Device Zeroization

This section contains the following topics:

Push Button

There is no actual button on the ESR6300, and the system integrator must configure their platform with a Push Button. Reset on an ESR6300 does not cause the device to reboot, but initiates the configured level of zeroization.

- When the system is running in IOS mode, pressing this push button for 4+ seconds will cause files erase in flash, and will reset to factory-default mode on boot up.

  The button must be pressed while the system is turned on at the same time.

  The push button must continue to be held for more than 4 seconds after the power is turned on.

  Config-reg setting is in NVRAM, and not changed by the button push.

Pressing the push button when in rommon mode has no effect.

Pressing the push button when in IOS mode causes a syslog message to appear and triggers a reload.

Pressing the push button for more than 4+ seconds after power up displays the following message 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
ESR-6300-CON-K9 platform with 4194304 Kbytes of main memory
MCU Version - Bootloader: 8, App: 10
MCU is in application mode.
Reset button push detected
```

How the router boots up depends on the following conditions:

- If there is a golden image in flash, the router will bootup with this image by default.
- If a golden image is not found, the router will be in rommon> mode as expected. Or, based on config-reg entry, may boot up first image in the flash:

Microcontroller Unit (MCU)

The MCU is part of the ESR6300 hardware. It performs the following functions:

- Monitors the Push button status at power up
- Monitors the system hardware watchdog output
- Maintains Reset Reason register
- Controls the SYS LED

The MCU versions are displayed using **show version**. Details on MCU version and upgrade status are also stored in Flash: as boothelper.log. The MCU is automatically upgraded by the software.

```
Router#show ver | i MCU
MCU bootloader version: 8
MCU application version: 10

Router#cat flash:boothelper.log
Logging at Fri Nov 15 05:00:54 Universal 2019
boot loader upgrade enabled
Bootloader is up-to-date
Current MCU App version is 10
MCU firmware is up-to-date
```

In the event the MCU Application is corrupt, or does not match the Release Notes version, this has to be repaired. Steps to recover from this state: Reload router, hit Ctrl+C to break into rommon mode.

```
Rommon>set MCU_UPGRADE=IGNORE - Ignore MCU firmware upgrade errors.
Rommon>sync
Rommon>reset
Rommon>boot bootflash:<image>
```

Once the MCU successfully upgrades, you can disable/unset this IGNORE option in rommon. Details on other MCU setting rommon options follow: (there are no available IOS configuration options or linux shell mode troubleshooting measures)

```
set MCU_UPGRADE=SKIP – Prevents MCU firmware upgrade from taking place.
set MCU_UPGRADE=FORCE – Forces MCU firmware upgrade to take place.
unset MCU_UPGRADE – Normal operation. Allows automatic upgrade
```

Device Zeroization

Zeroization consists of erasing any and all potentially sensitive information in the device.

This includes erasure of Main memory, cache memories, 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.

By default, the router will have the zeroization feature disabled. SPI: Flash, I2C, and ACT2 are not impacted by this feature.

When zeroization is functionally active, SYS LED indicates blinking yellow until the router reloads.

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. On triggering zeroization, the eMMC, NVRAM will be erased completely.

The zeroization process starts as soon as the push button is pressed down or the command is triggered.. The CLI command, "service declassify", is used to set the desired action in response to push button press. To prevent accidental erasure of the system configuration/image, the default setting is set to "no service declassify".

Note For complete details on how the process of zeroization works, see the Cisco Embedded Service 6300 Series Router Hardware Technical Guide.

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.

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

In the above example, erase-nvram will wipe only nvram, while erase-all will wipe bootflash: and nvram as well.

To confirm if the feature is enabled:

```
Router#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:

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

To Trigger Zeroization

To trigger the zeroization from the command line:

```
Router#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.

Important Notice about Zeroization

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

Warning 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.

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.

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.**

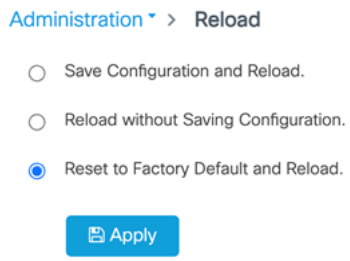
Declassification procedures are unique to each Government organization. Cisco solely provides the technical detail of the erasure operation here, not the policy distinction or any specific recommendation per classification.

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.



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.

Command Line Interface

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

```
device(config)#service declassify ?
erase-nvram    Enable erasure of device 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
```

The perma-locked bootable image(s) in the flash file system will still be available and can be used for booting the device.

The “erase-all” level of zeroization process erases the entire flash file system. This also wipes out all files and perma-locked bootable image(s). All interfaces are shut down before this process. Here, erasure of individual files in the flash file system is not possible and the only option is to erase the entire flash file system. This also erases packet data, ASIC data and processors related caches along with scrubbing Main memory.

With any level of zeroization, the device always fall back to the ROMMON prompt on the console after the erasure of configuration files or flash file system.

Zeroization Support in bootloader

The zeroization process may take several minutes, depending on several system parameters such as the size of DDR memory, EMMC disk size, etc.

It is possible that the zeroization may get interrupted by a power cycle before it completes. Since the primary OS image on EMMC itself gets purged during zeroization, it becomes impossible to continue zeroization after a power cycle. To solve this, zeroization support has been in the bootloader and will run it to completion even if it gets interrupted by power cycles.

Troubleshooting Zeroization

The following table shows troubleshooting tips for zeroization:

Table 1:

Symptoms	Troubleshooting
Upon zeroization, some files are still left behind in the bootflash:	In that case, collect logs to pass on to Cisco support/engineering teams for further debugging. Upon zeroization, the ideal state of the router should be in rommon mode displayed as $. When executing $dir bootflash:, the partition should not even be mounted.
How do you know if zeroization was triggered if the router is in $$ prompt?	Symptoms are bootflash: unmounted, router should be stuck in rommon and environment variables [rommon>set] should be wiped clean. There is no syslogging or reset reason displayed.

Symptoms	Troubleshooting
Zeroization does not take effect on trigger.	Check MCU version: **show ver | I MCU**. MCU application version should reflect the latest as per the Release Notes. If it is anything else, likely Application is corrupt and we need to re-trigger MCU upgrade. Capture the output from **show platform declassify**. Ensure you have actually enabled zeroization, before executing **declassify trigger**. Ensure you have privilege 15 to trigger zeroization. Ensure you are on console access. SSH/telnet remote session zeroization trigger is considered a security issue and is therefore disabled.