.742723+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: cert #1
2024-03-05T10:44:46.742815+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Version: 3 (0x2)
2024-03-05T10:44:46.742913+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T10:44:46.742986+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Issuer: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.743055+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Subject: C=us, O=acumen, OU=cc, CN=DTLSS-ICA
2024-03-05T10:44:46.743125+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Validity: Not Before: Feb 13 11:38:00 2024 GMT  Not After : Feb 13 11:33:00 2025 GMT
2024-03-05T10:44:46.743194+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: CA:TRUE
2024-03-05T10:44:46.743264+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign
2024-03-05T10:44:46.743334+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:44:46.804663+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) ca-cert-chain flash: DTLSS-CAICA.pem : Success
2024-03-05T10:44:56.353994+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) no ca-cert-chain flash: : Success
2024-03-05T10:44:58.897270+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

Ability to import X.509v3 certificates to the TOE's trust store:

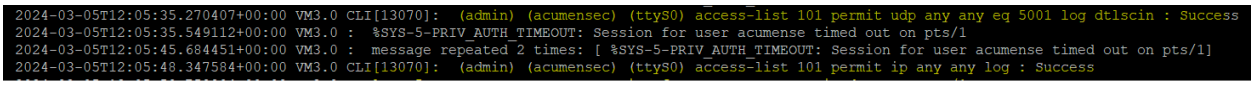

```
2024-03-05T10:44:16.197243+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) configure terminal : Success
2024-03-05T10:44:17.063448+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) certmgr trustpoint DTLSS_ROOTCA RSA : Success
2024-03-05T10:44:46.702206+00:00 VM3.0 %CERTMGR:  Loading Certificate file: DTLSS-CAICA.pem
2024-03-05T10:44:46.742027+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: cert #0
2024-03-05T10:44:46.742131+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Version: 3 (0x2)
2024-03-05T10:44:46.742208+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T10:44:46.742284+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Issuer: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.742358+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Subject: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.742429+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Validity: Not Before: Feb 13 11:33:00 2024 GMT  Not After : Feb 13 11:33:00 2025 GMT
2024-03-05T10:44:46.742501+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: CA:TRUE
2024-03-05T10:44:46.742572+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign
2024-03-05T10:44:46.742644+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:44:46.742723+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: cert #1
2024-03-05T10:44:46.742815+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Version: 3 (0x2)
2024-03-05T10:44:46.742913+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T10:44:46.742986+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Issuer: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:44:46.743055+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Subject: C=us, O=acumen, OU=cc, CN=DTLSS-ICA
2024-03-05T10:44:46.743125+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: Validity: Not Before: Feb 13 11:38:00 2024 GMT  Not After : Feb 13 11:33:00 2025 GMT
2024-03-05T10:44:46.743194+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: CA:TRUE
2024-03-05T10:44:46.743264+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign
2024-03-05T10:44:46.743334+00:00 VM3.0 %CERTMGR:  %INFO: DTLSS-CAICA.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:44:46.804663+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) ca-cert-chain flash: DTLSS-CAICA.pem : Success
2024-03-05T10:44:56.353994+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) no ca-cert-chain flash: : Success
2024-03-05T10:44:58.897270+00:00 VM3.0 CLI[13070]:  (admin) (acumensec) (ttyS0) exit : Success
```

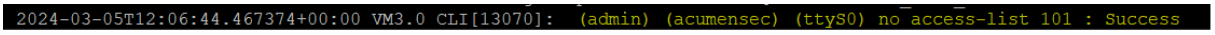

Ability to manage the trusted public keys database:

```
2024-03-05T10:51:26.331423+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement
2024-03-05T10:51:26.331492+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:51:26.331568+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: cert #1
2024-03-05T10:51:26.331640+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: Version: 3 (0x2)
2024-03-05T10:51:26.331709+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: Serial Number: Signature Algorithm: sha256WithRSAEncryption
2024-03-05T10:51:26.331777+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: Issuer: C=us, O=acumen, OU=cc, CN=DTLSS-CA
2024-03-05T10:51:26.331857+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: Subject: C=us, O=acumen, OU=cc, CN=DTLSS-ICA
2024-03-05T10:51:26.331935+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: Validity: Not Before: Feb 13 11:38:00 2024 GMT  Not After : Feb 13 11:33:00 2025 GMT
2024-03-05T10:51:26.332004+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: CA:TRUE
2024-03-05T10:51:26.332074+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: X509v3 Key Usage:  Digital Signature, Key Encipherment, Data Encipherment, Key Agreement, Certificate Sign
2024-03-05T10:51:26.332142+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: X509v3 Extended Key Usage:  TLS Web Server Authentication, TLS Web Client Authentication
2024-03-05T10:51:26.332217+00:00 VM3.0 %CERTMGR:  %INFO: 10.1.3.142.pem: cert #2
```

- FMT_SMF.1/FFW
    - All management activities of TSF data (including creation, modification and deletion of firewall rules

```
VM3.0(config)# access-list 101 permit udp any any eq 5001 log dtlscin
VM3.0(config)# access-list 101 permit ip any any log
```





- FPT_STM_EXT.1
  - Discontinuous changes to time - either Administrator actuated or changed via an automated process

2023-09-06T23:50:00.023110+00:00 VM3.0 %SYS-6-CLOCKUPDATE: System clock has been updated from 22:50:01 UTC Wed Sep 06 2023 to 23:50:00 UTC Wed Sep 06 2023 user acumensec at ttyS0
2023-09-06T22:47:36.258368+00:00 VM3.0 CLI[7757]: (admin) (acumensec) (ttyS0) show clock : Success
2023-09-06T23:50:00.088550+00:00 VM3.0 CLI[7757]: (admin) (acumensec) (ttyS0) clock set 18:50:00 9 6 2023 : Success

- FPT_TUD_EXT.1
  - Initiation of update; result of the update attempt (success or failure)

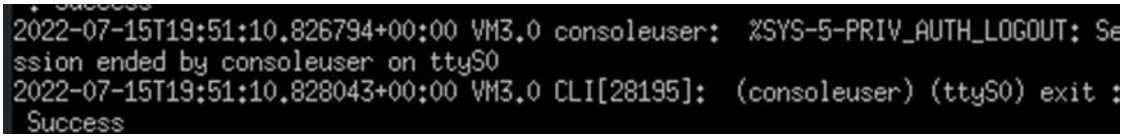
Success
2023-10-06T17:27:20.016568+00:00 VM3.0 CLI[2615]: (admin) (acumensec) (ttyS0) boot system flash KlasOS.keel.v5.4.0-bad-sig.bin : Success

Failure
2023-10-06T17:22:10.604961+00:00 VM3.0 %SIG_VER: Verifying boot image file
2023-10-06T17:22:14.237495+00:00 VM3.0 %IMG_VER_FAIL: The signature could not be verified.
2023-10-06T17:22:14.277175+00:00 VM3.0 CLI[2615]: (admin) (acumensec) (ttyS0) verify /bootimgver KlasOS.keel.v5.4.0-bad-sig.bin : Success

- FTA_SSL_EXT.1
  - The termination of a local session by the session locking mechanism

- FTA_SSL.3
  - The termination of a remote session by the session locking mechanism

- FTA_SSL.4
  - The termination of an interactive session



- FTP_ITC.1 (SSH)

- Initiation of the trusted channel

2024-02-26T20:55:59.297643+00:00 VM3.0 ssh[2920]: %SYS-6-SSH: Authenticating to 10.1.3.170:22 as 'acumensec'
2024-02-26T20:55:59.323828+00:00 VM3.0 ssh[2920]: %SYS-6-SSH_AUTH_REKEY: SSH rekey completed with 10.1.3.170
2024-02-26T20:55:59.349288+00:00 VM3.0 ssh[2920]: %SYS-6-SSH: Authenticated to 10.1.3.170 ([10.1.3.170]:22).

- Termination of the trusted channel

2023-12-28T21:38:58.685412+00:00 VM3.0 ssh[29329]: %SYS-6-SSH: Connection to 10.1.3.170 closed by remote host.

- Failure of the trusted channel functions

2024-03-15T15:08:33.826363+00:00 VM3.0 ssh[21478]: %SYS-5-SSH_AUTH_FAILURE: Failed to negotiate with 10.1.3.170: no matching host key type found.

- FTP_ITC.1 (DTLS)
  - Initiation of the trusted channel

```
2023-07-27T11:40:40.576547+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] map1_Eth0/0 peer chain cert-
2023-07-27T11:40:40.576632+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 1 s:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T11:40:40.576717+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T11:40:40.576802+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] map1_Eth0/0 peer chain cert-
2023-07-27T11:40:40.576905+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 0 s:/C=US/O=Acumen/OU=CC/CN=10.1.3.179
2023-07-27T11:40:40.576992+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T11:40:40.577077+00:00 VM3.0 %SDWAN0-PROC:  [WARN/dtls] valid ext key usage server auth
2023-07-27T11:40:40.577162+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] ExtKeyUsage check successful
2023-07-27T11:40:40.577248+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] Basic OCSP validation policy configured, skipping this process for
peer-cert, continuing with session setup
2023-07-27T11:40:40.577332+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] Basic identifier match policy configured, skipping this process for
 peer-cert, continuing with session setup
2023-07-27T11:40:40.577417+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] map1_Eth0/0 > new DTLS session created
```

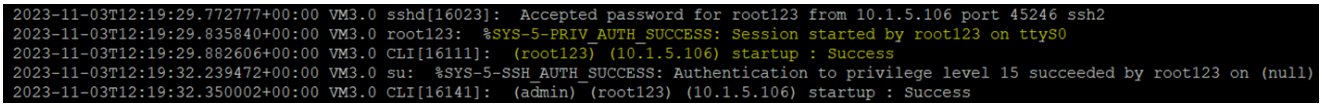
  - Termination of the trusted channel

```
2023-11-29T07:07:49.551665+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] map1_Eth0/0 closing read/write socket
2023-11-29T07:07:49.551757+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] map1_Eth0/0 > tunnel down
2023-11-29T07:07:49.551847+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] sdwan0 redis: changing map map1_Eth0/0 status to
DISCONNECTED
```

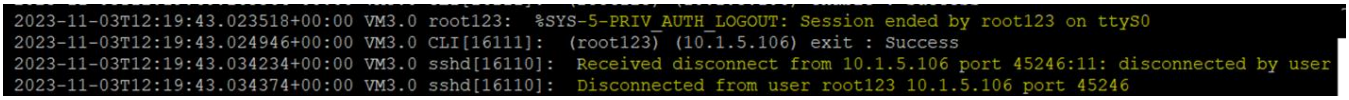
  - Failure of the trusted channel functions

```
2023-07-27T11:37:32.540352+00:00 VM3.0 %SDWAN0:  check_map_phys_settings()
2023-07-27T11:37:32.540414+00:00 VM3.0 %SDWAN0_MapPhys3:  DEBUG[sdwan0MapPhys3] - no ifaceName
2023-07-27T11:37:33.357943+00:00 VM3.0 %SDWAN0-PROC:  [INFO] map1_Eth0/0 bind to 10.1.5.142
2023-07-27T11:37:33.358069+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] map1_Eth0/0 peer chain cert-
2023-07-27T11:37:33.358161+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls] 0 s:/C=US/O=Acumen/OU=CC/CN=10.1.3.179
2023-07-27T11:37:33.358250+00:00 VM3.0 %SDWAN0-PROC:  [INFO/dtls]   i:/C=US/O=Acumen/OU=CC/CN=KlasCA
2023-07-27T11:37:33.358339+00:00 VM3.0 %SDWAN0-PROC:  [WARN/dtls] verification error - unsuitable certificate purpose - in above cert
ificate of chain
2023-07-27T11:37:33.358427+00:00 VM3.0 %SDWAN0-PROC:  [WARN/protocol] DTLS protocol error on map1_Eth0/0 - certificate verify failed
2023-07-27T11:37:33.358512+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] map1_Eth0/0 closing read/write socket
2023-07-27T11:37:33.358597+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] map1_Eth0/0 > tunnel down
2023-07-27T11:37:33.358684+00:00 VM3.0 %SDWAN0-PROC:  [INFO/protocol] sdwan0 redis: changing map map1_Eth0/0 status to DISCONNECTED
```

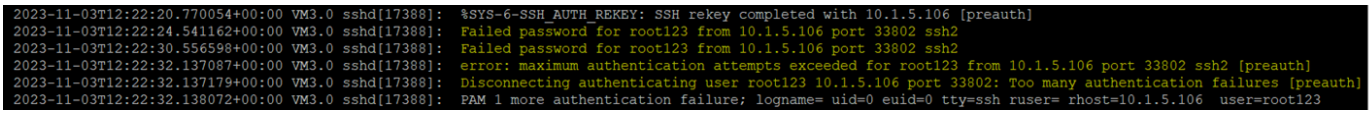
- FTP_TRP.1/Admin
    - Initiation of the trusted path



    - Termination of the trusted path.



    - Failure of the trusted path functions.



8.2.1. Non-administrative User Authentication

8.2.1.1. Local Console

This log event describes a successful attempt by a user to login to the ToE on the local console:

```
2024-04-30T20:31:44.176220+00:00 KlasOS login[8471]: %SYS-5-PRIV_AUTH_SUCCESS:
Authentication Accepted by klas on ttyS0
```

where 'ttyS0' is the local console port and 'klas' is the username.

Unsuccessful attempt by the user on the local console would look as follows:

```
2024-04-30T20:31:44.176220+00:00 KlasOS login[8517]: %SYS-5-PRIV_AUTH_FAIL:
Authentication Failed by klas on ttyS0
```

When the non-administrative user logs out of the ToE, the following message is displayed in the system log:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_LOGOUT: Session ended by
klas on ttyS0
```

When the non-administrative user session times out the following is logged:

```
2024-04-30T20:31:44.176220+00:00 KlasOS %SYS-5-PRIV_AUTH_TERMINATED: Session Ended for
klas on ttyS0
```

8.2.1.2. Remote SSH Session

Authentication to the ToE over a remote SSH session can be performed using either public-key authentication or password authentication. Here are some examples of log events for both methods and whether the authentication attempt succeeded or failed.

This log event is for a user successfully authenticating using a remote SSH session:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[8662]: %SYS-5-SSH_AUTH_STATUS: Password
Authentication Accepted by klas on 192.168.3.10
```

NOTE: By default the ToE tries to authenticate using a public key (See Section 7 'Remote Administration using SSH'. If no public key is found the following message will be displayed before the password authentication is accepted:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[8662]: %SYS-5-SSH_AUTH_STATUS: Public Key
Authentication Failed by klas on 192.168.3.10
```

A **successful** authentication attempt by a user using public-key authentication would look like the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[15527]: %SYS-5-SSH_AUTH_STATUS: Public
Key Authentication Accepted by klas on 192.168.3.10
```

This log event is for a user **unsuccessfully** authenticating using a remote public-key SSH session:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[15571]: %SYS-5-SSH_AUTH_STATUS: Public
Key Authentication Failed by klas on 192.168.3.10
```

This log event is for a user **unsuccessfully** authenticating using a remote password-based SSH session:

```
2024-04-30T20:31:44.176220+00:00 KlasOS sshd[9132]: %SYS-5-SSH_AUTH_STATUS: Password
Authentication Failed by klas on 192.168.3.10
```

On SSH logout the following message is displayed in the system log:

```
2024-04-30T20:31:44.176220+00:00 KlasOS: %SYS-5-SSH_AUTH_LOGOUT: Session ended by klas
on 192.168.1.10
```

When an SSH user session is timed out the following message would be displayed:

```
2024-04-30T20:31:44.176220+00:00 KlasOS [15370]: %SYS-5-SSH_AUTH_TERMINATED: Session
Ended for klas on 192.168.1.10
```

8.2.2. Administrator Authentication

After a user has logged in, they get access to a basic set of '**`show`**' commands, can do a '**`ping`**' or '**`traceroute`**' or check the firmware version.

To get administrator access, they must enter the **enable** command. The administrator must ensure this privilege escalation command is password protected using the guidelines set out in Section 3.1 for generating passwords.

Administrator authentication is also logged by the ToE. The following gives some examples of administrator authentication log events.

Successful login by administrator user at local console:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_SUCCESS: Authentication to
privilege level 15 succeeded by klas on ttyS0
```

Unsuccessful login by administrator user at local console:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_FAIL: Authentication to
privilege level 15 failed by klas on ttyS0
```

Successful login by administrator user over remote SSH session:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_SUCCESS: Authentication to
privilege level 15 succeeded by klas on 192.168.3.10
```

Unsuccessful login by administrator user over remote SSH session:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_FAIL: Authentication to
privilege level 15 failed by klas on 192.168.3.10
```

When the administrator is logged out of the ToE the following message is logged to the system log:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_LOGOUT: Authentication
level 15 ended by klas on ttyS0
```

Or if the session was over SSH:

```
2024-04-30T20:31:44.176220+00:00 KlasOS : %SYS-5-PRIV_AUTH_LOGOUT: Authentication
level 15 ended by klas on 192.168.3.10
```

8.4. Sending Logs to Syslog Server

The ToE allows the administrator to specify a syslog server to which all relevant logs can be sent.

On configuration of a remote syslog server, all contents of the System log and Audit log will be sent to the syslog server in real-time. If at any time the connection to the syslog server is broken, the administrator shall verify and restore physical/network connectivity between the TOE and the remote entity and reinitiate the connection.

To configure the logs to be sent to a syslog server, use the following command in global configuration mode:

- **`logging host 127.0.0.1`**
    - The log event for this would look as follows:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[3330]: (admin) (admin) (ttyS0) logging
host 127.0.0.1 : Success
```

To stop the logs being sent to a syslog server do:

- **`no logging host 127.0.0.1`**

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[3330]: (admin) (admin) (ttyS0) no logging
host 127.0.0.1 : Success
```

IMPORTANT NOTE: All contents of the Audit log and System log are simultaneously sent to both the local logs on the ToE and the audit/syslog server.

9. SSH Tunnel for Trusted Channel

The ToE uses an SSH tunnel for the Trusted Channel for syslog messages that are sent from the ToE to a remote syslog server.

NOTE: SSH client functionality invoked from the CLI used for anything other than establishing the Trusted Channel is not evaluated.

SSH client on KlasOS supports SSH version 2 only. SSH version 1 is not supported.

SSH client on the ToE is restricted to the following algorithms:

- Encryption using AES-CTR-128, AES-CTR-256, AES-CBC-256 or AES-CBC-128
- Public key user and host authentication using SSH-RSA, ECDSA-SHA2-NISTP256 or ECDSA-SHA2-NISTP384
- Integrity using HMAC-SHA1, HMAC-SHA2-256, or HMAC-SHA2-512
- Key exchange using DIFFIE-HELLMAN-GROUP14-SHA1, ECDH over NIST P256 with SHA2 or ECDH over NIST P384 with SHA2.

NOTE: These algorithms are not configurable by an administrator. The algorithm used will depend on the algorithms the SSH server is using, and the type of key generated on the ToE and is restricted to the algorithms outlined above. The use of any other cryptographic engines other than those listed above were not evaluated or tested during the CC evaluation of the ToE. The TOE does not require configuration for key size(s) and mode(s) for data encryption/decryption, since it is pre-configured and fixed.

9.1 Add public key to SSH/syslog Server

Before configuring the tunnel on the ToE, copy the generated public key on the ToE to the syslog server authorized key file, normally located in /home/<user>/.ssh/authorized_keys. If the file does not exist, it can be created. Instructions on generating a keypair on the ToE is explained in Section 6.1. This is the same keypair used by the ToE SSH Server for the Host Key. To get an SECSH formatted public key from the ToE, run the following command in privileged EXEC mode:

- `show ip ssh`

The output will look like:

```
SSH Enabled - version 2
Keys in SECSH format:
ecdsa-sha2-nistp384 AAAAE2VjZHNhLXNoYTItbmlzdHAzODQAAAAIbmlzdHAzODQAAABhBDmTldLQ
TazBRPog7uKtSoAryCtyb73YZCOPZcVAZ4+JAbvfIsicvzWZ+9APv3XvWwpEClau2U/CAXqfYpuD+EqT
Y/chx3Hvo22UeKDwyHa4Va4fgCMuEEhoKz/ai9wiRA== ec-key-384
```

The SECSH formatted key is on the third line. It includes the ecdsa-sha2-nistp256 part, but not the “ec-key-256” label at the end of the line:

```
ecdsa-sha2-nistp384 AAAAE2VjZHNhLXNoYTItbmlzdHAzODQAAAAIbmlzdHAzODQAAABhBDmTldLQ
TazBRPog7uKtSoAryCtyb73YZCOPZcVAZ4+JAbvfIsicvzWZ+9APv3XvWwpEClau2U/CAXqfYpuD+EqT
Y/chx3Hvo22UeKDwyHa4Va4fgCMuEEhoKz/ai9wiRA==
```

9.2. Add SSH Server Host Key to TOE known_hosts

Ensure that strict host key checking is enabled. Run the following command in global configuration mode.

- **ip ssh strict hostkey checking**

Add the SSH server's hostkey to the known_hosts file on the ToE. The SSH server's hostkey can normally be found under the /etc/ssh/ssh_host_*.pub file for a linux server. Run the following commands in global configuration mode to add the hostkey on the ToE.

- **ip ssh known-hosts**
- **key-string <server ip address> <public key algorithm> <key data>**

  Replace <server ip address> with IP address of syslog server

  Replace <public key algorithm> with the server's public key algorithm

  Replace <key data> with the public key string.

An example command:

- **key-string 192.168.1.9 ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCg3EmANUfJ+yWMq3ZLdCetmujs24f6xpeXI7VQeocmvDaZwy+qPJoxQ8szSHKlKBCoMd9eUgw0NM4aH6927Ba8P2tnxXZhXoSHp9ZKE4zmK/+nHVU9QfdefYBH/YOGx4qum0KbXNHICDEtIw9df41Sanfqn2KXG/sHWLRb8N08CiO9+5/JQWYVMkNqDo46e2T8Ie08Rr9+VljvpkhGc7p0P3TYfs2I21rLYpRKXCje1uIr612zEh0ednuzgsRilXsn09ckaSSxECoeRptoim5Xsui6JqRpX5hc8j3u+U9ROY2XJhzCJ6zD97qPBYRhKYiN1LrfMNA4XEnvRwh96WQr**

Adding entries to the known_hosts file is logged to the syslog on the ToE and would look similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS SSH-6-INFO: Key for host 192.168.1.9 added to
known hosts
```

An entry in the audit log is also made which would look similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[3150]: (admin) (admin) (ttyS0) key-string
192.168.1.9 ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDTBQ5e/Zl3vNgreuO2sL4K/tNfn1CgJQovlNK44LTl7e6/lWjoNQVGfQ
zvwQ0V/AUFxJ6XGkuDMfcTOsuQVW3ZjLcr/1FkGg4UaxzFWHPE5Ck+nHV5vNeNQNBms22Uj1a9mM5BQJarXrVx
snIWcZgTlr11Lg6N2V18x9jZSqd3LfDbxirKHEKuLynXbBbcz02ZfNYw3kspTdkgEBckn/QPCc6+05KJY76EFP
c4rK3Yu+fgRLw5TIjQ/4jUFk/tmsvmaIFJavImR/H57Md8K66T3/eBC50tE5QcCQ8cqjqqHTc8yyLCMu9YzfLu
8dgfve9RL4cmOX6U4j5+0DRcuuXx : Success
```

To view the entries in the known_hosts file on the ToE, run the following command in privileged EXEC mode:

- **show ip ssh known-hosts**

To delete a specific entry in the known_hosts file, run the following commands in global configuration mode:

- **ip ssh known-hosts**
- **no key-string <server ip address> <public key algorithm>**

  Replace <server ip address> with IP address of syslog server

  Replace <public key algorithm> with the server's public key algorithm

Removing a specific entry from the known_hosts file is logged in the audit log with a message similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[4603]: (admin) (admin) (ttyS0) no key-string 192.168.1.9 ssh-rsa : Success
```

To clear all entries in the known_hosts file, run the following command in privileged EXEC mode:

- `clear ip ssh known-hosts`

Removing all entries from the known_hosts is logged to the syslog with a message similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS SSH-6-INFO: Removing ssh-rsa key for known host 192.168.1.9
```

9.3. Configure SSH Tunnel

To configure the SSH tunnel on the ToE, run the following command in global configuration mode.

- **ssh tunnel username <username> host <syslog server IP> localport 50514 remoteport 514**

  Replace <username> with the correct username on the syslog server we will be building the SSH tunnel to.

  The <syslog server IP> is the IP address of the syslog server.

  Localport can be any unused port on the ToE

  Remote port is the port the syslog server will be listening to for incoming syslog messages

Initiation of the SSH tunnel is logged to the audit log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for initiating the SSH tunnel would look similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2827]: (admin) (klasconsole) (ttyS0) ssh
tunnel username buildmaster host 192.168.1.9 localport 50514 remoteport 514 : Success
```

If the tunnel builds successfully, the following message will be displayed:

```
2024-04-30T20:31:44.176220+00:00 KlasOS /usr/bin/ssh[4875]: %SYS-6-SSH: Authenticated to
192.168.1.9 ([192.168.1.9]:22).
```

If the tunnel **FAILS** to build successfully, a message is logged to the syslog similar to the following::

```
2024-04-30T20:31:44.176220+00:00 KlasOS /usr/bin/ssh[5388]: %SYS-6-SSH: publickey
authentication failed
```

SSH tunnel failures are displayed in the log in the following format:

- `[Date and Time] [Hostname of ToE] /usr/bin/ssh: [tag]: [reason for failed connection attempt]`

To terminate the SSH tunnel on the ToE, run the following command in global configuration mode.

- **`no ssh tunnel username <username> host <syslog server IP> localport 50514 remoteport 514`**

  Replace <username> with the correct username on the syslog server we will be building the SSH tunnel to.

  The <syslog server IP> is the IP address of the syslog server.

  Localport can be any unused port on the ToE

  Remote port is the port the syslog server will be listening to for incoming syslog messages

Termination of the SSH tunnel is logged to the audit log. See Section 8 'Logging and Auditing' for information on the audit log and the format of the log messages. The log message for terminating the SSH tunnel would look similar to the following:

```
2024-04-30T20:31:44.176220+00:00 KlasOS CLI[2827]: (admin) (klasconsole) (ttyS0) no
ssh tunnel username buildmaster host 192.168.1.9 localport 50514 remoteport 514 :
Success
```

The ToE ensures that a SSH rekey happens after no more than 1 GB of data has been transmitted and received or after 1 hour, whichever is arrived at first. When a SSH rekey occurs the following message is displayed in the system log:

SSH Client Message:

```
2024-04-30T20:31:44.176220+00:00 KlasOS /usr/bin/ssh[2909]: %SYS-6-SSH_AUTH_REKEY: SSH
rekey completed with x.x.x.x
```

The SSH tunnel will attempt to reconnect automatically when it detects the connection to the remote SSH server is broken. An administrator can also manually restart the tunnel by performing the following commands in global configuration mode:

- **`no ssh tunnel username <username> host <syslog server IP> localport 50514 remoteport 514`**
- **`ssh tunnel username <username> host <syslog server IP> localport 50514 remoteport 514`**

  Replace <username> with the correct username on the syslog server we will be building the SSH tunnel to.

  The <syslog server IP> is the IP address of the syslog server.

  Localport can be any unused port on the ToE

  Remote port is the port the syslog server will be listening to for incoming syslog messages

10. Introduction to Certificate Manager

The Certificate Manager (certmgr) is a feature that allows a KlasOS device to create certificate signing requests, store and manage certificate files through 'certmgr trustpoint' objects.

The feature can be used with SDWAN in 'pki-DTLS' encryption-mode and in 'ip http secure-server'.

Cert manager is used in conjunction with an External Certificate Authority (CA). This allows administrators to install Certificates on KlasOS devices using their existing PKI infrastructure.

To create a fully configured 'certmgr trustpoint', the KlasOS device goes through the following stages.

Configuration of Keys
The device must have an RSA keypair configured. (There are plans to also support other PKI algorithms (ECDSA) in the future for cert generation)

Device 'subj' information
The subj information is configured under the certmgr trustpoint

Certificate signing request generation
The certificate manager generates a Certificate signing Request which is then provided to a trusted Certificate Authority (CA). For testing or getting started, xCA at https://hohnstaedt.de/xca is a good application to use.

Certificate install
The trusted CA signs the certificate and returns the signed certificate and its CA (root) certificate to be installed on the device and configured with the 'certmgr trustpoint' object

NOTE: The TOE can be configured to use OCSP (Online Certificate Status Protocol) as part of the SDWAN DTLS connection setup for each connection attempt. If a certificate's validity cannot be determined, the certificate is rejected by the TOE. If a certificate is being used for DTLS, it needs to have either the "server authentication" or "client authentication" extendedKeyUsage set depending on if the device is a client or server.

10.1. Generating and Adding Certificates to a Certificate Manager

Certificate manager certificates may only be used in conjunction with SDWAN ('certmgr' setting). The certificate manager feature allows multiple SDWAN interfaces to use the same certificates configured by a single trustpoint, thus simplifying configuration.

The steps to create and apply certificate to a trustpoint are as follows:

1. Create a key pair on the device and give it a label.

Check if there are already keys present on the system by running the command:

```
show ip ssh
```

If there are keys on the system, the public key will be displayed in the format:

```
<key algorithm> <public key in sesch format> <key id>
```

Example:

```
KlasOS# show ip ssh
SSH Enabled - version 2
Keys in OpenSSH format:
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDfJBEQ/NDx0u0idDKwlLrBhSZmmaZDem/GrLgxI+ATgeKdF7qvs
dTYZxHDYBdo2vmQdiCvzBQT92XYrPH5mADxJs6ix44o5zEkcNopWm0x3INDheOXMesSbBKGbAwpqcUjpa
zCpCR4cL/v2ryVwg7d8xxoPs9WEj/STkXIjWM6nRs7e1Eu4HHT91qI5RRw/OlxzjKvL6oriSHdC0OUU/5
b2eygjK3UoG3ZM8kpbhVnUQPZovfktJCooojsDhRs9ghwN/+xGLBG3jYk1XQJhGO6Pb2KI5QiR4qmqyZP
jVLipWVl1oAgBQSI+PgJr4bPDXy65j7ClpF4IJIR7HzVBquf clientkey
```

If no keys are present on the system, a key pair must be generated by using the following command (only RSA is supported for now):

```
crypto key generate [ec|rsa] general-keys modulus [2048|3072|4096] label
<mylabel>
```

2. Create a certmgr trustpoint on the device

```
KlasOS# configure terminal
KlasOS(config)# certmgr trustpoint <trustpoint name> RSA
KlasOS(cert-tp-mytrustpoint)#
```

3. Reference the key pair label using the 'key-id' command.

```
KlasOS(cert-tp-mytrustpoint)# key-id <mylabel>
```

4. Give the trustpoint a valid 'subj' field using the 'subj' command.

Example:

```
KlasOS(cert-tp-mytrustpoint)# subj "/CN=yourdomain.com"
```

NOTE: Use quotes if the subject line contains any spaces.

NOTE: The subject line has the following restrictions:

```
The subject string must begin with a '/'
Fields or Key/value pairs are separated with an '='

Valid keys are:
'commonName'

The following fields can be no longer than 64 characters:
'commonName'
```

5. Generate a Certificate Signing Request

Generate a Certificate Signing Request using the 'certmgr generate CSR trustpoint' command.

```
KlasOS# certmgr generate CSR trustpoint <trustpoint name> flash: <CSR filename>
```

This will generate the CSR and place it in flash.

6. Provide the newly created CSR to the Certificate Authority

The newly created CSR can now be sent to the Certificate Authority who will then provide your device certificate and the CA certificate.

The contents of the CSR can be displayed by running:

```
KlasOS# more flash: <CSR filename>
```

Or the CSR file can be copied to another device for processing by the trusted CA using the 'copy flash: scp' command.

```
KlasOS# copy flash: scp:
Address of remote host [192.168.3.10]?
Destination username []? user1
Source filename []? device.csr
Destination filename []? device.csr
```

7. Copy the device and CA certificate to flash.

Once the device certificate and the CA's certificate have been provided by the Certificate Authority, copy them to flash using the 'copy scp flash:' command.

```
KlasOS# copy scp: flash:
Address of remote host []? 192.168.3.10
Source username []? user1
Source filename []? device_cert.pem

KlasOS# copy scp: flash:
Address of remote host []? 192.168.3.10
Source username []? user1
Source filename []? CA_cert.pem
```

8. Reference the device and CA certificate in the trustpoint

```
KlasOS# configure terminal
KlasOS(config)# certmgr trustpoint mytruspoint RSA
KlasOS(cert-tp-mytruspoint)# device-cert flash: device_cert.pem
KlasOS(cert-tp-mytruspoint)# ca-cert-chain flash: CA_cert_chain.pem
```

NOTE: The `CA_cert_chain.pem` file may contain multiple intermediate CA certificates, however the `device_cert.pem` may only contain the certificate for the device.

The trustpoint is now ready to be used with SDWAN.

It can be used with pki-DTLS transport/encryption modes.
Use the 'encryption-mode' command as below when configuring SDWAN

```
KlasOS(config)# interface sdwan 5
KlasOS(config-if-sdwan5)# encryption-mode pki-DTLS AES-GCM
KlasOS(config-if-sdwan5)# certmgr mytruspoint
```

9. Configuring a reference identifier for a trustpoint

```
KlasOS(cert-tp-mytruspoint)# reference-id FQDN (FQDN or IPv4/IPv6)
KlasOS(cert-tp-mytruspoint)#  validation identifier-check (strict/basic)
```

Note: “validation identifier-check” should always be set to “strict” in the Common Criteria configuration. The TOE supports both IPv4 and IPv6 as well as FQDN in the CN/SAN of a certificate. SAN always takes priority if it is present in the certificate.

10.2. Certificate Manager/Trustpoint Troubleshooting

There are several 'show' commands that may help troubleshoot issues with certificates.

Examples are shown below.

- Show all configured trustpoints on the system

```
KlasOS# show certmgr trustpoints

Certificate Manager          Status
-------------------          ------
servertp                     Validated
```

- Show certificate details of a named trustpoint

```
KlasOS# show certmgr trustpoints servertp

Certificate Manager: servertp

CA Certificate
--------------
Name:          CA_cert.pem
Status:        Valid
Expires:       Sep 19 15:30:20 2120 GMT


Device Certificate
--------------
Name:          server.pem
Status:        Valid
Expires:       Oct 11 15:30:56 2030 GMT
```

- Show details of a CSR (Certificate Signing Request)

```
KlasOS# show certmgr CSR flash: server.csr
Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=US, ST=Utah, L=Lehi, O=Your Company, Inc., OU=IT,
CN=yourdomain.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:d9:2e:cf:75:0f:1a:8b:3f:f5:d8:67:79:39:8b:
                    1c:59:c2:c4:00:97:f9:14:f8:f8:d8:1a:80:23:53:
                    de:9b:72:aa:a5:ff:94:fe:46:e3:82:db:d9:e5:a7:
                    8c:30:3a:cc:63:3e:05:06:24:70:ce:7f:35:e5:8c:
                    2d:0c:e3:fc:71:14:8b:f3:3d:2d:ae:ba:4b:9a:71:
                    a4:c4:61:7e:3e:41:c2:dd:56:fa:e8:21:ea:e4:12:
                    7a:90:be:ed:69:c1:4e:80:fa:da:6b:1c:ee:4b:99:
                    3b:f0:93:b2:53:ed:95:ca:b7:0b:52:7d:3f:97:be:
                    50:e9:2e:7b:37:13:a8:45:d5:a8:1e:38:2f:8c:a4:
                    60:a6:c0:a2:55:82:7a:a4:b2:fe:f1:de:83:6a:01:
                    19:b5:dc:10:24:74:f1:fa:10:96:27:4a:5c:2c:7c:
                    84:1d:e3:47:b5:61:bb:c6:70:56:29:40:73:b8:24:
                    71:de:c3:7a:41:2e:a5:78:7b:b2:a0:24:09:61:f0:
                    f2:61:a0:c4:74:0f:13:6b:83:35:db:42:d2:57:d3:
                    c4:78:fd:02:be:e9:5d:e4:c8:42:26:21:e3:2a:5c:
                    ab:98:ad:d3:5e:2d:37:8a:04:d8:42:ab:68:b5:e8:
                    d0:37:60:db:2c:c4:b9:e0:4d:1g:7d:ea:37:79:2c:
                    6e:0d
                Exponent: 65537 (0x10001)
        Attributes:
            a0:00
    Signature Algorithm: sha384WithRSAEncryption
         bc:1d:af:48:d4:86:30:62:4a:5b:05:8f:e8:36:a8:b1:7c:51:
         8f:88:3b:00:a4:83:d0:bf:5e:73:c9:cf:99:dc:ea:f5:e0:70:
         3f:88:01:ff:66:e1:ca:bd:2d:a7:fb:8a:8c:e2:88:53:f1:c2:
         55:63:eb:98:74:1e:24:1b:1a:b7:10:50:09:3f:30:96:33:e3:
         9a:65:a3:f2:b4:72:5d:8d:76:8d:91:15:f1:ad:61:43:15:ff:
         3c:27:f6:7e:f7:02:ab:4b:7c:70:86:fc:87:91:e4:4c:d8:b4:
         e1:e0:5c:54:11:7c:07:1a:79:52:42:a0:4a:a1:35:34:6c:69:
         35:46:92:84:9d:8e:f7:eb:e6:cd:05:7d:16:e8:2e:65:73:df:
         ff:a4:c3:ad:eb:90:c7:05:d7:87:a9:39:a3:78:ff:92:c1:b4:
         ea:ce:f8:f8:d5:0c:cb:76:58:8f:af:26:52:7d:79:12:eb:69:
         ef:30:ff:e4:b1:30:d7:f5:55:ba:55:fa:08:09:bd:76:55:08:
         ce:7b:29:dd:e5:53:31:1d:f2:aa:3a:f2:18:7b:c8:ff:56:88:
         9b:e6:6f:2e:68:c2:c4:ab:e1:58:ef:c7:8b:58:02:a7:40:40:
         c9:9d:f2:85:aa:db:4e:4f:02:d3:cg:aa:42:6a:47:34:c0:c7:
         57:2b:ef:b0
```

- Show details of a certificate

```
KlasOS# show certmgr cert flash: server.pem
```

```
Certificate: server.pem
---------------------------

Version:                                                                    3
(0x2)
Serial Number:
69:3a:65:d8:2b:a2:24:56:e2:4a:fb:74:f7:83:16:7c:55:2e:c6:4b
Issuer:
CN=ca.cert.local
Subject:                      C=US, L=Lehi, O=ServerCompany, OU=IT,
CN=ServerCompany.com
Validity:
   Not Before: Oct 13 15:30:56 2020 GMT
   Not After : Oct 11 15:30:56 2030 GMT
CA:FALSE
X509v3 Key Usage:
   Digital Signature
   Non Repudiation
   Key Encipherment
   Data Encipherment
   Key Agreement
   Certificate Sign
```

- Show full details of a certificate

```
KlasOS# show certmgr cert flash: server.pem full

Certificate: server.pem
---------------------------

Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            69:3a:65:d8:2b:a2:24:57:e2:4a:fc:74:f7:83:16:7c:55:2e:c6:4b
    Signature Algorithm: sha384WithRSAEncryption
        Issuer: CN=ca.cert.local
        Validity
            Not Before: Oct 13 15:30:56 2020 GMT
            Not After : Oct 11 15:30:56 2030 GMT
        Subject: C=US, L=Lehi, O=ServerCompany, OU=IT, CN=ServerCompany.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:b2:c0:54:4c:06:fc:60:3b:6c:50:b6:6d:f7:5d:
                    b3:4e:56:01:1b:c8:d5:d0:44:47:cd:ea:12:29:aa:
                    2c:e8:82:63:55:bc:24:7d:ca:4d:0b:36:9b:8d:1e:
                    43:17:54:71:fb:a4:61:ef:cf:f9:00:89:6b:e7:8e:
                    c4:1c:d7:c2:2a:f4:8b:10:20:ac:db:02:bd:dc:f3:
                    ff:3b:9e:ab:81:33:a9:9d:b2:5f:2b:f6:f8:94:11:
                    14:20:b9:80:1b:a6:9b:6a:00:a7:e4:06:ea:ca:f5:
```

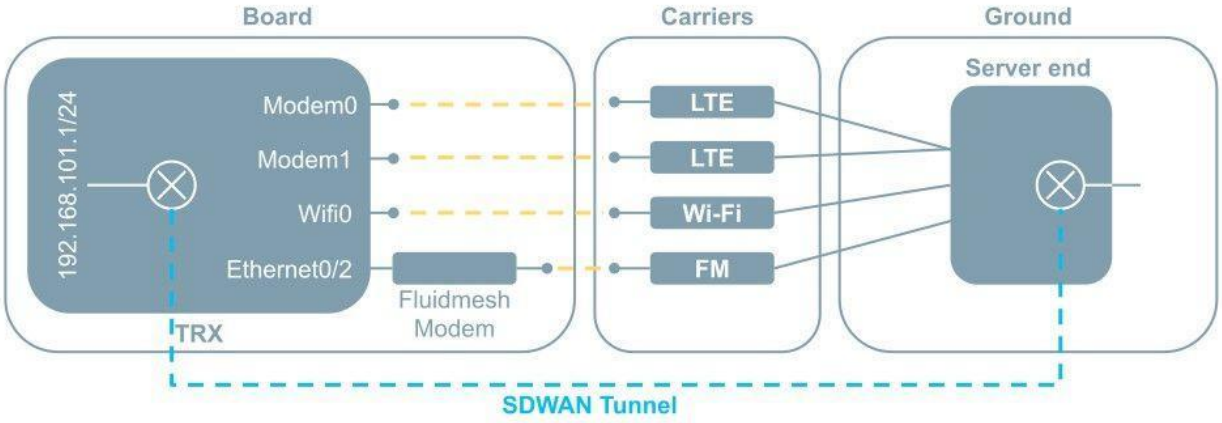
```
                    e0:db:51:5d:c0:3e:fe:47:1f:bb:43:50:2d:76:7e:
                    dc:57:f7:e0:ac:ea:73:50:7d:73:a6:f1:40:d8:1d:
                    3d:9c:99:00:7c:74:55:d4:a8:1c:b7:a1:a1:8b:3d:
                    6f:63:fc:a6:1a:e2:da:c6:b0:16:8b:01:0d:d4:3b:
                    07:05:6e:ce:55:0e:70:43:8c:b7:08:5b:d7:a1:6e:
                    04:73:ca:5b:0c:da:b1:06:7f:0c:0e:ef:00:58:5c:
                    86:48:db:99:dc:e5:78:d3:af:0b:07:86:4c:28:0e:
                    fe:aa:e2:68:a7:03:9c:b0:c0:33:1d:e2:4a:c8:7c:
                    2b:29:c7:08:a6:1d:55:a8:ce:47:1d:22:75:36:a0:
                    52:a7:e9:b8:57:f1:06:5e:cc:55:0c:c7:4a:58:3d:
                    a5:bf
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            X509v3 Key Usage:
                Digital Signature, Non Repudiation, Key Encipherment, Data
Encipherment, Key Agreement, Certificate Sign
            X509v3 Subject Key Identifier:
                B8:A7:CE:A9:8F:C1:5F:20:0D:68:DD:20:3E:53:72:2C:9B:AE:06:32
            X509v3 Authority Key Identifier:
                keyid:C8:0A:2A:14:29:38:04:74:2B:EF:5B:A0:28:74:3E:CD:B6:79:70:E0

            X509v3 Extended Key Usage: critical
                Time Stamping
    Signature Algorithm: sha384WithRSAEncryption
         4f:bd:28:89:40:bb:71:80:3f:c3:9e:29:8c:b8:66:8d:8a:c4:
         30:38:6e:42:0d:cf:fe:66:a9:04:76:d9:fe:e3:53:0d:b4:a1:
         9f:e2:8a:05:0b:01:c5:91:c6:14:22:e2:be:0a:89:d3:af:8f:
         c0:d4:38:97:3d:8e:60:a4:a5:66:1f:72:3f:3d:97:64:30:17:
         ba:1b:e3:4d:6d:b9:23:09:73:a7:5d:0b:64:91:a7:84:0e:3a:
         cd:41:43:26:0c:0c:bf:94:9f:23:eb:58:f4:bb:f7:8e:b2:96:
         89:ae:46:53:41:34:91:54:63:32:da:02:00:52:df:2d:2d:19:
         99:29:39:c2:87:07:c6:23:3b:77:d1:11:b1:59:74:cb:88:20:
         3a:b5:cb:a7:ce:be:73:13:19:aa:25:04:bd:41:ae:ec:15:30:
         0c:99:7a:19:03:34:71:fd:95:03:c8:70:0c:5b:df:e2:9f:c1:
         a3:78:12:bb:83:26:28:5f:d8:3e:e0:55:8d:81:56:a7:70:c5:
         13:a3:d0:5b:56:b5:74:4e:64:0a:42:94:4d:32:05:3f:81:27:
         c6:fb:14:f2:5f:28:68:5f:25:a0:32:52:ce:b9:58:ee:fc:e7:
         07:e3:56:95:26:21:04:60:0c:92:3f:d9:17:b8:fc:a8:9e:1d:
         5e:8a:50:5f:bb:fe:81:5e:9a:ff:99:ac:45:b5:ef:44:a7:76:
         2a:c1:cd:f0:0a:dd:5c:b9:9b:04:75:52:c6:19:b9:a5:d5:6f:
         64:7d:ae:90:ee:29:f0:48:de:eb:97:83:4c:f4:6a:b0:ee:85:
         63:6e:cb:d2:19:94:11:71:c4:5e:a8:c4:15:06:13:62:dd:08:
         34:ae:3d:88:04:c3:80:81:f7:0c:c7:1f:32:d3:2e:f2:24:8a:
         c3:77:12:9e:37:12:93:56:97:42:0e:86:5b:8d:04:5e:84:d0:
         48:38:1c:59:2f:cb:ba:c9:f6:9d:6a:ed:c0:b9:d5:5f:b9:19:
         a2:7d:81:bb:fa:e6
```

10.3. SDWAN Configuration Example

In the following example there are four WAN options on a TRX device:

1. LTE connection via Modem 0
2. LTE connection via Modem 1
3. Track-side Wi-Fi using Fluidmesh Mobi Radio
4. Station Wi-Fi via Cisco Wi-Fi network

A ground routable IP subnet must be allocated to each individual SDWAN client. In this example, 192.168.101.1/24 will be allocated to the TRX-R.



10.4. Network Interfaces Configuration

10.4.1. Client-side Configuration

Configure a local IP address for KlasOS on the routable SDWAN client subnet. This interface is the address that the SDWAN server associates with a given SDWAN client, and a dedicated vSwitch is recommended for the purpose.

```
interface vSwitch 101
 description "SDWAN Client network"
 ip address 192.168.101.1 255.255.255.0
!
```

Now configure all 4 WAN links, from a networking point of view.

10.4.2. Server-side Configuration

In this example, we are running the SDWAN server-end as a KVM Guest. There are 3 interfaces required for inter-connects to the WAN options being used.

10.4.2.1. SDWAN Configuration

A 'physical mapping' is configured for each link, using a pre-defined UDP port on the server-end. A matching configuration is required on the server-end as below.

The following parameters are matched on either end:

- Remoteport and bindport on client and server respectively. The tunnel is established using UDP traffic between the endpoints on these ports. Appropriate firewall and port-forwarding configurations will be required on the server-end to allow this traffic.
- Transit links over LTE networks for example can have MTUs below 1500 bytes. The MTU parameter must therefore be configured to ensure packet drops don't occur.

Client configuration	Server configuration

interface sdwan 0
 map physical 0
  bindport 5000
  phys modem 0
  remotehost 192.168.20.6
  remoteport 5000
 map physical 1
  bindport 5001
  phys modem 1
  remotehost 192.168.20.6
  remoteport 5001
 map physical 2
  bindport 5002
  phys vswitch 32
  remotehost 10.0.0.1
  remoteport 5002
 map physical 3
  bindport 5003
  gateway 10.1.0.1
  phys ethernet 0/0
  remotehost 10.1.0.1
  remoteport 5003
 filter 0 mapid 2 ip
 filter 1 mapid 3 ip
 ip tcp adjust-mss 1250
 hash-filter
 mtu size 1300
 password 0 Klas
 timeout 5
 version 3
 wrr mode srtt
 mode client
 local-service-ip 192.168.101.1
!
	interface sdwan 0
	 map client 0
	  client network 192.168.101.0 255.255.255.0
	 map physical 0
	  bindport 5000
	  phys ethernet 0/0
	 map physical 1
	  bindport 5001
	  phys ethernet 0/0
	 map physical 2
	  bindport 5002
	  phys ethernet 0/1
	 map physical 3
	  bindport 5003
	  phys ethernet 0/2
	filter 0 mapid 2 ip
	 filter 1 mapid 3 ip
	 ip tcp adjust-mss 1250
	 hash-filter
	 mtu size 1300
	 password 0 Klas
	 timeout 5
	 version 3
	 wrr mode srtt
	 mode server
	!

The above configuration example will enable a default route on the Client for all traffic to flow through the tunnel. On the server side, a route will be created in order to send traffic to 192.168.101.0/24 through the tunnel. The Hash-filter feature will consistently assign TCP or UDP flows to a single map physical.

10.5. Troubleshooting

After verifying that all transit connections are passing traffic, the following commands can be used to troubleshoot SDWAN on either the Client or Server. If at any time the SDWAN connection is broken, the administrator shall verify and restore physical/network connectivity between the TOE and the remote entity and reinitiate the connection.

10.5.1. Verify which tunnels are up

```
KlasOS# show sdwan
Ent  Flag        Physical Map
---  ----------  ------------------------
0    up          phys 0: modem0 (up)
                 phys 1: modem1 (up)
                 phys 2: vSwitch32 (down)
                 phys 3: Eth0/0 (down)
```

10.5.2. Get more in-depth information on the SDWAN status use

```
KlasOS# show ip sdwan 0
Interface sdwan 0
  Proc     up
  Flag      up

  Mode                :               client
  Timeout in seconds  :                    5
  Reorder buffer size :                    0
  Loss tolerance (%)  :                  100
  MTU size in bytes   :                 1300
  Encryption          :                  OFF
  hash-filter         :              enabled
  Protocol Version    :                    3
  WRR Mode            :                 srtt
  Local Service IP    :        192.168.101.1

  No Client Maps


  Physical Maps:
    ID  Interface     RemotePort  BindPort    Upload B/W  Fallback  Status
    --  ------------  ----------  ----------  ----------  --------  ------
     0  modem0        5080        *                       Unset     up
     1  modem1        5081        *                       Unset     up
     2  vSwitch32     5082        *                       Unset     down
     3  Eth0/0        5083        *                       Unset     down

  Filters:
    Priority  Map (Interface)   Filter Syntax
```

```
--------  ----------------  -------------
       1  3 (Eth0/0)        "ip"
       0  2 (vSwitch32)     "ip"
```

SDWAN uses policy based routing based on the source address to ensure that outgoing traffic is sent over the desired link. Check that the source routing tables match the corresponding interfaces as below:

```
KlasOS# show ip route sdwan 0
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, P - PIM,
       > - selected route, * - FIB route

map physical 0
S>*    default [1/0] via 192.168.150.25, modem0
C>*    192.168.150.24/30 is directly connected, modem0


map physical 1
S>*    default [1/0] via 192.168.150.13, modem1
C>*    192.168.150.12/30 is directly connected, modem1


map physical 2


map physical 3
S>*    default [1/0] via 10.1.0.1, Eth0/0
C>*    10.1.0.0/24 is directly connected, Eth0/0
```

On the client, all traffic is routed to the head-end so a default route is added accordingly:

```
KlasOS# show ip route
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, P - PIM,
       > - selected route, * - FIB route

S*>*    0.0.0.0/0 [1/0] via 0.0.0.0, SDWAN
C>*    10.1.0.0/24 is directly connected, Eth0/0
S>*    172.16.1.0/24 [1/0] via 192.168.104.1, wifi0
S>*    172.16.2.0/24 [1/0] via 192.168.104.1, wifi0
S>*    172.16.3.0/24 [1/0] via 192.168.104.1, wifi0
C>*    192.168.101.0/24 is directly connected, vSwitch101
C>*    192.168.104.0/24 is directly connected, wifi0
C>*    192.168.111.0/24 is directly connected, vSwitch111
C>*    192.168.150.12/30 is directly connected, modem1
C>*    192.168.150.24/30 is directly connected, modem0
```

However, on the server, only a route via sdwan for specified networks is needed:

```
sdwan_12172# show ip route
Codes: C - connected, S - static, R - RIP, O - OSPF, B - BGP, P - PIM,
       > - selected route, * - FIB route
```

```
S>*    default [1/0] via 192.168.20.1, Eth0/0
C>*    10.0.0.0/24 is directly connected, Eth0/1
C>*    10.1.0.0/24 is directly connected, Eth0/2
C>*    192.168.20.0/24 is directly connected, Eth0/0
C>*    192.168.101.0/24 is directly connected, sdwan0
S>*    192.168.150.0/24 [1/0] via 192.168.20.5, Eth0/0
```

11. SDWAN Encryption and encryption-mode

For KlasOS Keel release 5.4.0, the 'encryption-mode' setting was added. This enabled the use of OpenSSL PKI and DTLS libraries to negotiate a shared key for encrypt/decrypt purposes.

NOTE: The default configuration **MUST** include 'pre-shared-key' setting.

11.1. encryption-mode setting

Encryption mode has the following options:

```
KlasOS(config-if-sdwan0)# encryption-mode
  pki-DTLS        Set encryption in DTLS mode
  psk             Set encryption in legacy mode
```

PKI-DTLS uses a DTLS session per SDWAN map physical to negotiate keys and encrypt traffic. Map phys MAY terminate in different SDWAN headend instances since the DTLS sessions are independent. This is important for some customer failover scenarios.

11.1.1. encryption-mode pki-DTLS

PKI-DTLS setting uses a single DTLS session per 'map physical' to negotiate session keys. Once the session keys are negotiated, they may be renegotiated every <pki-reath-time> periods.

pki-DTLS example:

```
Current configuration:
!
certmgr trustpoint TRAIN10 RSA
 subj /CN=TRAIN10
 ...
!
interface sdwan 0
...
 version 3
 wrr mode srtt
 encryption-mode pki-DTLS AES-GCM
 pki-reauth-time minutes 10
 pre-shared-key 0 klas123
 mode client
 certmgr TRAIN10
 local-service-ip 192.168.99.1
!
```

'certmgr' setting is described in a separate article, but it is used to set the sdwan interface x.509 certificate to use for the DTLS key negotiation process. The only ciphers supported for this DTLS mode are the following:

- TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 3268,
- TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 3268,

- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246,
- TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288,
- TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288

NOTE: When the TOE is configured to be in a CC compliant configuration, DTLSv1.2 is the only accepted version of DTLS. Session ID's and session tickets are not supported in this configuration. No other configuration steps are necessary to operate in a CC compliant state. The TOE does not support Elliptic Curves or Group Extensions. Fallback authentication is not supported for DTLS. If a DTLS connection is broken on the TOE, the TOE will reattempt the connection automatically. An administrator can also attempt to reconnect to the DTLS server from the TOE using the steps found in section 11.

11.1.1.1. X509 Certificate Validation

- X509 Certificate validation happens every DTLS connection attempt for both the server and client
- DTLS connections always use mutual authentication
- The following extendedKeyUsage fields are required in X509 certificates:
    - OCSP signing must be present in the OCSP signing certificate.
    - Server Authentication must be present in any DTLS server certificates.
    - Client Authentication must be present in any DTLS client certificates.
- OCSP is used for checking the revocation status of both leaf and intermediate certificates during DTLS connections
    - Configure OCSP to be turned on using the following commands:
        - **(config)#certmgr trustpoint <trustpoint>**
        - **(config)#validation revocation-check strict**

NOTE: OCSP must be turned on when performing any DTLS connections using this device in order to be operated in a CC compliant state. OCSP checking is performed on all certificates in the presented chain. If a connection cannot be established to the OCSP server, the DTLS connection will be dropped, and the administrator will have to reattempt. No other configuration is needed to place the TOE in the proper operating environment to use the certificates.

11.1.1.2. Configuring client-side certificates for Mutual Authentication

This is a guide for generating and adding certificates to a certificate manager or trustpoint. Currently, certificate manager certificates may only be used in conjunction with SDWAN ('certmgr' setting). The certificate manager feature allows multiple SDWAN interfaces to use the same certificates configured by a single trustpoint, thus simplifying configuration.

The steps to create and apply certificate to a trustpoint are as follows:

1. Create a key pair on the device and give it a label.

Check if there are already keys present on the system by running the command:

```
show ip ssh
```

If there are keys on the system, the public key will be displayed in the format:

```
<key algorithm> <public key in sesch format> <key id>
```

Example:

```
KlasOS# show ip ssh

SSH Enabled - version 2

Keys in OpenSSH format:

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDfJBEQ/NDx0u0idDKwlLrBhSZmmaZDem/GrLgxI+ATgeKdF7qvsdTYZx
HDYBdo2vmQdiCvzBQT92XYrPH5mADxJ
s6ix44o5zEkcNopWm0x3INDheOXMesSbBKGbAwpqcUjpazCpCR4cL/v2ryVwg7d8xxoPs9WEj/STkXIjWM6nRs
7e1Eu4HHT91qI5RRw/OlxzjKvL6oriS
HdC0OUU/5b2eygjK3UoG3ZM8kpbhVnUQPZovfktJCooojsDhRs9ghwN/+xGLBG3jYk1XQJhGO6Pb2KI5QiR4qm
qyZPjVLipWVl1oAgBQSI+PgJr4bPDXy 65j7ClpF4IJIR7HzVBquf clientkey
```

If no keys are present on the system, a key pair must be generated by using the following command (only RSA is supported for now):

```
crypto key generate [ec|rsa] general-keys modulus [2048|3072|4096] label <mylabel>
```

2. Create a certmgr trustpoint on the device.

```
KlasOS# configure terminal
KlasOS(config)# certmgr trustpoint <trustpoint name> RSA
KlasOS(cert-tp-mytrustpoint)#
```

3. Reference the key pair label using the 'key-id' command.

```
KlasOS(cert-tp-mytrustpoint)# key-id <mylabel>
```

4. Give the trustpoint a valid 'subj' field using the 'subj' command.

Example:

```
KlasOS(cert-tp-mytrustpoint)# subj "/C=US/ST=Utah/L=Lehi/O=Your Company,
Inc./OU=IT/CN=yourdomain.com"
```

NOTE: Use quotes if the subject line contains any spaces.

NOTE: The subject line has the following restrictions:

The subject string must begin with a '/'
Fields or Key/value pairs are separated with an '='

Valid keys are:

```
'C', 'ST', 'L', 'O', 'OU', 'CN', 'DC', 'UID', 'countryName',
'stateOrProvinceName', 'localityName', 'organizationName',
'organizationalUnitName', 'commonName', 'emailAddress', 'name'
```

Valid country codes are:

```
'US', 'CA', 'AX', 'AD', 'AE', 'AF', 'AG', 'AI', 'AL', 'AM', 'AN', 'AO', 'AQ', 'AR',
'AS', 'AT', 'AU', 'AW', 'AZ', 'BA', 'BB', 'BD', 'BE', 'BF', 'BG', 'BH', 'BI', 'BJ',
'BM', 'BN', 'BO', 'BR', 'BS', 'BT', 'BV', 'BW', 'BZ', 'CA', 'CC', 'CF', 'CH', 'CI',
'CK', 'CL', 'CM', 'CN', 'CO', 'CR', 'CS', 'CV', 'CX', 'CY', 'CZ', 'DE', 'DJ', 'DK',
'DM', 'DO', 'DZ', 'EC', 'EE', 'EG', 'EH', 'ER', 'ES', 'ET', 'FI', 'FJ', 'FK', 'FM',
'FO', 'FR', 'FX', 'GA', 'GB', 'GD', 'GE', 'GF', 'GG', 'GH', 'GI', 'GL', 'GM', 'GN',
'GP', 'GQ', 'GR', 'GS', 'GT', 'GU', 'GW', 'GY', 'HK', 'HM', 'HN', 'HR', 'HT', 'HU',
'ID', 'IE', 'IL', 'IM', 'IN', 'IO', 'IS', 'IT', 'JE', 'JM', 'JO', 'JP', 'KE', 'KG',
'KH', 'KI', 'KM', 'KN', 'KR', 'KW', 'KY', 'KZ', 'LA', 'LC', 'LI', 'LK', 'LS', 'LT',
'LU', 'LV', 'LY', 'MA', 'MC', 'MD', 'ME', 'MG', 'MH', 'MK', 'ML', 'MM', 'MN', 'MO',
'MP', 'MQ', 'MR', 'MS', 'MT', 'MU', 'MV', 'MW', 'MX', 'MY', 'MZ', 'NA', 'NC', 'NE',
'NF', 'NG', 'NI', 'NL', 'NO', 'NP', 'NR', 'NT', 'NU', 'NZ', 'OM', 'PA', 'PE', 'PF',
'PG', 'PH', 'PK', 'PL', 'PM', 'PN', 'PR', 'PS', 'PT', 'PW', 'PY', 'QA', 'RE', 'RO',
'RS', 'RU', 'RW', 'SA', 'SB', 'SC', 'SE', 'SG', 'SH', 'SI', 'SJ', 'SK', 'SL', 'SM',
'SN', 'SR', 'ST', 'SU', 'SV', 'SZ', 'TC', 'TD', 'TF', 'TG', 'TH', 'TJ', 'TK', 'TM',
'TN', 'TO', 'TP', 'TR', 'TT', 'TV', 'TW', 'TZ', 'UA', 'UG', 'UM', 'US', 'UY', 'UZ',
'VA', 'VC', 'VE', 'VG', 'VI', 'VN', 'VU', 'WF', 'WS', 'YE', 'YT', 'ZA', 'ZM'
```

The following fields can be no longer than 64 characters:

```
'O', 'OU', 'CN', 'organizationName', 'organizationalUnitName', 'commonName',
'emailAddress'
```

The following fields can be no longer than 128 characters:

```
'ST,' 'L','stateOrProvinceName', 'localityName'
```

5. Generate a Certificate Signing Request

Generate a Certificate Signing Request using the 'certmgr generate CSR trustpoint' command.

```
KlasOS# certmgr generate CSR trustpoint <trustpoint name> flash: <CSR filename>
```

6. Provide the newly created CSR to the Certificate Authority

The newly created CSR can now be sent to the Certificate Authority who will then provide your device certificate and the CA certificate. The contents of the CSR can be displayed by running:

```
KlasOS# more flash:
```

Or the CSR file can be copied to another device using the 'copy flash: scp' command.

```
KlasOS# copy flash: scp:

Address of remote host [192.168.3.10]?

Destination username []? user1

Source filename []? device.csr

Destination filename []? device.csr
```

7. Copy the device and CA certificate to flash.

Once the device certificate and the CA's certificate have been provided by the Certificate Authority, copy them to flash using the 'copy scp flash:' command.

```
KlasOS# copy scp: flash:

Address of remote host []? 192.168.3.10

Source username []? user1

Source filename []? device_cert.pem

KlasOS# copy scp: flash:

Address of remote host []? 192.168.3.10

Source username []? user1

Source filename []? CA_cert.pem
```

8. Reference the device and CA certificate in the trustpoint

```
KlasOS# configure terminal

KlasOS(config)# certmgr trustpoint mytruspoint RSA

KlasOS(cert-tp-mytruspoint)# device-cert flash: device_cert.pem

KlasOS(cert-tp-mytruspoint)# ca-cert-chain flash: CA_cert.pem
```

The trustpoint is now ready to be used with SDWAN. It can be used with pki-DTLS transport/encryption modes. Use the 'encryption-mode' command as below when configuring SDWAN

```
KlasOS(config)# interface sdwan 5

KlasOS(config-if-sdwan5)# encryption-mode pki-DTLS AES-GCM

KlasOS(config-if-sdwan5)# certmgr mytruspoint
```

11.2. SDWAN Debug and Troubleshooting Guide

SDWAN configuration involves a set of routing and WAN link selection options. This guide gives an overview of how to debug issues that normally arise when setting up SDWAN for the first time.

11.2.1. Configuration Debug

The following debug options are available through the sdwan configuration for debug:

```
KlasOS(config-if-sdwan0)# debug
  config               SDWAN code level debug
  drops                SDWAN dropped packets debug
  hash-filter-streams  Mapping between flows and hash-filter buckets/tunnels
  protocol             SDWAN protocol negotiation information
  reorder              Packet reordering information
  tunnel               Map physical information
  weights              Round Robin packet scheduling weight information as it is
updated per packet
```

11.2.2. ‘debug config’

This setting results in the internal KlasOS SDWAN function names being printed to the log. This is useful when debugging a particular configuration, as it will show whether SDWAN is fully configured, e.g. after removing the ‘pre-shared-key’ setting and restarting SDWAN, the following messages were seen in the log:

```
Mar  5 12:35:11 KlasOS %SDWAN0: check_min_config()
Mar  5 12:35:11 KlasOS %SDWAN0: pre-shared-key must be configured for encryption-
mode psk sdwan0
Mar  5 12:35:11 KlasOS %SDWAN0:  sdwan0 not fully configured, exiting
Mar  5 12:35:11 KlasOS %SDWAN0: stop_dev()
```

This gives extra information to show that SDWAN is failing the ‘check_min_config’ check and is not fully configured.

11.2.3. WAN Link Selection Debug Counters

The following debug options are available through the sdwan configuraiton for debug:

```
KlasOS(config-if-sdwan0)# debug
  config               SDWAN code level debug
  drops                SDWAN dropped packets debug
  hash-filter-streams  Mapping between flows and hash-filter buckets/tunnels
  protocol             SDWAN protocol negotiation information
  reorder              Packet reordering information
  tunnel               Map physical information
  weights              Round Robin packet scheduling weight information as it is
updated per packet
```

Each of these options result in more logs being generated to the system log. This is not always the easiest way to see SDWAN counter information. Now, (as of release KlasOS v5.3.6) these counters may also be viewed using the

'show sdwan X detail …' command.

11.2.4. show sdwan X detail

The following options are available through 'show sdwan X detail':

```
KlasOS# show sdwan 0 detail
  hash-filter     show map filters information
  filters         show map filters information
  reordering      Show reordering information
  weights         Show link overview
  config          Show configuration summary
  counters        Show tunnel counters
  <cr>
```

Each of these settings may be used to see different counter and settings information for sdwan.

For example, per 'map physical' send/recieve sequence number information may be seen as follows:

```
KlasOS# show sdwan 0 detail counters tunnel con_seq
IFACE: SDWAN0

MAPID: map3_Eth1/1

statistic                              value
---------                              -----
send_seq                                2057
recv_data_seq                              0
recv_seq                               51954
seq_vect                  18446744073709551615
lost_packets                               0
seq_last                               51954

MAPID: map0_Eth1/0

statistic                              value
---------                              -----
send_seq                                2056
recv_data_seq                              0
recv_seq                               51957
seq_vect                  18446744073709551615
lost_packets                               0
seq_last                               51957
```

It can be useful to view this information in real time use the 'continuous' modifier:

```
show sdwan 0 detail counters tunnel con_seq continuous
```

This will result in the counters being displayed in the shell, updating every second:

```
IFACE: SDWAN0

MAPID: map3_Eth1/1

statistic                              value
---------                              -----
send_seq                                2098
recv_data_seq                              0
recv_seq                               51994
seq_vect                  18446744073709551615
lost_packets                               0
seq_last                               51994

MAPID: map0_Eth1/0

statistic                              value
---------                              -----
send_seq                                2097
recv_data_seq                              0
recv_seq                               51997
seq_vect                  18446744073709551615
lost_packets                               0
seq_last                               51997


Hit <ENTER> to exit
```

'**debug sdwan 0 detail counters holdtime continuous**' option is used to see the holdtime countdown as it is occurring.

Example config

```
interface sdwan 0
 map physical 0
  phys ethernet 0/3
  holdtime 10
...

 map physical 1
  phys ethernet 0/2
  holdtime 10
...
```

Example output

```
KlasOS# show sdwan 0 detail counters holdtime
IFACE: SDWAN0

MAPID: map3_Eth1/1
```

```
statistic                              value
---------                              -----
status                                AUTHOK
localholdingtime                           0
holdticks                                  0
onholdtime                                 0
hold_attempts                              1
holdmaxdelay                               4
holdtime                                  10

MAPID: map0_Eth1/0

statistic                              value
---------                              -----
status                                AUTHOK
localholdingtime                           0
holdticks                                  0
onholdtime                                 0
hold_attempts                              1
holdmaxdelay                               4
holdtime                                  10
```

11.2.5. Monitor Capture

11.2.5.1 SDWAN data - 'monitor capture interface sdwan X'

Running 'monitor capture interface sdwan X' can give information on what traffic is being routed over SDWAN. Here IP 192.168.50.1 is being ping'd from the client side with an IP address not matching the 'map client' ip address range on the server, followed by one matching the configuration:

```
KlasOS# monitor capture interface sdwan 0
%INFO: starting packet dump on interface  sdwan0 , Ctrl-C to cancel
09:14:56.801446 ip: 192.168.2.11 > 192.168.50.1: ICMP echo request, id 2680, seq
3, length 64
09:14:57.801579 ip: 192.168.2.11 > 192.168.50.1: ICMP echo request, id 2680, seq
4, length 64
09:14:58.801696 ip: 192.168.2.11 > 192.168.50.1: ICMP echo request, id 2680, seq
5, length 64
09:15:04.892208 ip: 192.168.99.1 > 192.168.50.1: ICMP echo request, id 3960, seq
0, length 64
09:15:05.194344 ip: 192.168.50.1 > 192.168.99.1: ICMP echo reply, id 3960, seq 0,
length 64
09:15:05.892334 ip: 192.168.99.1 > 192.168.50.1: ICMP echo request, id 3960, seq
1, length 64
09:15:06.078591 ip: 192.168.50.1 > 192.168.99.1: ICMP echo reply, id 3960, seq 1,
length 64
```

```
09:15:06.892446 ip: 192.168.99.1 > 192.168.50.1: ICMP echo request, id 3960, seq
2, length 64
09:15:07.078501 ip: 192.168.50.1 > 192.168.99.1: ICMP echo reply, id 3960, seq 2,
length 64
```

11.2.5.2 SDWAN protocol - 'monitor capture interface <map phys iface x/y>'

It is possible to see SDWAN protocol packets using the interfaces that are configured as 'map physical' 'phys' ports.

From the example configuration:

```
KlasOS# show running-config interface sdwan 0
Building configuration...

Current configuration:
!
interface sdwan 0
...
 map physical 0
  bindport 7000
  holdtime 10
  phys ethernet 1/0
  remotehost 192.168.4.12
  remoteport 7000
...
```

Here, sdwan protocol traffic may be seen using:

- 'monitor capture interface Ethernet 1/0'
- 'monitor capture port 7000'
- 'monitor capture host 192.168.4.12'

Example output pre sdwan client connect:

This is also seen if the port is not open on the destination server:

```
SC_CLIENT# show sdwan
Ent  Flag        Proc
---  ----------  ----------
0    down        down

SC_CLIENT# monitor capture port 5000
%INFO: starting packet dump on interface  any , Ctrl-C to cancel
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
12:15:39.128902 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
```

```
12:15:40.128558 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
```

Example SDWAN client connecting:

```
SC_CLIENT# monitor capture port 5000
%INFO: starting packet dump on interface  any , Ctrl-C to cancel
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on any, link-type LINUX_SLL (Linux cooked), capture size 262144 bytes
12:16:32.128883 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
12:16:33.129233 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
12:16:34.129520 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
12:16:35.129034 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
12:16:35.129493  In 00:11:22:ee:ff:b5 ethertype IPv4 (0x0800), length 82:
192.168.4.12.5000 > 192.168.4.11.5000: UDP, length 38
12:16:36.128989 Out 00:11:22:ee:ff:ae ethertype IPv4 (0x0800), length 82:
192.168.4.11.5000 > 192.168.4.12.5000: UDP, length 38
12:16:36.129686  In 00:11:22:ee:ff:b5 ethertype IPv4 (0x0800), length 82:
192.168.4.12.5000 > 192.168.4.11.5000: UDP, length 38
```

12. KlasOS Firewall Introduction

KlasOS allows packet filtering using ACL and interface 'access-group' settings. This is a brief overview of firewall related functionality for KlasOS, and troubleshooting guide. More detailed guides may be added in future in a separate set of helpdesk documents.

12.1. Configuring Firewall rules on an interface

By default, no firewall rules are applied on any interface. To select which rules are applied on a given interface, a mix of ACLs and interface configuration is used. Example configuration:

```
access-list 100 deny icmp any any
access-list 100 permit ip any any
access-list 100 permit ipv6 ip any any

interface Ethernet 0/0
 ip access-group 100 in
 ip address dhcp
In general, once 'ip|ipv6 access-group' is applied on an interface, only traffic
matching 'permit' ACL rules is allowed through.
```

12.2. ACL configuration

'Access Control Lists' are used to select IP or IPv6 flows to match with firewall rules. IPv4 and IPv6 rules are applied together when added to an interface, usually using a command with 'access-group' in the name.

IPv4 access lists have the format:

```
access-list <0-199> <permit|deny|remark> <protocol list> <packet selectors>
<packet match modifiers>
```

IPv6 access lists are generally specified in the same way, but add the 'ipv6' modifier in the place of <ipv4 protocol> above:

```
access-list <0-199> <permit|deny|remark> ipv6 <protocol list> <packet selectors>
<packet match modifiers>
```

N.B. 'protocol list' includes 'ip' for Layer 3 matching, and must be used under 'ipv6' also:

```
access-list 100 permit ipv6 tcp any any eq 8080
access-list 100 permit tcp any any eq 8080
access-list 100 permit ipv6 ip any 2010::10
access-list 100 permit ip any 192.168.0.100
```

12.3. Interface 'ip access-group <ACL num> [in|out]'

'ip access-group' is used to select ACLs to be used to filter interface ingress and egress traffic. Example configuration for denying icmp on ingress, tcp on egress:

```
access-list 100 deny icmp any any
access-list 100 permit ip any any
access-list 100 permit ipv6 ip any any
access-list 101 deny tcp any any
access-list 101 permit ip any any
access-list 101 permit ipv6 ip any any
interface Ethernet 0/0
 ip access-group 100 in
 ip access-group 101 out
 ip address dhcp
```

As can be seen above, matches for IPv4 and IPv6 must be included in the same ACL rule list. ACL num may be re-used for 'in' or 'out'

12.4. Interface 'ip security', 'ipv6 security' settings

'ip security' settings are used to do early (on ingress) or late (on egress) packet matching. They may also be used to do firewall based rate limiting when mixed with ACLs. Example configuration:

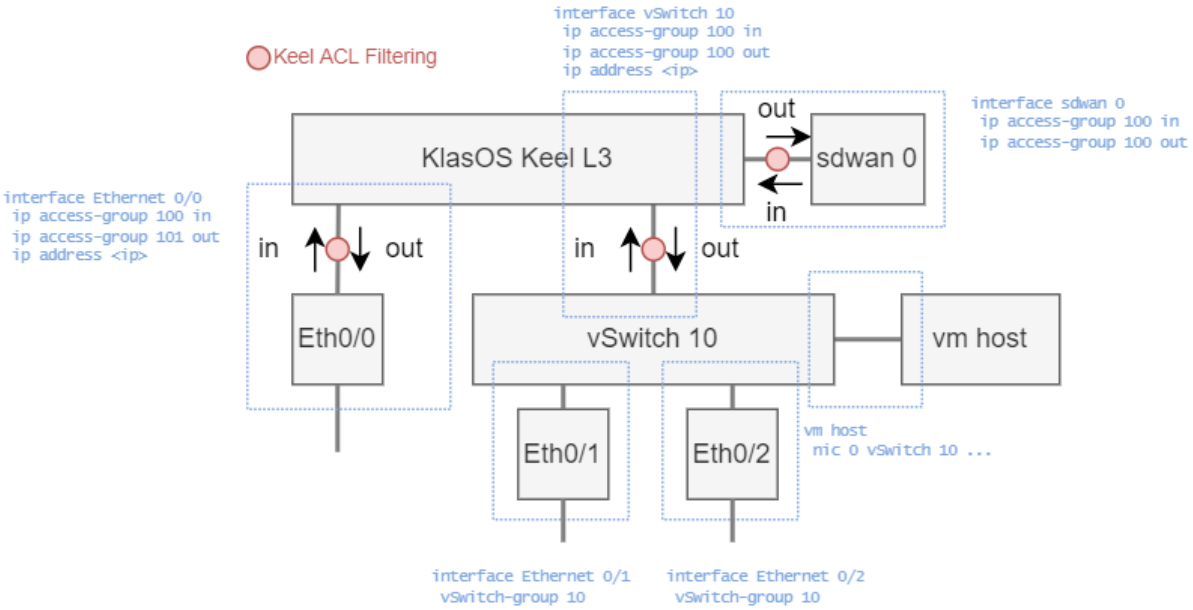
```
access-list 100 deny icmp any any
access-list 100 permit ip any any
access-list 100 permit ipv6 ip any any
access-list 101 deny tcp any any
access-list 101 permit ip any any
access-list 101 permit ipv6 ip any any
...
interface Ethernet 0/2
 ip access-group 100 in
 ip address 192.168.200.10 255.255.255.0
 ip security drop in special-purpose saddr 0x1
 ipv6 address 2010::10/24
 ip security rate-limit access-group 101 in 100 kbytes 1 kbytes
```

One thing to note is that 'rate-limit' only works with 'deny' rules, so in the above configuration, only tcp packets will be rate limited.

12.5. 'vSwitch' interface bridging settings combined with ACL/ip access-group/ip security settings

In general, KlasOS firewall rules take effect at Layer 3 (ip/ipv6) layer of packet processing. As such, when 'vswitch-group' setting is used on an interface, the firewall rules may be bypassed depending on the route of the packet.

The following illustrates when firewall rules may be hit (Keel ACL Filtering):



12.5.1. Firewall rules when interfaces are bridged

The general rules when an interface is bridged are (E.g. interface Eth0/0 on vswitch-group 104 - interface vSwitch 104 bridge).

Example configuration:

```
interface vSwitch 104
 ip access-group 100 in
 ip access-group 100 out
 ip address 10.10.4.80 255.255.255.0
!
...
interface Ethernet 0/0
 vSwitch-group 104

access-list 100 permit tcp host 10.10.4.1 eq 22 any
```

1. When pinging from a separate device to the firewall device:
   -Rules applied to the vswitch are hit in the input and output chains
   -Rules applied to the slaved interface are not hit.
   -when pinging through to a guestOS - rules are bypassed altogether.

2. Rules applied to the vswitch in the forward chain are hit if the packet's source and destination are not the firewall device itself.

12.5.2. Firewall rules when interfaces are not bridged

General rules when the interface is not bridged (E.g. Eth0/0 and vSwitch 104 exist as separate interfaces without the vswitch-group setting).

Example configuration:

```
interface vSwitch 104
  ip address 192.168.102.1 255.255.255.0
!
...
interface Ethernet 0/0
 ip access-group 100 in
 ip access-group 100 out
 ip address 10.10.4.80 255.255.255.0

access-list 100 permit tcp host 10.10.4.1 eq 22 any
```

1. Ingress/egress traffic hit rules applied to the WAN/LAN interface entering/exiting the device (Eth0/0 not the vswitch).
2. Rules in the forward chain are hit (on the WAN/LAN interface) if the packet's source and destination are not the firewall device itself.
3. Packets hit only rules applied to the vswitch in the forward chain when pinging through to a guestOS.

12.5.3. Tips for validating that firewall rules are hit in a certain configuration

KlasOS includes logging and counting for each firewall rule. Counting is always enabled, logging must be added as an ACL parameter. See next section for more information.

Secondly, use of ‘monitor capture' on each interface is advised, including guest OS internal interface:

1. Configure routing/bridging with no firewall rules
2. Generate test traffic corresponding to expected user flows
3. For each relevant interface in the configuration take note of what traffic is seen
4. Begin to add simple firewall rules (e.g. add accept rules to each interface), see which rule counters increment
5. Once you are happy that INPUT, FORWARD and OUTPUT traffic is matching the simple firewall rules in the expected locations, more complex firewall rules should be added.

12.6. Firewall debug - logging and counters for system, 'ip access-group' and 'ip security' settings

It is possible to see the firewall stored configuration and system configuration (and test it) in a number of ways. E.g. it is a good idea to create a set of rules, generate some example traffic, and see whether certain firewall rules are being hit, rather than just falling through to the default DROP for an interface.

N.B. Logging should be enabled as part of an ACL rule setting or 'ip security' setting, however in general it is best to use counters in a live system where possible.

12.6.1. ACLs

```
KlasOS# show access-lists
Extended IP access list 100
Standard IP access list 11
        10 permit 192.168.105.0, wildcard bits 0.0.0.255
KlasOS# show access-lists detail
Extended IP access list 100
Services:
SERVICE ID            APP ID                                    Direction
----------            ------                                    ---------
interface             Eth0/2                                    in
RULES:

Standard IP access list 11
Services:
SERVICE ID            APP ID                                    Direction
----------            ------                                    ---------
interface             sdwan0                                    in
interface             sdwan11                                   in
RULES:
RULE NUM: '10', action: 'permit', IP version: 'IPv4'
    CMD: 'access-list 11 permit 192.168.105.0 0.0.0.255'
    SRC IP: 192.168.105.0/24, ipmode: network, ipmask: 0.0.0.255
    Protocol: IP
```

12.6.2. Access groups (relating to 'ip access-group' settings)

```
KlasOS# show access-group interface sdwan 0
SERVICE ID            APP ID                                    Direction
----------            ------                                    ---------
interface             sdwan0                                    in
ACL Number: 11
RULE: access-list 11 permit 192.168.105.0 0.0.0.255
Interface: sdwan0
Filter Rules
------------
TABLE: filter, CHAIN: INPUT
 IPv4
```

```
pkts       bytes      target   prot   opt    in              out             Source
Destination
table: filter, chain: INPUT  IPv4
0          0          ACCEPT   all    --     sdwan0          *
192.168.105.0/24     0.0.0.0/0
TABLE: filter, CHAIN: FORWARD
 IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
table: filter, chain: FORWARD  IPv4
0          0          ACCEPT   all    --     sdwan0          *
192.168.105.0/24     0.0.0.0/0
Implicit Deny Rules
-------------------
table: filter, chain: INPUT  IPv4
0          0          DROP     all    --     sdwan0          *               0.0.0.0/0
0.0.0.0/0
table: filter, chain: FORWARD  IPv4
0          0          DROP     all    --     sdwan0          *               0.0.0.0/0
0.0.0.0/0
table: filter, chain: INPUT  IPv6
0          0          DROP     all    --     sdwan0          *               ::/0
::/0
table: filter, chain: FORWARD  IPv6
0          0          DROP     all    --     sdwan0          *               ::/0
::/0
```

12.6.3. show ip firewall system|configured

These show 'ip access-group' related rules

```
KlasOS# show ip firewall system
TABLE: filter, CHAIN: INPUT
 IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          ACCEPT   all    --     sdwan0          *
192.168.105.0/24     0.0.0.0/0
0          0          DROP     all    --     sdwan0          *               0.0.0.0/0
0.0.0.0/0
TABLE: filter, CHAIN: FORWARD
 IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          ACCEPT   all    --     sdwan0          *
192.168.105.0/24     0.0.0.0/0
0          0          DROP     all    --     sdwan0          *               0.0.0.0/0
0.0.0.0/0
KlasOS# show ip firewall configured
IPv4 TABLE: filter, CHAIN: INPUT
ACL number: 11
    access-list 11 permit 192.168.105.0 0.0.0.255
```

```
    Rule number: 10, Status: IPT SUCCESS
IPv4 TABLE: filter, CHAIN: INPUT
ACL number: 11
    access-list 11 permit 192.168.105.0 0.0.0.255
    Rule number: 10, Status: IPT store - no iface
IPv4 TABLE: filter, CHAIN: FORWARD
ACL number: 11
    access-list 11 permit 192.168.105.0 0.0.0.255
    Rule number: 10, Status: IPT SUCCESS
IPv4 TABLE: filter, CHAIN: FORWARD
ACL number: 11
    access-list 11 permit 192.168.105.0 0.0.0.255
    Rule number: 10, Status: IPT store - no iface
```

12.6.4. show ip security system|configured

These show 'ip security' related rules

```
KlasOS# show ip security system interface Ethernet 0/2
Net Device rules:

TABLE: netdevfilter_ipv4, CHAIN: Eth0/2_netdevfilter_ipv4
 IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     Eth0/2          *               0.0.0.0/8
*                    Comment:     ipv4:saddr:'This network'  [RFC791], Section 3.2
KlasOS# show ip security configured interface Ethernet 0/2
IPv4 Security Rules for interface Eth0/2
'ip security drop in special-purpose saddr' Enabled
MASK: 0x1
(*) - selected, (_) - not selected
 0x1        (*)   (0.0.0.0/8) 'This network'  [RFC791], Section 3.2
 0x2        (_)   (0.0.0.0/32) 'This host on this network' [RFC1122], Section 3.2.1.3
 0x4        (_)   (10.0.0.0/8) Private-Use [RFC1918]
 0x8        (_)   (100.64.0.0/10) Shared Address Space    [RFC6598]
 0x10       (_)   (127.0.0.0/8) Loopback    [RFC1122], Section 3.2.1.3
 0x20       (_)   (169.254.0.0/16) Link Local  [RFC3927]
 0x40       (_)   (172.16.0.0/12) Private-Use [RFC1918]
 0x80       (_)   (192.0.0.0/24) IETF Protocol Assignments   [RFC6890], Section 2.1
 0x100      (_)   (192.0.0.0/29) IPv4 Service Continuity Prefix  [RFC7335]
 0x200      (_)   (192.0.0.8/32) IPv4 dummy address  [RFC7600]
 0x400      (_)   (192.0.0.9/32) Port Control Protocol Anycast   [RFC7723]
 0x800      (_)   (192.0.0.10/32) Traversal Using Relays around NAT Anycast
[RFC8155]
 0x1000     (_)   (192.0.0.170/32) NAT64/DNS64 Discovery   [RFC8880][RFC7050], Section
2.2
 0x2000     (_)   (192.0.0.171/32) NAT64/DNS64 Discovery   [RFC8880][RFC7050], Section
2.2
 0x4000     (_)   (192.0.2.0/24) Documentation (TEST-NET-1)  [RFC5737]
 0x8000     (_)   (192.31.196.0/24) AS112-v4    [RFC7535]   2014-12
 0x10000    (_)   (192.52.193.0/24) AMT [RFC7450]   2014-12
```

```
 0x20000    (_)   (192.88.99.0/24) Deprecated (6to4 Relay Anycast) [RFC7526]
 0x40000    (_)   (192.168.0.0/16) Private-Use [RFC1918]
 0x80000    (_)   (192.175.48.0/24) Direct Delegation AS112 Service [RFC7534]
 0x100000   (_)   (198.18.0.0/15) Benchmarking    [RFC2544]
 0x200000   (_)   (198.51.100.0/24) Documentation (TEST-NET-2)  [RFC5737]
 0x400000   (_)   (203.0.113.0/24) Documentation (TEST-NET-3)  [RFC5737]
 0x800000   (_)   (240.0.0.0/4) Reserved    [RFC1112], Section 4
 0x1000000  (_)   (255.255.255.255/32) Limited Broadcast   [RFC8190] [RFC919], Section
7
 0x2000000  (_)   (224.0.0.0/4) Multicast addresses
 0x4000000  (_)   pkt saddr eq iface local address (ignored for daddr)
 0x8000000  (_)   pkt saddr eq iface broadcast address (ignored for daddr)
```

12.7. Troubleshooting

Here configured ACLs to match drop ICMP traffic, but now no other traffic is getting past the firewall rules

Once an ACL is applied to an interface, a default DROP policy is applied for that interface. To allow certain traffic, through, an ACL rule must be present which matches that traffic. Example configuration to drop all ICMP ingress packets, but allow all other packets:

```
access-list 100 deny icmp any any
access-list 100 permit ip any any
access-list 100 permit ipv6 ip any any

interface Ethernet 0/0
 ip access-group 100 in
 ip address dhcp
```

Note that ‘ipv6’ flows must also be selected for ‘permit’ or they will also be dropped.

13. ACL guide for KlasOS Firewall

There are two main types of filter selection:

- (Standard) Access list
- Extended Access list

If these are applied to an interface or service, an implicit deny rule is also added for both IPv4 and IPv6 traffic, even if the applied filters only pertain to one of those protocols.

N.B. *Only these rules should be used for Firewall PP conformance, as described below. All network interfaces on the TOE have the ability to perform packet filtering on packets being received or sent to the external network. Due to this, there is only one distinct network interface type where firewall rules can be configured.*

13.1. Standard Access lists

Standard Access lists pertain to IPv4 configuration only and are a simpler form of filter config. The are selected with access-list number 1-99.

The command format is:

```
KlasOS(config)# access-list [1-99] [permit|deny] [source host IPv4 address|'any'>
<source host wildcard bits> <log>
```

N.B. For the Firewall, we use 'permit' and 'deny' rules. 'bypass' is used to route traffic beside an ipsec tunnel on the same interface. I.e. the traffic is 'permitted' out the interface, but is not encrypted. In the context of the Firewall it has the same effect as 'permit'

An example is:

```
KlasOS(config)# access-list 1 permit 192.168.102.1 0.0.0.255 log
```

The result of this access list is to permit all traffic from '192.168.102.1' and log matches to this rule.

13.2. Extended Access lists

These access lists may be used to filter IPv4 and IPv6 traffic. They are also capable of filtering TCP, UDP, ICMP and other IP protocols. They are selected by using access-list number 100-199.

The below is an overview - there are too many combinations to go into, but Extended access lists allow filtering based on source/destination ipv4/ipv6 address, source/destination udp/tcp ports, along with other layer 4 algorithms, including icmp and icmpv6

13.2.1. Extended Access List command format

The following discusses settings for Extended ACLs

Layer 4 filtering

```
KlasOS(config)# access-list
KlasOS(config)# access-list [100-200] [permit|deny]
  <0-255>  An IP protocol number (0..255)
  esp      Encapsulation Security Payload
  icmp     Internet Control Message Protocol
  igmp     Internet Gateway Message Protocol
  ip       Any Internet Protocol
  ipv6     Any Internet Protocol Version 6
  ospf     OSPF routing protocol
  tcp      Transmission Control Protocol
  udp      User Datagram Protocol
```

After selecting 'permit' or 'deny', a mix of layer 3 and layer 4 protocols are available for selection. Here, the listed layer 4 protocols (esp, icmp, igmp, ospf, tcp, udp, <protocol numbers>) relate to ipv4 packets only.

IPv4 example:

```
access-list 100 permit ipv4 udp 192.168.100.0/24 eq 8080 any log
```

This access list permits accesses to UDP port 8080 from hosts in the CIDR domain 192.168.100.0/24, and to any ipv4 destination host. If the rule is matched, the result is logged.

Selecting Layer 4 filters for IPv6:

```
KlasOS(config)# access-list 100 permit ipv6
  <0-255>  An IP protocol number (0..255)
  esp      Encapsulation Security Payload
  icmpv6   Internet Control Message Protocol Version 6
  ip       Any Internet Protocol
  ospf     OSPF routing protocol
  tcp      Transmission Control Protocol
  udp      User Datagram Protocol
```

IPv6 example:

```
access-list 100 permit ipv6 udp 2001::1/64 eq 8080 any log
```

This access list permits access to UDP port 8080 from hosts in the CIDR domain 2001::1/64, and to any ipv6 destination host. If the rule is matched, the result is logged.

Layer 3 filtering

Layer 3 protocols are special cases:

- 'ip' filter all ipv4 traffic based on IP header options
- 'ipv6' - goes to menu of similar options, but for ipv6

Example for IPv4:

```
KlasOS(config)# access-list 100 permit ip
  any      Any source host
  host     A single source host
  A.B.C.D  Source address
```

Example for IPv6:

```
KlasOS(config)# access-list 100 permit ipv6 ip
  any                 Any source host
  host                A single source host
  A:B:C:D::H          Source address
  X:X:X:X::X/<1-128>  Source address and CIDR
```

13.3. Access list logging

The access-list logging feature provides the ability to log messages about packets that are permitted or denied by either a standard or an extended IP access list. Any packet that matches the access list rules causes an information log message about the packet to be sent to the system log. Log messages include information about the access list number, the source and destination IP address and ports of packets and the incoming and/or outgoing interface. Access-lists are assigned to a distinct network interface by the administrator.

IPv4 Example:

```
KlasOS(config)# access-list 102 permit tcp host 192.168.1.10 host 192.168.20.1 log
KlasOS(config)# interface sdwan 10
KlasOS(config-if-sdwan10)# ip access-group 102 out
```

IPv6 Example:

```
KlasOS(config)# access-list 101 permit ipv6 tcp host 2004::2 host 2003::11 log
KlasOS(config)# interface eth0/3
KlasOS(config-if)# ip access-group 101 out
```

Access list logging can be viewed with the command 'show logging firewall [full|continuous]'

Example of ACL log message:

2021-09-27T15:35:39.915962+00:00 sdwan_server kernel: [13194.859043] %SEC-5-ACL: list 101 permit IN=sdwan10 OUT=eth3 MAC= SRC=192.168.1.2 DST=192.168.3.11 LEN=60 TOS=0x00 PREC=0x00 TTL=63 ID=15372 DF PROTO=TCP SPT=35164 DPT=22 WINDOW=64584 RES=0x00 SYN URGP=0

12.3.1 Other Access List settings

12.3.1.1. access-list remark (add a description for the access list)

There is also the possibility to add a remark for the access list

```
KlasOS(config)# access-list [1-199] remark "this access-list permits all traffic on
the company intranet only"
```

14. Connection tracking in KlasOS Firewall using ACLs

Connection tracking is possible in KlasOS using a mix of ACL and interface configurations. Below is an overview of updates for connection tracking in KlasOS

14.1. Configuration

The updates extend the existing 'extended' ACL configuration set.

Extended ACL format:

```
access-list <100-199> <action> <protocol> <ip options, tcp/udp ports> <protocol
options>
access-list <100-199> <action> ipv6 <protocol> <ip options, tcp/udp ports> <protocol
options>
```

This update will add 'conn' to the list of protocol options to enable connection tracking rules:

```
KlasOS(config)# access-list 100 permit tcp any any conn
  (INVALID|ESTABLISHED|NEW|RELATED|UNTRACKED|SNAT|DNAT)  Any combination of the list
in double quotes and separated with a comma

KlasOS(config)# access-list 100 permit ipv6 tcp any any conn
  (INVALID|ESTABLISHED|NEW|RELATED|UNTRACKED|SNAT|DNAT)  Any combination of the list
in double quotes and separated with a comma
```

Each of these options can be used with 'permit' or 'deny'

conn option	description
Established	match established icmp/tcp/udp sessions
New	match new sessions
Related	FTP, SIP sessions open new ports for the control plane. If 'related' is selected, then these new ports will also match the rule. E.g. This may be used to allow packets from established SIP and FTP sessions to avoid further firewall rule processing
Invalid	Not related to a known connection
Snat	A virtual state, matching if the original source address differs from the reply destination.
Dnat	A virtual state, matching if the original destination differs from the reply source

14.1.1. New chain modifiers of the existing command set

The following extra modifiers have been added on top of the existing command set:

Name	Comments
Raw	Raw table may be used to match packet fragments etc before the packets get to the conntrack module. It only works with PREROUTING and OUTPUT rules, selected by 'ip access-group in' or 'out'. **N.B. 'conn' cannot be selected as a sub-option here** **N.N.B. This rule is applied pre-packet mangling/nat for output rules**
input-only	Apply rule to the input table only **N.B. This flag is ignored if 'ip access-group out' is selected, but a warning is printed.**
forwarding-only	Apply rule to the forwarding table only **N.B. This flag is ignored if 'ip access-group out' is selected, but a warning is printed.**
output-only	cannot be selectedcannot be selectedcannot be selectedcannot be selectedAdd rule to filter traffic on output from the interface only (rule is not added to the forward table)

Updated ACL format (new options are optional, old behavior is used without those flags):

```
access-list <100-199> <action> <protocol> <ip options, tcp/udp ports> <protocol
options> <raw/input/forward>
access-list <100-199> <action> ipv6 <protocol> <ip options, tcp/udp ports> <protocol
options> <raw/input/forward>
```

14.2. Show commands

Show open TCP connections through the TOE

```
KlasOS# show conn protocol tcp
ipv4 2 tcp 6 431345 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52416
src=192.168.3.11 dst=192.168.3.2 sport=52416 dport=22 [ASSURED] secctx=null use=2
ipv4 2 tcp 6 431017 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52420
src=192.168.3.11 dst=192.168.3.2 sport=52420 dport=22 [ASSURED] secctx=null use=2
```

Show open UDP connections through the TOE

```
KlasOS# show conn protocol udp
ipv4 2 udp 17 9 src=192.168.3.2 dst=192.168.3.11 sport=54895 dport=514
[UNREPLIED] src=192.168.3.11 dst=192.168.3.2 sport=514 dport=54895
secctx=null use=2
ipv4 2 udp 17 5 src=127.0.0.1 dst=127.0.0.1 sport=46921 dport=53 [UNREPLIED]
src=127.0.0.1 dst=127.0.0.1 sport=53 dport=46921 secctx=null use=2
```

Show session states

Connections can be filtered by states (NEW/ESTABLISHED/RELATED/INVALID)

```
KlasOS# show conn state ESTABLISHED
ipv4 2 tcp 6 430335 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52416
src=192.168.3.11 dst=192.168.3.2 sport=52416 dport=22 [ASSURED] secctx=null use=2
ipv4 2 tcp 6 430007 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52420
src=192.168.3.11 dst=192.168.3.2 sport=52420 dport=22 [ASSURED] secctx=null use=2

KlasOS# show conn protocol tcp state ESTABLISHED
ipv4 2 tcp 6 430180 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52416
src=192.168.3.11 dst=192.168.3.2 sport=52416 dport=22 [ASSURED] secctx=null use=2
ipv4 2 tcp 6 429851 ESTABLISHED src=192.168.3.2 dst=192.168.3.11 sport=22 dport=52420
src=192.168.3.11 dst=192.168.3.2 sport=52420 dport=22 [ASSURED] secctx=null use=2
```

15. Firewall 'ip|ipv6 security' settings

Some firewall requirements can be said to be standalone in that there is a specific way to meet the requirement and rules based on the requirement don't need to be possibly combined with other flow attributes since they talk about invalid IP addressing. 'ip|ipv6' security settings allows:

- selecting these rules for early matching,
- reverse path routing check for interfaces
- early packet fragment matching for ipv4 and ipv6
- rate-limiting ACL selected flows

Note: The TOE stores ACL rules in the order they were configured. This is the same order the rules are checked when checking an inbound or outbound packet. If a packet is sent through the TOE that does not match the ruleset configured, that packet will be dropped.

15.1. Settings overview

Each of these settings are per interface, so you might enable them as follows:

```
KlasOS(config)# interface Eth0/2
KlasOS(config-if)# ip security
  drop    drop packets with invalid source address
  verify  Drop packets with invalid reverse path route
KlasOS(config-if)# ip security verify reverse-path
  log  log reverse-pathped invalid packets
  <cr>

KlasOS(config-if)# ip security drop [in|out] special-purpose [saddr|daddr]
  show           Show the result of a given mask combination (up to) 0xffffffff
  0x0-0xFFFFFFF  Combine below masks up to 0xffffffff to select below options
  0x0            Block all special-purpose addresses
  0x1            (0.0.0.0/8) 'This network'  [RFC791], Section 3.2
  0x2            (0.0.0.0/32) 'This host on this network' [RFC1122], Section 3.2.1.3
  0x4            (10.0.0.0/8) Private-Use [RFC1918]
...

KlasOS(config-if)# ip security drop in special-purpose saddr 0x0
  log  log dropped invalid packets
  <cr>

KlasOS# show ip security configured interface Ethernet 0/2
IPv4 Security Rules for interface Eth0/2
'ip security verify reverse' path Enabled


KlasOS(config-if)# ipv6 security verify reverse-path
  log  log reverse-pathped invalid packets
```

```
  <cr>
KlasOS(config-if)# ipv6 security drop [in|out]
exthdrs         special-purpose
KlasOS(config-if)# ipv6 security drop in exthdrs
  show      Show the result of a given mask combination (up to) 0xff
  0x0-0xFF  Combine below masks up to 0xff to select below options
  0x0       Block all ipv6 extension headers
  0x1       (0)  Hop by Hop header
  0x2       (60) Destination Options header
  0x4       (43) Routing Options header
  0x8       (44) Fragmentation Options header
  0x10      (135) Mobility Options header
  0x20      (51) Auth Option (nexthdr match)
  0x40      (50) ESP Option (nexthdr match)
  0x80      (59) No next header Option (nexthdr match)

KlasOS(config-if)# ipv6 security drop in exthdrs 0x0
  log  log dropped invalid packets
  <cr>

KlasOS(config-if)# ip security rate-limit access-group 100 in
  <1-10240>  bytes count (1..10240)
  <cr>

KlasOS# show ipv6 security configured interface Ethernet 0/2
IPv6 Security Rules for interface Eth0/2
'ipv6 security drop inexthdrs' Enabled
MASK: 0x0
 0x1        (*)   (::1/128) Loopback Address       [RFC4291]
 0x2        (*)   (::/128) Unspecified Address    [RFC4291]
 0x4        (*)   (::ffff:0:0/96) IPv4-mapped Address    [RFC4291]
 0x8        (*)   (64:ff9b::/96) IPv4-IPv6 Translat.    [RFC6052]
 0x10       (*)   (64:ff9b:1::/48) IPv4-IPv6 Translat.    [RFC8215]
 0x20       (*)   (100::/64) Discard-Only Address Block    [RFC6666]
 0x40       (*)   (2001::/23) IETF Protocol Assignments     [RFC2928]
 0x80       (*)   (2001::/32) TEREDO    [RFC4380] [RFC8190]
...
```

Each rule has an optional 'log' prefix. All firewall rules can be applied to the TOE using the local or remote CLI interface. When logging in via a local interface, there will be a prompt to "Press RETURN to get started."

Setting	Firewall Requirements	Comment
ip|ipv6 security verify reverse-path log <prefix>	• packets with source address on the broadcast (ipv4) network for the interface should be dropped • packets with source address equal to the interface should be dropped • Packets with source address which cannot be routed back through the interface should be dropped	broadcast address doesn't exist for ipv6 (multicast is used). For the routing requirement, the KlasOS routing table is looked up to verify whether the packet is routable back out the interface. If a default route is present for the interface, then the packet will not be dropped.
ip|ipv6 security drop [in|out] special-purpose [saddr|daddr] [mask] log <prefix>	• packets with multicast source address • link local match • loopback match • 'unspecified', reserved for future use (ipv4) • 'unspecified', reserved for future use (ipv6)	[in|out] - block ingress or egress traffic [saddr|daddr] - packet source or destination address is matched [mask] - select iana/multicast networks to block (more info below)
ipv6 security drop in exthdrs [mask] log <prefix> log <prefix>	• drop packets with ipv6 extension headers, including fragmentation header	Blocks ipv6 extension headers. IPv6 extension headers may be nested within the packet header and multiple extension types may be in the same packet. This has been added to match packet fragment headers for IPv6 since using 'conntrack' negates catching them in the iptables INPUT chain, however other extension headers are also possible to be matched.

access-list 100 permit tcp any any eq 80 security-rate-limit <bytes> <units> <burst> <units> ip security rate-limit access-group 100 [in|out]<bytes> <units> <burst> <units>	• Stateless firewall rate limiting for selected traffic	Adds an ACL based for rate limiting • only some ACL parameters are supported ○ unsupported params are logged on rate limit setup • ACL 'security-rate-limit' values override 'ip security rate-limit' defaults access-list 'security-rate-limit' is not supported by any other service (interface/http/etc.)
show [ip|ipv6] security interface <iface type/name>	• Ability to see rule counters being hit per interface	This output can be quite large even for a small number of settings.
ip security drop in fragments <prefix>	• Firewall rule to drop packets that are invalid fragments • Firewall rule to drop fragments that cannot be completely re-assembled	This has been added to filter out any fragmented packets that may or may not be part of a session
access-list 100 deny ip any any advanced raw ipv4options [option] <prefix>	• Firewall rule to filter ipv4 traffic with loose source, strict source and record routing	[option] – specifies which type of routing to filter 3 = loose source 7 = record route 9 = strict source routing

15.2. ip|ipv6 security verify reverse-path log <prefix>

```
KlasOS(config-if)# ip security verify reverse-path
KlasOS(config-if)# do show ip security
Net Device rules:
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          drop     all    ==     Eth0/2          *
192.168.200.10, 192.168.200.255 *                    Comment:     Eth0/2:ipv4:invalid
pkt saddr
0          0          drop     all    ==     Eth0/2          *               *
*                     Comment:     Eth0/2:ipv4: verify reverse path route
```

15.3. ip|ipv6 security drop [in|out] special-purpose [saddr|daddr] [mask] log <prefix>

Most of these are based on 'iana special purpose registry' networks, (except multicast):

```
KlasOS(config-if)# ip security drop in special-purpose saddr
  0x0-0x3FFFFFF  Value to AND with 0x3ffffff to choose a combination of options
  0x0            Block all special-purpose addresses
  0x1            'This network'  [RFC791], Section 3.2
  0x2            'This host on this network' [RFC1122], Section 3.2.1.3
  0x4            Private-Use [RFC1918]
  0x8            Shared Address Space    [RFC6598]
  0x10           Loopback    [RFC1122], Section 3.2.1.3
...
```

Example just selecting one option:

```
KlasOS(config-if)# ip security drop in special-purpose saddr 0x80
KlasOS(config-if)# do show ip firewall
configured system
KlasOS(config-if)# do show ip firewall system
table   security  verbose
KlasOS(config-if)# do show ip security interface Ethernet 0/2
Net Device rules:
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          drop     all    ==     Eth0/2          *
192.0.0.0/24         *                    Comment:     ipv4:saddr:IETF Protocol
Assignments   [RFC6890], Section 2.1
```

It is possible to see which masked bits will be selected using the following:

```
KlasOS(config-if)# ip security drop in special-purpose saddr show 0x8f
MASK: 0x8f
(*) - selected, (_) - not selected
 0x1        (*)   (0.0.0.0/8) 'This network'  [RFC791], Section 3.2
 0x2        (*)   (0.0.0.0/32) 'This host on this network' [RFC1122], Section 3.2.1.3
 0x4        (*)   (10.0.0.0/8) Private-Use [RFC1918]
 0x8        (*)   (100.64.0.0/10) Shared Address Space    [RFC6598]
 0x10       (_)   (127.0.0.0/8) Loopback    [RFC1122], Section 3.2.1.3
 0x20       (_)   (169.254.0.0/16) Link Local  [RFC3927]
 0x40       (_)   (172.16.0.0/12) Private-Use [RFC1918]
 0x80       (*)   (192.0.0.0/24) IETF Protocol Assignments   [RFC6890], Section 2.1
 0x100      (_)   (192.0.0.0/29) IPv4 Service Continuity Prefix  [RFC7335]
 0x200      (_)   (192.0.0.8/32) IPv4 dummy address  [RFC7600]
 0x400      (_)   (192.0.0.9/32) Port Control Protocol Anycast   [RFC7723]
 0x800      (_)   (192.0.0.10/32) Traversal Using Relays around NAT Anycast
[RFC8155]
...
```

15.4. ipv6 security drop in exthdrs [mask] log <prefix> log <prefix>

ipv6 extension headers to match (frag is only matchable pre conntrack module, and not possible to match with standard iptables rules):

```
KlasOS(config-if)# ipv6 security drop in exthdrs
  0x0-0xFF  Value to AND with 0xff to choose a combination of options
  0x0       Block all ipv6 extension headers
  0x1       (0)  Hop by Hop header
  0x2       (60) Destination Options header
  0x4       (43) Routing Options header
  0x8       (44) Fragmentation Options header
  0x10      (135) Mobility Options header
  0x20      (51) Auth Option (nexthdr match)
  0x40      (50) ESP Option (nexthdr match)
  0x80      (59) No next header Option (nexthdr match)
```

Pre configuration check:

```
KlasOS(config-if)# ipv6 security drop in exthdrs show 0x11
 0x1        (*)   (0)  Hop by Hop header
 0x2        (_)   (60) Destination Options header
 0x4        (_)   (43) Routing Options header
 0x8        (_)   (44) Fragmentation Options header
 0x10       (*)   (135) Mobility Options header
 0x20       (_)   (51) Auth Option (nexthdr match)
 0x40       (_)   (50) ESP Option (nexthdr match)
 0x80       (_)   (59) No next header Option (nexthdr match)
```

15.5. ip security rate-limit access-group [in|out] <bytes count><bytes units> <burst count> <burst units>

15.5.1 Settings

This service adds firewall based rate limiting to KlasOS. The related settings are as follows:

```
access-list 100 permit tcp any any eq www security-rate-limit 1000 kbytes 10 bytes
access-list 100 permit tcp any any eq 8080
access-list 100 permit ipv6 tcp any any eq www security-rate-limit 300 kbytes 10 bytes
interface Ethernet 0/2
 ip address 192.168.2.11 255.255.255.224
 ip security rate-limit access-group 100 in 2000 Kbytes 100 bytes
```

In the above configuration, for the first rule is set such that:

- a maximum rate of 1000 KBytes matching the rule is allowed per second
- a 10 byte burst of data over that limit is allowed within a second

It is possible to configure the rate limit in the ACL and the ‘access-group’ setting. ACL overrides the access-group. If no rate limit is set between the ACL and the access-group, the rule is skipped. Similarly for invalid rate-limit/ACL combinations, the rule is skipped. Finally, if ‘security-rate-limit’ is used with ‘ip access-group’ or other services, the rule will be skipped.

Similarly to ‘ip access-group’, if ‘in’ is selected, rules are added to INPUT and FORWARD filter chains. If ‘out’ is selected, rules are added to FORWARD and OUTPUT filter chains.

It is important to note that rate limit rules are matched BEFORE ‘ip access-group' rules, with one exception. In the ‘out’ direction if ‘advanced raw' is selected, the rate limit rule is added to a POSTROUTING chain, meaning that it will be checked after ‘ip access-group’ OUTPUT chain rules have been hit.

N.B. It is possible to configure a low enough rate limit/burst combination such that no traffic will flow.

15.5.2 Show commands

15.5.2.1 show access-group security-rate-limit

```
KlasOS(config)# do show running-config | include acc
 ip security rate-limit access-group 100 in 256 kbytes 100 bytes
 wifi mode access-point
access-list 100 deny tcp any any eq 8080
access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512 kbytes
200 bytes
access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100
kbytes

KlasOS(config)# do show access-group security-rate-limit
SERVICE ID            APP ID                                    Direction
----------            ------                                    ---------
security_rate_limit   Eth0/2                                    in
ACL Number: 100
RULE: access-list 100 deny tcp any any eq 8080
Interface: Eth0/2
Security Rate Limit Rules
-------------------------
TABLE: filter, CHAIN: input  IPv4
pkts       bytes      target   prot   opt    in              out
Source               Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100
TABLE: filter, CHAIN: forward  IPv4
pkts       bytes      target   prot   opt    in              out
Source               Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100
```

```
RULE: access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512
kbytes 200 bytes
Interface: Eth0/2
Security Rate Limit Rules
-------------------------
TABLE: filter, CHAIN: input  IPv6
pkts       bytes      target   prot   opt    in             out
Source               Destination
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate kbytes/s 512, burst bytes/s 200
TABLE: filter, CHAIN: forward  IPv6
pkts       bytes      target   prot   opt    in             out
Source               Destination
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate kbytes/s 512, burst bytes/s 200
RULE: access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100
kbytes
Interface: Eth0/2
Security Rate Limit Rules
-------------------------
TABLE: filter, CHAIN: input  IPv4
pkts       bytes      target   prot   opt    in             out
Source               Destination
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100
TABLE: filter, CHAIN: forward  IPv4
pkts       bytes      target   prot   opt    in             out
Source               Destination
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2         *              *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100
```

15.5.2.2 show [ip|ipv6] security configured

```
KlasOS(config-if)# do show running-config | include sec
username klas secret 5 $1$WESyhCSx$bSeieQmzXgd9P7rygaY1P0
 ip security drop in special-purpose saddr 0x1F
 ip security rate-limit access-group 100 in 256 kbytes 100 bytes
access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512 kbytes
200 bytes
access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100
kbytes
ip http secure-server


KlasOS(config-if)# do show ipv6 security configured
IPv6 Security Rules for interface Eth0/2
 TABLE: filter, CHAIN: INPUT
ACL number: 100
```

```
    access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512
kbytes 200 bytes
    Rule number: 20, Status: IPT SUCCESS
 TABLE: filter, CHAIN: FORWARD
ACL number: 100
    access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512
kbytes 200 bytes
    Rule number: 20, Status: IPT SUCCESS


KlasOS(config-if)# do show ip security configured
IPv4 Security Rules for interface Eth0/2
 TABLE: filter, CHAIN: INPUT
ACL number: 100
    access-list 100 deny tcp any any eq 8080
    Rule number: 10, Status: IPT SUCCESS
ACL number: 100
    access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100
kbytes
    Rule number: 30, Status: IPT SUCCESS
 TABLE: filter, CHAIN: FORWARD
ACL number: 100
    access-list 100 deny tcp any any eq 8080
    Rule number: 10, Status: IPT SUCCESS
ACL number: 100
    access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100
kbytes
    Rule number: 30, Status: IPT SUCCESS
'ip security drop in special-purpose saddr' Enabled
MASK: 0x1F
(*) - selected, (_) - not selected
 0x1        (*)   (0.0.0.0/8) 'This network'  [RFC791], Section 3.2
 0x2        (*)   (0.0.0.0/32) 'This host on this network' [RFC1122], Section
3.2.1.3
 0x4        (*)   (10.0.0.0/8) Private-Use [RFC1918]
 0x8        (*)   (100.64.0.0/10) Shared Address Space    [RFC6598]
 0x10       (*)   (127.0.0.0/8) Loopback    [RFC1122], Section 3.2.1.3
 0x20       (_)   (169.254.0.0/16) Link Local  [RFC3927]
 0x40       (_)   (172.16.0.0/12) Private-Use [RFC1918]
 0x80       (_)   (192.0.0.0/24) IETF Protocol Assignments   [RFC6890], Section
2.1
 0x100      (_)   (192.0.0.0/29) IPv4 Service Continuity Prefix  [RFC7335]
 0x200      (_)   (192.0.0.8/32) IPv4 dummy address  [RFC7600]
 0x400      (_)   (192.0.0.9/32) Port Control Protocol Anycast   [RFC7723]
 0x800      (_)   (192.0.0.10/32) Traversal Using Relays around NAT Anycast
[RFC8155]
 0x1000     (_)   (192.0.0.170/32) NAT64/DNS64 Discovery   [RFC8880][RFC7050],
Section 2.2
...
```

15.5.2.3 show ip security system

```
KlasOS(config-if)# do show running-config | include sec
username klas secret 5 $1$WESyhCSx$bSeieQmzXgd9P7rygaY1P0
 ip security drop in special-purpose saddr 0x1F
 ip security rate-limit access-group 100 in 256 kbytes 100 bytes
access-list 100 deny ipv6 tcp any any eq 8081 security-rate-limit 512 kbytes 200 bytes
access-list 100 deny udp any any eq 5120 security-rate-limit 1 mbytes 100 kbytes
ip http secure-server

KlasOS(config-if)# do show ip security system
Net Device rules:
TABLE: netdevfilter_ipv4, CHAIN: Eth0/2_netdevfilter_ipv4  IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     *               *               0.0.0.0/8
*                     Comment:     ipv4:saddr:'This network'  [RFC791], Section 3.2
0          0          DROP     all    ==     *               *               0.0.0.0
*                     Comment:     ipv4:saddr:'This host on this network' [RFC1122],
Section 3.2.1.3
0          0          DROP     all    ==     *               *
10.0.0.0/8           *                    Comment:     ipv4:saddr:Private-Use
[RFC1918]
0          0          DROP     all    ==     *               *
100.64.0.0/10        *                    Comment:     ipv4:saddr:Shared Address
Space    [RFC6598]
0          0          DROP     all    ==     *               *
127.0.0.0/8          *                    Comment:     ipv4:saddr:Loopback
[RFC1122], Section 3.2.1.3
TABLE: filter_ipv4, CHAIN: input  IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100
TABLE: filter_ipv4, CHAIN: forward  IPv4
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 256, burst bytes/s 100
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate mbytes/s 1, burst kbytes/s 100

KlasOS(config-if)# do show ipv6 security system
Net Device rules:
TABLE: filter_ipv6, CHAIN: input  IPv6
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 512, burst bytes/s 200
TABLE: filter_ipv6, CHAIN: forward  IPv6
```

```
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          DROP     all    ==     Eth0/2          *               *
*                     Comment:     rate kbytes/s 512, burst bytes/s 200
```

15.6. Example output for 'show ip security system'

```
KlasOS(config-if)# do show ipv6 security system interface Ethernet 0/2
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          drop     all    ==     Eth0/2          *               ::1
*                     Comment:     ipv6:saddr:Loopback Address       [RFC4291]
0          0          drop     all    ==     Eth0/2          *               ::
*                     Comment:     ipv6:saddr:Unspecified Address    [RFC4291]
0          0          drop     all    ==     Eth0/2          *
::ffff:0.0.0.0/96     *                    Comment:     ipv6:saddr:IPv4-mapped Address
[RFC4291]
0          0          drop     all    ==     Eth0/2          *
64:ff9b::/96          *                    Comment:     ipv6:saddr:IPv4-IPv6 Translat.
[RFC6052]
0          0          drop     all    ==     Eth0/2          *
64:ff9b:1::/48        *                    Comment:     ipv6:saddr:IPv4-IPv6 Translat.
[RFC8215]
0          0          drop     all    ==     Eth0/2          *               100::/64
*                     Comment:     ipv6:saddr:Discard-Only Address Block    [RFC6666]
0          0          drop     all    ==     Eth0/2          *               2001::/23
*                     Comment:     ipv6:saddr:IETF Protocol Assignments     [RFC2928]
0          0          drop     all    ==     Eth0/2          *               2001::/32
*                     Comment:     ipv6:saddr:TEREDO    [RFC4380] [RFC8190]
0          0          drop     all    ==     Eth0/2          *               2001:1::1
*                     Comment:     ipv6:saddr:Port Control Protocol Anycast
[RFC7723]
0          0          drop     all    ==     Eth0/2          *               2001:1::2
*                     Comment:     ipv6:saddr:Traversal Using Relays around NAT
Anycast    [RFC8155]
0          0          drop     all    ==     Eth0/2          *
2001:2::/48           *                    Comment:     ipv6:saddr:Benchmarking
[RFC5180][RFC Errata 1752]
0          0          drop     all    ==     Eth0/2          *
2001:3::/32           *                    Comment:     ipv6:saddr:AMT    [RFC7450]
0          0          drop     all    ==     Eth0/2          *
2001:4:112::/48       *                    Comment:     ipv6:saddr:AS112-v6
[RFC7535]
0          0          drop     all    ==     Eth0/2          *
2001:10::/28          *                    Comment:     ipv6:saddr:Deprecated
(previously ORCHID)    [RFC4843]
0          0          drop     all    ==     Eth0/2          *
2001:20::/28          *                    Comment:     ipv6:saddr:ORCHIDv2
[RFC7343]
```

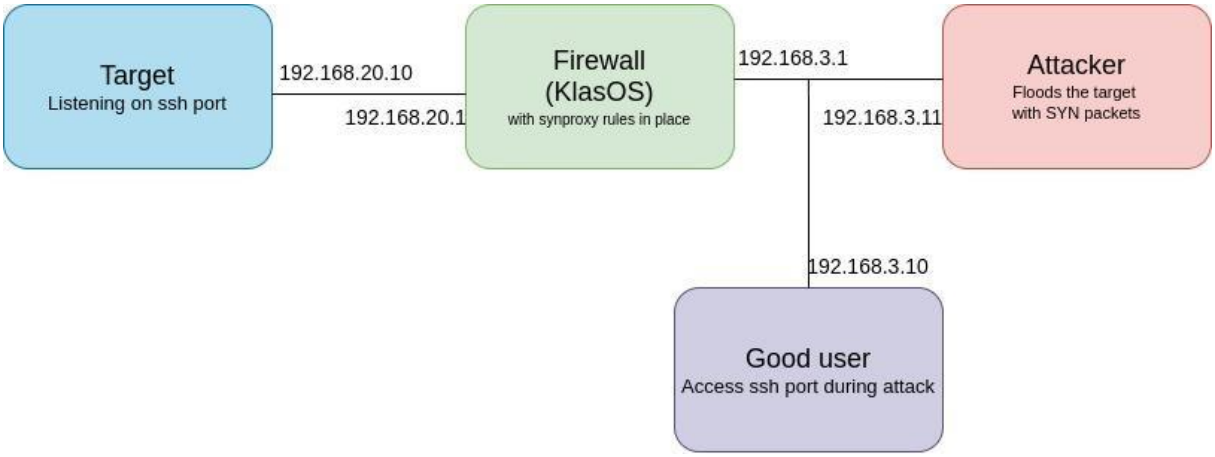
```
0          0          drop     all    ==     Eth0/2          *
2001:db8::/32        *                    Comment:     ipv6:saddr:Documentation
[RFC3849]
0          0          drop     all    ==     Eth0/2          *               2002::/16
*                    Comment:     ipv6:saddr:6to4    [RFC3056
0          0          drop     all    ==     Eth0/2          *
2620:4f:8000::/48    *                    Comment:     ipv6:saddr:Direct Delegation
AS112 Service    [RFC7534]
0          0          drop     all    ==     Eth0/2          *               fc00::/7
*                    Comment:     ipv6:saddr:Unique-Local    [RFC4193] [RFC8190]
0          0          drop     all    ==     Eth0/2          *               fe80::/10
*                    Comment:     ipv6:saddr:Link-Local Unicast    [RFC4291]
0          0          drop     all    ==     Eth0/2          *               ff00::/8
*                    Comment:     ipv6:saddr:Multicast addresses
KlasOS(config-if)# ip security verify reverse-path
KlasOS(config-if)# do show ip security system interface Ethernet 0/2
pkts       bytes      target   prot   opt    in              out             Source
Destination
0          0          drop     all    ==     Eth0/2          *               0.0.0.0/8
*                    Comment:     ipv4:saddr:'This network'  [RFC791], Section 3.2
0          0          drop     all    ==     Eth0/2          *               0.0.0.0
*                    Comment:     ipv4:saddr:'This host on this network' [RFC1122],
Section 3.2.1.3
0          0          drop     all    ==     Eth0/2          *
10.0.0.0/8           *                    Comment:     ipv4:saddr:Private-Use
[RFC1918]
0          0          drop     all    ==     Eth0/2          *
100.64.0.0/10        *                    Comment:     ipv4:saddr:Shared Address
Space    [RFC6598]
0          0          drop     all    ==     Eth0/2          *
127.0.0.0/8          *                    Comment:     ipv4:saddr:Loopback
[RFC1122], Section 3.2.1.3
0          0          drop     all    ==     Eth0/2          *
169.254.0.0/16       *                    Comment:     ipv4:saddr:Link Local
[RFC3927]
0          0          drop     all    ==     Eth0/2          *
172.16.0.0/12        *                    Comment:     ipv4:saddr:Private-Use
[RFC1918]
0          0          drop     all    ==     Eth0/2          *
192.0.0.0/24         *                    Comment:     ipv4:saddr:IETF Protocol
Assignments   [RFC6890], Section 2.1
0          0          drop     all    ==     Eth0/2          *
192.0.0.0/29         *                    Comment:     ipv4:saddr:IPv4 Service
Continuity Prefix  [RFC7335]
0          0          drop     all    ==     Eth0/2          *               192.0.0.8
*                    Comment:     ipv4:saddr:IPv4 dummy address  [RFC7600]
0          0          drop     all    ==     Eth0/2          *               192.0.0.9
*                    Comment:     ipv4:saddr:Port Control Protocol Anycast
[RFC7723]
0          0          drop     all    ==     Eth0/2          *
192.0.0.10           *                    Comment:     ipv4:saddr:Traversal Using
Relays around NAT Anycast   [RFC8155]
```

```
0          0          drop     all    ==     Eth0/2          *
192.0.0.170          *                     Comment:     ipv4:saddr:NAT64/DNS64
Discovery   [RFC8880][RFC7050], Section 2.2
0          0          drop     all    ==     Eth0/2          *
192.0.0.171          *                     Comment:     ipv4:saddr:NAT64/DNS64
Discovery   [RFC8880][RFC7050], Section 2.2
0          0          drop     all    ==     Eth0/2          *
192.0.2.0/24         *                     Comment:     ipv4:saddr:Documentation
(TEST-NET-1)  [RFC5737]
0          0          drop     all    ==     Eth0/2          *
192.31.196.0/24      *                     Comment:     ipv4:saddr:AS112-v4
[RFC7535]   2014-12
0          0          drop     all    ==     Eth0/2          *
192.52.193.0/24      *                     Comment:     ipv4:saddr:AMT [RFC7450]
2014-12
0          0          drop     all    ==     Eth0/2          *
192.88.99.0/24       *                     Comment:     ipv4:saddr:Deprecated (6to4
Relay Anycast) [RFC7526]
0          0          drop     all    ==     Eth0/2          *
192.168.0.0/16       *                     Comment:     ipv4:saddr:Private-Use
[RFC1918]
0          0          drop     all    ==     Eth0/2          *
192.175.48.0/24      *                     Comment:     ipv4:saddr:Direct Delegation
AS112 Service [RFC7534]
0          0          drop     all    ==     Eth0/2          *
198.18.0.0/15        *                     Comment:     ipv4:saddr:Benchmarking
[RFC2544]
0          0          drop     all    ==     Eth0/2          *
198.51.100.0/24      *                     Comment:     ipv4:saddr:Documentation
(TEST-NET-2)  [RFC5737]
0          0          drop     all    ==     Eth0/2          *
203.0.113.0/24       *                     Comment:     ipv4:saddr:Documentation
(TEST-NET-3)  [RFC5737]
0          0          drop     all    ==     Eth0/2          *
240.0.0.0/4          *                     Comment:     ipv4:saddr:Reserved
[RFC1112], Section 4
0          0          drop     all    ==     Eth0/2          *
255.255.255.255      *                     Comment:     ipv4:saddr:Limited Broadcast
[RFC8190] [RFC919], Section 7
0          0          drop     all    ==     Eth0/2          *
224.0.0.0/4          *                     Comment:     ipv4:saddr:Multicast addresses
0          0          drop     all    ==     Eth0/2          *               *
*                     Comment:     Eth0/2:ipv4: verify reverse path route
```

16. Mitigating TCP flood attacks using SYNPROXY feature

Synproxy intercepts new TCP connections and handles the initial 3-way handshake using syncookies instead of conntrack to establish the connection. Running synproxy on a listening server port thus prevents a SYN flood attack on that port from consuming limited conntrack resources. With conntrack, false SYN-ACK and ACK packets can be filtered out before they hit the "listen" state lock.

The diagram below shows the network of a possible attack. The target is listening on port 22 (ssh). The attacker is attempting to flood the target with SYN packets The firewall has synproxy rules configured that protect port 22 which enables the 'good user' to establish an ssh session with the target.



The following example configuration can be used to protect the ssh port (port 22)

```
access-list 100 permit tcp any any eq 22 advanced raw ct notrack
access-list 100 permit tcp any any eq 22 conn INVALID,UNTRACKED advanced forward-only
synproxy wscale 7 mss 1460

ip tcp conntrack strict

interface vSwitch 3
ip access-group 100 in
```

The first section creates an extended access list 100 that consists of 2 rules.

First rule:

```
access-list 100 permit tcp any any eq 22 advanced raw ct notrack
```

This rule ensures connections that need protection don't create new conntrack entries for SYN packets.

WARNING: Other firewall rules which rely on conntrack will not work if this setting is in place.
Note this rule is added to the raw table. In all cases, TCP half open connections are monitored per TCP client. The default state is to allow 2048 half open TCP connections if nothing is configured. After this number is reached, new SYN packets are dropped. Each new SYN attempt is monitored for 31 seconds, based on retrying 5 SYN-ACK responses.

The TCP SYN backlog, counting the number of half open sessions, maybe modified as follows:

KlasOS(config)# ip tcp max-syn-backlog

<1-2099999999> SYN backlog max (1..2099999999)

Second rule:

```
access-list 100 permit tcp any any eq 22 conn INVALID,UNTRACKED advanced forward-
only synproxy wscale 7 mss 1460
```

This line actually creates two rules on the system:

- The first rule catches UNTRACKED SYN and INVALID packets that contain the ACK and directs them to the SYNPROXY target module.
- The second rule catches the INVALID state packets that fell-through the SYNPROXY module and drops them. Basically, this will drop SYN-ACK based floods.

These rules are added to the filter table. Note that 'advanced forward-only' is specified here, so the rules will only be applied to the FORWARD chain.
If 'input-only' is selected, the rule will protect the system that the firewall is running on and no other devices.
'forward-only' and 'input-only' are the only two 'advanced' options can be used with the synproxy feature.
'raw' and 'output-only' cannot be used with synproxy.
The source and destination addresses are specified as 'any' but a unique host/network can be specified here as with any extended access list.

Synproxy parameters:

Parameter	Description	Range
Mss	maximum segment size	500-1460
Wscale	window scale The TCP window scale is an option used to increase the maximum window size from 65,535 bytes by a factor of n (wscale) The window scale option is used only during the TCP three-way handshake. The window scale value represents the number of bits to left-shift the 8-bit window size field. The window scale value can be set from 0 (no shift) to 7. This value must match that of the backend server. On a linux machine it can be found with cat /proc/sys/net/ipv4/tcp_window_scaling To modify the value use 'echo' For example, to change the window scale to 4 echo 4 > /proc/sys/net/ipv4/tcp_window_scaling	0-7

The second part of the configuration enables the tcp conntrack strict setting:

```
ip tcp conntrack strict
```

This setting will make the connection tracking system more strict in this categorization.
This is necessary to have ACK packets marked as INVALID state. Specifically this will (among other things) help catch ACK-floods.

The last part of the configuration simply applies the access list to an interface. In this case vSwitch3.
Note that synproxy rules may only be added to interfaces and not to other services such as webserver or vty.
Synproxy rules may only be added with direction 'in'. Unless 'advanced forward-only' or 'advanced input-only' is specified in the rule, the rule will be applied to both INPUT and FORWARD chains.

```
interface vSwitch 3
ip access-group 100 in
```

16.1. How to count dropped SYN packets and SYN packets that have not been dropped

`show ip firewall system` can be used to view details of the rules and packet counters.

Note: the below table shows an acl applied to vSwitch 3 that has an explicit 'deny any any' rule (lines 4 and 11)

```
KlasOS# show ip firewall system
TABLE: filter, CHAIN: INPUT  IPv4
pkts       bytes      target   prot   opt    in             out            Source
Destination
861        71063      ACCEPT   all    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0
0          0          DROP     all    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0

TABLE: filter, CHAIN: FORWARD  IPv4
pkts       bytes      target   prot   opt    in             out            Source
Destination
7002M      700G       SYNPROXY tcp    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0            Extra opts:  tcp, dpt:22, ctstate, INVALID,UNTRACKED
3501M      140G       DROP     all    --     *              *              0.0.0.0/0
0.0.0.0/0            Extra opts:  ctstate, INVALID,UNTRACKED
54269      2377K      ACCEPT   all    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0
0          0          DROP     all    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0

TABLE: raw, CHAIN: PREROUTING  IPv4
pkts       bytes      target   prot   opt    in             out            Source
Destination
3501M      560G       CT       tcp    --     vSwitch3       *              0.0.0.0/0
0.0.0.0/0            Extra opts:  tcp, dpt:22, flags:0x17/0x02
```

16.1.1. To count SYN packets that have not been dropped

Line 9

This rule catches UNTRACKED SYN and INVALID packets that contain the ACK and directs them to the SYNPROXY target module. The number of packets not dropped and directed to the synrpoxy module can be seen in the first column. In this example, 7002M.

16.1.2. To count dropped SYN packets

Line 10

This rule catches the INVALID state packets that fell-through the SYNPROXY module and drops them. The number of packets dropped can be seen in the first column. In this example, 3501M.

17. Configuring NTP

17.1. Configuration Commands

17.1.1. NTP Client

Configure an NTP client in KlasOS with the following command in CONFIGURATION MODE

- `KlasOS(conf)# ntp server <IP address>`
  - IP address is the IP address of the NTP server

Note: To configure multiple NTP servers, this command must be entered for every NTP server the administrator wants to sync to. The only version of NTP supported by the TOE is NTPv3.

17.1.1.1. Client Authentication Key

If the NTP server supports cryptographic authentication using SHA-1, configure the correct key in KlasOS by appending a [key] option at the end of the ntp server command. In CONFIGURATION MODE:

```
KlasOS(conf)# ntp authenticate
KlasOS(conf)# ntp authentication-key 1 sha1 <shared secret>
KlasOS(conf)# ntp trusted-key 1
KlasOS(conf)# ntp server <IP address> key 1
```

17.1.2. NTP Server

Configure an NTP server in KlasOS with the following command in CONFIGURATION MODE:

```
KlasOS(conf)# ntp master
```

17.1.2.1. NTP Server Stratum

To configure the stratum level of the NTP server in KlasOS, append a stratum number after the ntp master configuration command:

```
KlasOS(conf)# ntp master 2
```

In NTP, the stratum number represents how many hops away the device is from reliable timing source, such as a device deriving timing from GPS or a Caesium beam.

17.1.3. Source Interface

NTP traffic sent from KlasOS can be configured to use a specific IP address as the source address. In CONFIGURATION MODE:

```
KlasOS(conf)# ntp source <X.X.X.X>
```

Where X.X.X.X is the source IP address in dotted decimal format.

17.2. Operational Commands

17.2.1. Show Status

To see the status of NTP, in PRIVILEGED EXEC Mode:

```
KlasOS# show ntp status
```

An example of the output:

```
KlasOS# show ntp status
Clock is synchronized, stratum 4, reference is 192.168.1.9
reference time is e1e56158.ad8994bf Wed, Feb 5 2020 15:44:24.677
clock offset is 0.1180 msec, root delay is 0.7000 msec
```

17.3. NTP Behavior

Once configured, synchronization can take up to 5 minutes.
The date and time on the NTP server CANNOT be set in the past before the build date of the KlasOS firmware image. The build date can be seen with the show version command:

```
KlasOS# show version
KlasOS keel v5.4.0
Built Tue Nov 26 18:06:13 GMT 2019
System uptime is 21 minutes
Configuration register is 0x2001
```

Note: The TOE automatically denies any NTP timestamp updates from a multicast or broadcast IP address.