Device Zeroization and Recovery

This chapter contains the following sections:

- Device Zeroization, on page 1
- Push Button, on page 2
- Important Notice about Zeroization, on page 2
- Zeroization Details, on page 3
- Command Line Interface, on page 4
- Zeroization Trigger, on page 4
- To Trigger Zeroization, on page 4
- Emergency Recovery Installation, on page 5

Device Zeroization

Zeroization consists of erasing any and all potentially sensitive information in the switch securely and followed by sanitize operation. This includes erasure of Main memory, license, logs, cache memories, IOS-XE packages, system configs, and other memories containing packet data, NVRAM, and Flash memory.

The process of zeroization is launched upon the initiation of a user command and a subsequent trigger.

Note Ensure that you are familiar with the Emergency Recovery Installation, on page 5 procedure **BEFORE** attempting to test the Zeroize feature.

On the ESS9300, the Push Button is used exclusively for triggering the Zeroization process. This process will zeroize and erase switch configuration files, or the entire flash file system, depending on the option provided under **service declassify**.

The Zeroization process starts as soon as the Push Button is pressed. The CLI command, **service declassify**, is used to set the desired action in response to the Push Button press. To prevent accidental erasure of the system configuration/image, the default setting is set to **no service declassify**.

Caution Zeroization does NOT erase removable media such as SD Card and USB Storage. This media must be removed from the system and erased or destroyed using procedures that are outside the scope of this document.

Push Button

There is no actual button on the ESS9300, and the system integrator must configure their platform with a Push Button. Reset on an ESS9300 does not cause the device to reboot, but initiates the configured level of zeroization.

Zeroization can be triggered by the Push Button, or software-triggered by a privilege 15 user with console access. There is no remote access for security reasons.

The Zeroization process starts as soon as the Push Button is pressed.

Important Notice about Zeroization

eMMC is a managed NAND. This means that the embedded switch system does not interact with the flash memory directly. The flash controller presents a block-style interface to the system, and it handles the flash management (analogous to the Flash Translation Layer). The embedded switch cannot access the raw flash directly.

The JEDEC standard has commands that are supposed to remove data from the raw flash. In Cisco’s implementation, the “Erase” and “Sanitize” commands are used. The eMMC standard JESD84-B51 defines “Sanitize” as follows:

“The Sanitize operation is a feature ... that is used to remove data from the device according to Secure Removal Type. The use of the Sanitize operation requires the device to physically remove data from the unmapped user address space”

After the sanitize operation is completed, no data *should exist* in the unmapped host address space.

⚠ **Caution** Zeroize does a very thorough wipe of all non-protected parts of the eMMC flash using the best technology designed by the flash manufacturer today and can do so using the push of a button without the need for a console, ssh, or management session of any kind. It is the integrator's and end user's responsibility to determine the suitability regardless of the CLI keyword used to enable the feature.

⚠ **Caution** Note: While Cisco IOS and Cisco IOS-XE use the command line text of “declassify” in the command line interface (CLI) to enable the zeroize feature, in no way does this represent any specific endorsement or acknowledgment of a Government approved flash erasure methodology.

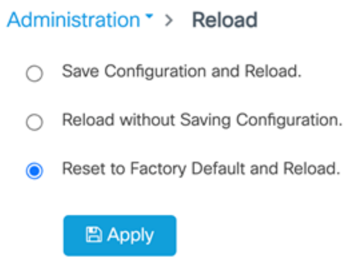
⚠ **Caution** Declassification procedures are unique to each Government organization. Cisco solely provides the technical detail of the erasure operation here, not the policy distinction or any specific recommendation per classification.

⚠ **Caution** Please refer to your respective Government Agency policies, procedures, and recommendations for the handling of sensitive data to see if this procedure meets with those requirements.

WARNING!

The CLI **service declassify erase-all** is literally a **software self-destruct mechanism** intended for defense and intelligence environments that attempts to wipe clean, all of the writable non-volatile storage on the device to clear the device configuration, other stored configurations and all security credentials including any additional license keys.

Please do not use this feature in lieu of doing a **write erase** from the CLI or from the Administration page, Reload option of the WebUI. Invoke the reload with the **Reset to Factory Default and Reload** option and click **Apply**. See the following figure.



Administration > Reload

○ Save Configuration and Reload.

○ Reload without Saving Configuration.

● Reset to Factory Default and Reload.

Apply

If **service declassify erase-all** is invoked, after restoring the IOS-XE image and device configuration, you must re-license the device using the standard Cisco Smart Licensing procedures which ultimately require a Cisco Smart Account and access to the internet or a satellite license server.

Zeroization Details

When zeroization is triggered from the Push Button, the following occurs:

- wipe persistent storage devices
- set flag to wipe RAM for bootloader, reload
- bootloader checks flags, wipes RAM

Tasks performed by IOS-XE

1. Shutdown interfaces and flash the zeroization LED.
2. Clear data path related memory from ASIC.
3. Set rommon variable for bootloader to trigger the RAM erasure.
4. Calls system_reload API to reload the device.

Tasks Performed By Bootloader

After system_reload is triggered from IOS-XE, control transfers to Bootloader. When Bootloader sees the zeroization triggers, it performs a secure erase and sanitization of all the unlocked eMMC partitions through secure erase opcodes & sanitize opcodes. The erased and sanitized areas are:

- Crash info: This has the system crash info file.
- ROMMON Variables: System vars, including user defined vars are erased.
- License & License Backup: System license files are stored here.
- OBFL: IOS-XE OBFL failure logs are stored here.

- Optional keys: Some optional keys are installed here, which are sanitized.
- Flash: is the partition where all the system configuration files, systems data and other user data are stored.

Note If a power cycle happens during zeroization, the bootloader would start zeroization over again since the rommon variable for zeroization is still present.

The following message appears on the console when reset has been triggered:

```
System Bootstrap, Version 1.4(DEV) [vandvisw-vandvisw 113], DEVELOPMENT SOFTWARE
Copyright (c) 1994-2019 by cisco Systems, Inc.
Compiled at Mon Jun 3 10:56:19 2019 by vandvisw
ESS-9300-CON-K9 platform with 4194304 Kbytes of main memory
MCU Version - Bootloader: 8, App: 10
MCU is in application mode.
Reset button push detected
```

Command Line Interface

There are two levels of zeroization actions, erase-nvram and erase-all. The following CLI shows the options:

```
switch(config)#service declassify ?
erase-nvram    Enable erasure of switch configuration as zeroization action. Default is no
 erasure.
erase-all      Enable erasure of both flash and nvram file systems as part of zeroization.
 Default is no erasure
```

Zeroization Trigger

Zeroization can be triggered by either software or by the push button. In either case, there are a series of commands that need to be entered.

```
switch#config terminal
switch(config)#service declassify {erase-nvram | erase-all}
```

To confirm if the feature is enabled:

```
switch#show declassify

Declassify facility: Enabled=Yes  In Progress=No
                     Erase flash=Yes  Erase nvram=Yes
  Declassify Console and Aux Ports
  Shutdown Interfaces
  Reload system
```

To remove the feature, use the following command:

```
switch(config)#no service declassify
```

To Trigger Zeroization

To trigger the zeroization from the command line:

```
switch#declassify trigger
```

To trigger the zeroization from the push button, press and hold the button for 4+ seconds. When the system auto reloads, it will come up in ROMMON mode: "$$" with bootflash: wiped clean.

Emergency Recovery Installation

The following procedure supports the Cisco ESS3300 and the Cisco ESS9300.

Note There is different terminology used when referring to the reset button depending on the product. The IE3x00 switches call this the Express Setup switch. Other products may refer to this as the Factory Default Switch. In either case, the functionality is the same.

If the other recovery methods fail, the switch has a trap door method that you can use in order to recover the system. You must have a terminal that is connected to port Gi1/3 of the switch that runs a TFTP server. Download a valid image file from CCO and store it in the root of the TFTP server.

It is likely that the switch is stuck at the **switch**: prompt. However, if you are in a boot loop, you can use the Express Setup switch on the front of the switch in order to break the cycle: hold the button for approximately <TBD> seconds, and the switch breaks the cycle and stops at the **switch**: prompt.

Complete these steps in order to perform an emergency recovery:

Step 1: Boot the emergency install image.

```
switch: switch: boot emgy0:<image-name>.SPA.bin
Booting golden bootloader...
Initializing disk drivers...
Initializing file systems...
************************************************************
* Rom Monitor for ESS3300                                  *
* Copyright (c) 2017-2018 by Cisco Systems, Inc.           *
* All rights reserved.                                     *
************************************************************
* Version:  1.1.1
* Compiled: Sun 01-Jul-18 22:17 [RELEASE SOFTWARE]
* Boot Partition: qspi-golden-bootloader
* Reset Reason: Soft Reset
Loading "emgy0:ess3x00-universalk9.17.04.01.SPA.bin" to memory...
Verifying image "emgy0:ess3x00-universalk9.17.04.01.SPA.bin"...
Image passed digital signature verification
Checking for Bootloader upgrade...
Bootloader upgrade not required
SUP PL (profile: 1) configuration done successfully
<...>
Press RETURN to get started!
Switch>
```

Step 2: Configure an IP address on the switch. Additional details on IP configuration can be found here

```
switch(config-if)# ip address <ip-address> <subnet-mask>
```

Step 3: Ping the terminal that contains the TFTP server in order to test the connectivity:

```
switch> ping 192.168.2.1
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 192.168.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Step 4: Copy the image via tftp

```
switch> copy tftp: //location/directory/<bundle_name> flash:
<...>
```

Step 5: Restart the system.