New Features

New Features for Cisco IOS XE 17.14.1a

New features in this release are listed below:

- SNMP MIB Support for DLEP

New Features for Cisco IOS XE 17.13.1

This chapter contains the following sections:

- IPv4 and IPv6 Multicast Over DLEP
- IPv6 Control Plane for DLEP
- SD-WAN Remote Access (SD-WAN RA)
- Change in CLI Output for the FN980 5G Modem

New Features for Cisco IOS XE 17.12.1a

This chapter contains the following sections:

- DLEP IPv6 Unicast
- Uncapped License Implementation

New Features for Cisco IOS XE 17.11.1a

This chapter contains the following sections:

- Change to Smart Licensing Packaging
- Galileo Support on the LTE Pluggable Modules

New Features for Cisco IOS XE 17.10.1a

This chapter contains the following sections:

- Support for the P-5GS6-GL Pluggable Module on the ESR6300
- MAB 802.1x Support
- Enable Secure Data Wipe Capabilities
- Rawsocket Keepalive Configuration CLI

New Features for Cisco IOS XE 17.9.1

This chapter contains the following sections:

- Install Mode Support
- Cellular Boot Time Improvements
- IOS XE Downgrade Warning
- SNMP Polling of Temperature OID
- GPS Mode Enabled By Default
- Cisco WebUI Access Point Name (APN)
- IPv6 Multicast over PPPoE

New Features for Cisco IOS-XE 17.8.1

Cellular Serviceability Enhancements

Enhancements have been made for cellular and GPS features as follows:

Trigger points and debug code can be enabled via controller cellular CLIs for generating and trap the debug data automatically without manual intervention. The following CLI options are available:

```
(config-controller)#lte modem serviceability ?
  gps               GPS debugging
  interface-resets  Interface resets/Bearer deletion
  modem-crash       Modem-crash debugging
  modem-resets      IOS initiated unknown modem-resets
```

The debug data includes the following:

- Context Based debug logs (tracebacks, and GPS locations).
- Well formatted debug messages.
- Vendor specific debug data at a broader range.

The debug logs are located in the following location of flash:

```
router#dir flash:servelogs
Directory of bootflash:/servelogs/

259340  -rw-              122   Sep 7 2021 17:40:44 +00:00  gpslog-slot5-20210907-174044
259339  -rw-             1734   Sep 7 2021 12:14:07 +00:00  celllog-slot5-20210905-164628
```

GPS and cellular log files are created separately with file names using the timestamp at the time of the creation. These files are created as follows:

- If the existing file has reached 10Mb, a new file will be created.
- A new file will be created if the feature (GPS, or cellular) is completely disabled, and then re-enabled.

GNMI Broker (GNMIB) Update

The GNMI Broker (GNMIB) has been extended to support the gRPC Network Operations Interface (gNOI) reset.proto service. This service provides functionality for restoring the device to its factory defaults via gRPC.

When the service is executed, it behaves similarly to the ‘factory-reset all’ command, and subsequently triggering a reload. Additionally, the service will maintain the current booted image. The additional steps below will be taken to comply with the reset.proto service:

- Set the rommon BOOT variable to the current booted image and maintain it through reload following factory-reset
- Enable autoboot to bring the device up on the current booted image following factory-reset.

gRPC Network Operations Interface Update

gNOI is the gRPC Network Operations Interface. gNOI defines a set of gRPC-based microservices for executing operational commands and procedure on network devices, such as OS Install, Activate, and Verification.

Through gNOI os.proto will be possible to perform operating system related tasks such as OS activation, install, detailed overview, internal OS commands, and finally to output a summary of OS operations.

Furthermore, gNOI os.proto can also be used to display the gnmib detailed state, check the gnmib operational statistics, and also to output modifiers.

Raw Socket Feature Enhancement

This enhancement allows the user to input the maximum number of retries available to the write socket. The range of the number of retries goes from 1 to 1000. The default number of retries is 10. To accommodate this feature, a new CLI has been created, **raw-socket tcp max-retries <1-1000>**. <1-1000> is the maximum number of retries.

SCADA Enhancement for TNB

This enhancement provides compatibility with TNB's WG RTUs, including the following:

- TNB RTUs require Reset-Link message to be sent out along with Link-Status message to ensure correct initialization of the serial. The feature can be selectively turned on using the new configuration CLI **scada-gw protocol force reset-link.**
- When clock passthru is enabled and if the router hasn't received the timestamp from the DNP3-IP master, the router's hardware time will be sent downstream to RTU. Upon receiving a new timestamp from DNP3-IP master, the router will start sending the new timestamp sourced from DNP3-IP master to RTU.
- The number of bufferable DNP3 events in memory will be increased from 600 to 10000.
- The **scada-gw protocol interlock** command will be supported for DNP3. Previously, the support only existed for T101/T104. With this new enhancement, the router will disconnect Serial link if the DNP3-IP master is down or unreachable. Similarly, when the Serial link to RTU is down, the TCP connection to DNP3-IP master will be untethered.
- Custom "requests" will be automatically ordered based on priority so that the user can specify them in any order that they would like to.

DLEP and Credit Based Radio Aware Routing Support

DLEP Support

DLEP addresses the challenges faced when merging IP routing and radio frequency (RF) communications. Cisco provides capabilities that enable:

- Optimal route selection based on feedback from radios
- Faster convergence when nodes join and leave the network
- Efficient integration of point-to-point, point-to-multipoint and broadcast multi-access radio topologies with multi-hop routing

- Flow-controlled communications between the radio and its partner router using rate-based Quality of Service (QoS) policies
- Dynamic shaping of fluctuating RF bandwidth in near real time to provide optimized use of actual RF bandwidth

Credit Based Radio Aware Routing Support

Radio-Aware Routing (RAR) is a mechanism that uses radios to interact with the routing protocol OSPFv3 to signal the appearance, disappearance, and link conditions of one-hop routing neighbors.

In a large mobile networks, connections to the routing neighbors are often interrupted due to distance and radio obstructions. When these signals do not reach the routing protocols, protocol timers are used to update the status of a neighbor. Routing protocols have lengthy timer, which is not recommended in mobile networks.

PPPoE extensions are used when the router communicates with the radio. In the Cisco IOS implementation of PPPoE, each individual session is represented by virtual access interface (connectivity to a radio neighbor) on which, QoS can be applied with these PPPoE extensions.

RFC5578 provides extensions to PPPoE to support credit-based flow control and session-based real time link metrics, which are very useful for connections with variable bandwidth and limited buffering capabilities (such as radio links).

Complete details can be found in Radio Aware Routing and Dynamic Link Exchange Protocol

New Features for Cisco IOS-XE 17.7.1

LTE Support for ESR6300

Cisco IOS-XE Release 17.7.1 adds support for LTE modules on the ESR6300 platform in slot 3. The related interface is Cellular 0/3/0.

ESR6300 will only support the following LTE modules:

- P-LTE-MNA (WP7610 modem)
- P-LTEA-LA (EM7430 modem)
- P-LTEA-EA (EM7455 modem)
- P-LTEAP18-GL (LM960 modem)

The cellular interface/module configuration will follow the same as on any other Polaris IOS-XE based platform and LTE modem.

Support 1G SFPs

Release 17.7.1 will add support for the following SFPs:

GLC-T-RGD

CWDM-SFP-1470=

CWDM-SFP-1610=

CWDM-SFP-1530=

DWDM-SFP-3033=

DWDM-SFP-3112=

GLC-BX-D-I=

GLC-BX-U-I=

GLC-TE

New Features for Cisco IOS-XE 17.6.1

Additional SFF Support

A Small Form Factor (SFF) transceiver is an optical transceiver that is soldered onto the PCB. It tends to be smaller in size than traditional SFPs.

There was a need for the ESR6300 to support SFFs in order to meet high vibration and shock requirements that can't be met with traditional SFPs. Therefore, support was added in 17.6.1 for the Finisar FTE8501K1LTN.

Since, this is not a Cisco transceiver, the end-user will need to use the **service unsupported-transceiver** CLI in order to get the router to accept and initialize the SFF.

Functionally, an SFF operates in the same way as an SFP.

VXLAN support

This feature functions the same as it does on the IR1101 running 17.5.1

See the section on VXLAN in the IR1101 Software Configuration Guide.

New Features for Cisco IOS-XE 17.5.1

RFC4884 ICMPv6 and MPLSv6

RFC 4884 redefines selected ICMP error messages to support multi-part operation.

A multi-part ICMP message carries all of the information that ICMP messages carried previously, as well as additional information that applications may require.

RFC 4884 feature introduces an 8-bit length attribute to the following ICMPv6 messages with extensions.

- Destination Unreachable (type = 1)
- Time Exceeded (type = 3)

As part of RFC 4884 feature, for applications like MPLS/trace route which add extensions to type 1 and type 3 ICMPv6 error messages, original datagram length will be added in ICMPv6 header.

Also, infra is added as part of RFC 4884 support. If any new application is adding extensions it has to call defined registries to be compliant with RFC 4884.

Backward compatibility is also taken care of as part of this feature.

This feature is enabled by default and a CLI *[no]* **ipv6 icmp od-length enable** is provided which is enabled by default.

Command Example

```
ipv6 icmp od-length enable
```

Limitations

RFC4884 ICMP v4 and MPLS v4 extensions will be supported in the IOS-XE 17.6.1 release.

Netboot Support

The Netboot (TFTP boot) feature is now supported on the ESR6300. The ESR6300 has two Combo ports, Copper and Fiber ports (SFP) ports that support TFTP boot.

The Netboot (TFTP boot) feature allows the user to recover their router in the case that there is no image in the bootflash or USB.

The following configuration needs to be in place in ROMMON:

- WAN port Gigabit-Ethernet 0/0/0 or 0/0/1 should be connected to a TFTP network
- Path to image should be in a TFTP directory
- set IP_ADDRESS=<IP address of router>
- set IP_SUBNET_MASK=<mask>
- set DEFAULT_GATEWAY=<IP address of gateway>
- boot tftp://<server IP address>/<path to image>

Alarm port Support on the ESR6300

There is one alarm port available on the ESR6300. The IOS name for the alarm port is Alarm Contact 0.

The following configuration commands are available in IOS:

- alarm contact 0 enable
- alarm contact 0 description
- alarm contact 0 severity
- alarm contact 0 trigger

The configuration commands also have their equivalent **no** prefaces.

Alarm Contact Command

The ESR6300 supports only one alarm contact, which is Alarm Contact 0. Options are described in the following table:

description	The description string is up to 80 alphanumeric characters in length and is included in any generated system messages.
severity	For severity, enter critical, major, minor or none. If you do not configure a severity, the default is minor.
trigger	For trigger, enter open or closed. If you do not configure a trigger, the alarm is triggered when the circuit is closed.
enable	Provides a mechanism for you to enable or disable alarm conditions for a port.

Alarm Trigger Command

The **trigger** command has two options as shown below:

closed	Assert alarm when contact is closed. Closed means that no current flows through the contact (normally open contact). The alarm is generated when current does flow.
open	Assert alarm when contact is open. Open means that the normal condition has current flowing through the contact (normally closed contact). The alarm is generated when the current stops flowing.

Note See the Alarm Port Configuration Examples, on page 9 for command examples.

Alarm LEDs

The alarm LED behavior is described in the following table:

Severity	LED Status
Critical	Flashing Red
Major	Flashing Red
Minor	Red

Note The LED behavior depends on both the trigger configuration as well as the severity configuration. The LED behavior does not differentiate between the Critical and Major severity.

Alarm Port Configuration Examples

To configure the feature, the alarm contact 0 needs to be enabled first. Perform the following:

```
Router#config term
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#alarm contact 0 enable
Router(config)#alarm contact 0 description test
Router(config)#alarm contact 0 severity critical
```

Alarm Enable/Disable

The alarm needs to be enabled to configure the severity and trigger. The following example shows the errors when the alarm is not enabled:

```
Router(config)#alarm contact 0 trigger open
Alarm / Digital IO Port 0 is not enabled.

Router(config)#alarm contact 0 severity major
Alarm / Digital IO Port 0 is not enabled.
```

Enabling the Alarm and Setting the Severity

See the following example:

```
Router(config)#alarm contact 0 enable

Router(config)#alarm  contact 0 severity ?
  critical  Critical alarm severity
  major     Major alarm severity
  minor     Minor alarm severity
  none      No alarm severity

Router(config)#alarm contact 0 severity major
Router(config)#end
Router#
*Oct 16 14:54:54.518: %IIOT_ALARM_CONTACT-0-EXTERNAL_ALARM_CONTACT_ASSERT: External
alarm/digital IO port (External alarm contact on Motherboard) asserted

*Oct 16 14:54:54.518: %IIOT_ALARM_CONTACT-0-EXTERNAL_ALARM_CONTACT_ASSERT: External
alarm/digital IO port (ASSERT) asserted
ha
*Oct 16 14:54:54.733: %SYS-5-CONFIG_I: Configured from console by cons
```

Viewing The Configuration

To view the configuration:

```
Router#show alarm
Alarm contact 0:
Description: External alarm contact on Motherboard
Status: Not Asserted
Application: Dry
Severity: major
Trigger: Open
Mode: Input
Router#

Router#show facility-alarm status
System Totals  Critical: 0  Major: 1  Minor: 0
```

```
Source                    Time                  Severity      Description [Index]
------                    ------                --------      -------------------

External alarm contact    Oct 16 2023 14:46:14  MAJOR         External alarm contact on
Motherboard [0]

Async0/2/0                Oct 15 2023 18:58:08  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/0      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/1      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/0      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/1      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/2      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/3      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

Cellular0/3/0             Oct 15 2023 18:58:22  INFO          Physical Port Administrative
 State Down [2]

Cellular0/3/1             Oct 15 2023 18:58:22  INFO          Physical Port Administrative
 State Down [2]

Router#
Router#show facility-alarm status major
System Totals  Critical: 0  Major: 1  Minor: 0

Source                    Time                  Severity      Description [Index]
------                    ------                --------      -------------------

External alarm contact    Oct 16 2023 14:46:14  MAJOR         External alarm contact on
Motherboard [0]
```

Alarm Trigger Commands

See the following example:

```
Router#show run | sec alarm
alarm contact 0 enable
alarm contact 0 trigger Open
logging alarm informational
Router#
Router#show alarm
Alarm contact 0:
   Description: External alarm contact on Motherboard
   Status:      Asserted
   Application: Dry
   Severity:    minor
   Trigger:     Open
   Mode:        Input
Router#
Router#show facility-alarm status
System Totals  Critical: 0  Major: 0  Minor: 1
```

```
Source                    Time                  Severity      Description [Index]
------                    ------                --------      -------------------

External alarm contact    Oct 16 2023 14:54:54  MINOR         External alarm contact on
Motherboard [0]

Async0/2/0                Oct 15 2023 18:58:08  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/0      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/1      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/0      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/1      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/2      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/3      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

Cellular0/3/0             Oct 15 2023 18:58:22  INFO          Physical Port Administrative
 State Down [2]

Cellular0/3/1             Oct 15 2023 18:58:22  INFO          Physical Port Administrative
 State Down [2]

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#alarm  contact 0 trigger closed
Router(config)#
*Oct 16 14:58:19.548: %IIOT_ALARM_CONTACT-0-EXTERNAL_ALARM_CONTACT_CLEAR: External
alarm/digital IO port (External alarm contact on Motherboard) cleared

*Oct 16 14:58:19.549: %IIOT_ALARM_CONTACT-0-EXTERNAL_ALARM_CONTACT_CLEAR: External
alarm/digital IO port (CLEAR) cleared

Router#sh facility-alarm status
System Totals  Critical: 0  Major: 0  Minor: 0

Source                    Time                  Severity      Description [Index]
------                    ------                --------      -------------------

Async0/2/0                Oct 15 2023 18:58:08  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/0      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/0/1      Oct 15 2023 18:58:21  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/0      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/1      Oct 15 2023 18:58:20  INFO          Physical Port Administrative
 State Down [2]
```

```
GigabitEthernet0/1/2       Oct 15 2023 18:58:20   INFO         Physical Port Administrative
 State Down [2]

GigabitEthernet0/1/3       Oct 15 2023 18:58:20   INFO         Physical Port Administrative
 State Down [2]

Cellular0/3/0              Oct 15 2023 18:58:22   INFO         Physical Port Administrative
 State Down [2]

Cellular0/3/1              Oct 15 2023 18:58:22   INFO         Physical Port Administrative
 State Down [2]

Router#

Router#show alarm
Alarm contact 0:
   Description: External alarm contact on Motherboard
   Status:      Not Asserted
   Application: Dry
   Severity:    minor
   Trigger:     Closed
   Mode:        Input
```

New Features for Cisco IOS-XE 17.4.1

Plug and Play (PnP) Support

This release enables PnP to work the same as on the IR1101. See the IR1101 Software Configuration Guide.

PnP Overview

The out of box configuration boots the platform up to the configuration wizard. The control stops at a prompt where the user is given an option to enter the startup configuration wizard or not. If the user does not have access to the router, or does not enter any options, PnP discovery kicks in. If the PnP agent successfully establishes a connection to the PnP Server, the device configurations are pushed from the Server. The platform gets configured according to the user preference.

If PnP is not setup for the Router, the WebUI is accessible without having to access the platform console.

New Features for Cisco IOS-XE 17.3.1

Support for Security-Enhanced Linux (SELinux)

Security-Enhanced Linux is a set of patches to the Linux kernel and some utilities to incorporate a strong, flexible mandatory access control (MAC) architecture into the major subsystems of the kernel. SELinux provides an enhanced mechanism to enforce the separation of information based on confidentiality and integrity requirements, which allows threats of tampering and bypassing of application security mechanisms to be addressed and enables the confinement of damage that can be caused by malicious or flawed applications.

SELinux enforces mandatory access control policies that confine user programs and system servers to the minimum amount of privilege they require to do their jobs. This reduces or eliminates the ability of these

programs and daemons to cause harm when compromised (for example, via buffer overflows or mis-configurations). This confinement mechanism operates independently of the traditional Linux access control mechanisms.

The are no additional requirements or configuration steps required to enable or operate the SELinux feature. The solution is enabled/operational by default as part of the base IOS-XE software on supported platforms.

The following are enhanced show commands that have been defined for viewing SELinux related audit logs.

show platform software audit all

show platform software audit summary

show platform software audit switch *<<1-8> | active | standby> <FRU identifier from a drop-down list>*

Command Examples

The following is a sample output of the **show software platform software audit summary** command:

```
Device# show platform software audit summary
===================================
AUDIT LOG ON switch 1
-----------------------------------
AVC Denial count: 58
===================================
```

The following is a sample output of the **show software platform software audit all** command:

```
Device# show platform software audit all
====================================
AUDIT LOG ON switch 1
------------------------------------
========== START ============
type=AVC msg=audit(1539222292.584:100): avc:  denied  { read } for  pid=14017
comm="mcp_trace_filte" name="crashinfo" dev="rootfs" ino=13667
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_disk_crashinfo_t:s0 tclass=lnk_file permissive=1
type=AVC msg=audit(1539222292.584:100): avc:  denied  { getattr } for  pid=14017
comm="mcp_trace_filte" path="/mnt/sd1" dev="sda1" ino=2
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_disk_crashinfo_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539222292.586:101): avc:  denied  { getattr } for  pid=14028 comm="ls"
 path="/tmp/ufs/crashinfo" dev="tmpfs" ino=58407
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_ncd_tmp_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539222292.586:102): avc:  denied  { read } for  pid=14028 comm="ls"
name="crashinfo" dev="tmpfs" ino=58407 scontext=system_u:system_r:polaris_trace_filter_t:s0
 tcontext=system_u:object_r:polaris_ncd_tmp_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539438600.896:119): avc:  denied  { execute } for  pid=8300 comm="sh"
name="id" dev="loop0" ino=6982 scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
 tcontext=system_u:object_r:bin_t:s0 tclass=file permissive=1
========== END ============
```

(output omitted for brevity)

The following is a sample output of the **show software platform software audit switch** command:

```
Device# show platform software audit switch active R0
========== START ============
type=AVC msg=audit(1539222292.584:100): avc:  denied  { read } for  pid=14017
comm="mcp_trace_filte" name="crashinfo" dev="rootfs" ino=13667
```

```
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_disk_crashinfo_t:s0 tclass=lnk_file permissive=1
type=AVC msg=audit(1539222292.584:100): avc:  denied  { getattr } for  pid=14017
comm="mcp_trace_filte" path="/mnt/sd1" dev="sda1" ino=2
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_disk_crashinfo_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539222292.586:101): avc:  denied  { getattr } for  pid=14028 comm="ls"
 path="/tmp/ufs/crashinfo" dev="tmpfs" ino=58407
scontext=system_u:system_r:polaris_trace_filter_t:s0
tcontext=system_u:object_r:polaris_ncd_tmp_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539222292.586:102): avc:  denied  { read } for  pid=14028 comm="ls"
name="crashinfo" dev="tmpfs" ino=58407 scontext=system_u:system_r:polaris_trace_filter_t:s0
 tcontext=system_u:object_r:polaris_ncd_tmp_t:s0 tclass=dir permissive=1
type=AVC msg=audit(1539438624.916:122): avc:  denied  { execute_no_trans } for  pid=8600
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438648.936:123): avc:  denied  { execute_no_trans } for  pid=9307
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438678.649:124): avc:  denied  { name_connect } for  pid=26421
comm="nginx" dest=8098 scontext=system_u:system_r:polaris_nginx_t:s0
tcontext=system_u:object_r:polaris_caf_api_port_t:s0 tclass=tcp_socket permissive=1
type=AVC msg=audit(1539438696.969:125): avc:  denied  { execute_no_trans } for  pid=10057
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438732.973:126): avc:  denied  { execute_no_trans } for  pid=10858
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438778.008:127): avc:  denied  { execute_no_trans } for  pid=11579
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438800.156:128): avc:  denied  { name_connect } for  pid=26421
comm="nginx" dest=8098 scontext=system_u:system_r:polaris_nginx_t:s0
tcontext=system_u:object_r:polaris_caf_api_port_t:s0 tclass=tcp_socket permissive=1
type=AVC msg=audit(1539438834.099:129): avc:  denied  { execute_no_trans } for  pid=12451
comm="auto_upgrade_se" path="/bin/bash" dev="rootfs" ino=7276
scontext=system_u:system_r:polaris_auto_upgrade_server_rp_t:s0
tcontext=system_u:object_r:shell_exec_t:s0 tclass=file permissive=1
type=AVC msg=audit(1539438860.907:130): avc:  denied  { name_connect } for  pid=26421
comm="nginx" dest=8098 scontext=system_u:system_r:polaris_nginx_t:s0
tcontext=system_u:object_r:polaris_caf_api_port_t:s0 tclass=tcp_socket permissive=1
========== END ============
==================================
```

Syslog Message Reference

Facility-Severity-Mnemonic

- %SELINUX-3-MISMATCH

Severity-Meaning

- ERROR LEVEL Log

Message Explanation

- A resource access was made by the process for which a resource access policy is not defined. The operation was flagged but not denied.

- The operation continued successfully and was not disrupted. A system log has been generated about the missing policy for resource access by the process as denied operation.

Recommended Action

- Please contact CISCO TAC with the following relevant information as attachments:
  - The message exactly as it appears on the console or in the system log.
  - Output of "show tech-support" (text file)
  - Archive of Btrace files from the box using the following command ("request platform software trace archive target <URL>") For Example: Device#**request platform software trace archive target flash:selinux_btrace_logs**

SD-WAN on the ESR6300

The ESR6300 supports SDWAN with release 17.3.1 or later. This release brings the ESR6300 into feature parity with the IR1101. The ESR6300 will require controller version 20.2 or later.

All of the available SDWAN documentation can be found here:

https://www.cisco.com/c/en/us/support/routers/sd-wan/series.html

New Features for Cisco IOS-XE 17.2.1

Serial Port Support

Additional protocol capabilities have been added to the ESR6300 to bring it into feature compatibility with the IR1101. These include:

- SCADA Gateway functionality (IEC10x and DNP3)
- Raw Socket (TCP and UDP)
- Line Relay
- Reverse Telnet

All of the configuration and show commands will be the same as are available on the IR1101 platform.

https://www.cisco.com/c/en/us/td/docs/routers/access/1101/software/configuration/guide/b_IR1101config.html

Boot from the USB

Support has been added in order to boot the device from configuration files located on the pluggable USB. Customized startup configuration files can be booted from IOS or from ROMMON.

Booting from IOS

The following configuration steps need to be taken in order to boot from the USB.

To display the boot options:

```
Router(config)#boot config ?
  bootflash:  URL of the config file
  flash:      URL of the config file
  nvram:      URL of the config file
  usbflash0:  URL of the config file
  webui:      URL of the config file
```

The syntax for the boot command is:

boot config usbflash0:<*file name*>

For example:

```
Router(config)#boot config usbflash0:startup-config
Router(config)#
Router#write memory
Building configuration...
[OK]
*Feb 10 10:20:11.990: %SYS-2-PRIVCFG_ENCRYPT: Successfully encrypted private config file
```

The environment variable CONFIG_FILE in the following example confirms that the startup-config is set to boot from usbflash0.

```
Router#show boot
BOOT variable =
CONFIG_FILE variable = usbflash0:startup-config
BOOTLDR variable does not exist
Configuration register is 0x1820
Standby not ready to show bootvar
```

Booting from ROMMON

The following configuration steps need to be taken in order to boot from the USB.

From the ROMMON prompt, execute **set CONFIG_FILE=usbflash0:** <*filename*>

For example:

```
rommon 2 > set CONFIG_FILE=usbflash0:my_startupcfg
rommon 3 > sync
rommon 4 > set
PS1=rommon ! >
MCU_UPGRADE=SKIP
THRPUT=
LICENSE_BOOT_LEVEL=
RET_2_RTS=
MCP_STARTUP_TRACEFLAGS=00000000:00000000
BSI=0
RANDOM_NUM=1275114933
BOOT=flash:Jun5_1.SSA,12
RET_2_RCALTS=951454376
CONFIG_FILE=usbflash0:my_startupcfg
```

Continue booting the IOS image as usual from the ROMMON prompt.

Booting from the USB Feature Summary

- Once the CONFIG_FILE is set to a non-default value, the **nvram:startup-config** command is aliased to this new location.
- Any change made to the config file in usbflash will be reflected in nvram:startup-config as well.
- The EXEC command **erase nvram:startup-config** erases the contents of NVRAM, and deletes the file referenced by CONFIG_FILE variable.
- If the USB is unplugged after setting the **boot config usbflash0:** <*filename*> variable, then the day 0 default configuration will take effect.
- When the configuration is saved using the **copy system:running-config nvram:startup-config** command, the device saves a complete version of the configuration file to the location specified by the CONFIG_FILE environment variable, and a distilled version to NVRAM. A distilled version is one that does not contain access list information.