# New Features for Cisco IOS-XE 17.2.1

# Native docker support

Native Docker Support has been added to the 17.2.1 release. This feature enables users to deploy the docker applications on the IR1101. The application lifecycle process is similar to the procedure in the Installing and Uninstalling Apps section. For docker applications, entry point configuration is required as part of the application configuration. Please refer to the following example for the entry point configuration.

```
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#app-hosting appid app3
Router(config-app-hosting)#app-vnic gateway0 virtualportgroup 0 guest-interface 0
Router(config-app-hosting-gateway0)#guest-ipaddress 192.168.0.7 netmask 255.255.255.0
Router(config-app-hosting-gateway0)#app-default-gateway 192.168.0.1 guest-interface 0
Router(config-app-hosting)#app-resource docker
Router(config-app-hosting-docker)#run-opts 1 "--entrypoint '/bin/sleep 10000'"
Router(config-app-hosting-docker)#end
Router#
```

The output for docker applications is shown in the following example:

```
Router#show app-hosting detail
App id : app1
Owner : iox
State : RUNNING
Application
Type : docker
Name : aarch64/busybox
Version : latest
Description :
Path : bootflash:busybox.tar
Activated profile name : custom
Resource reservation
Memory : 431 MB
Disk : 10 MB
```

```
CPU : 577 units
VCPU : 1
Attached devices
Type Name Alias
---------------------------------------------
serial/shell iox_console_shell serial0
serial/aux iox_console_aux serial1
serial/syslog iox_syslog serial2
serial/trace iox_trace serial3
Network interfaces
---------------------------------------
eth0:
MAC address : 52:54:dd:e9:ab:7a
IPv4 address : 192.168.0.7
Network name : VPG0
Docker
------
Run-time information
Command :
Entry-point : /bin/sleep 10000
Run options in use : --entrypoint '/bin/sleep 10000'
Application health information
Status : 0
Last probe error :
Last probe output :
Router#
```

# Yang Data Model Support for Raw Socket Transport

Release 17.2.1 adds support for additional Yang Data Models. These additional models include Raw Socket Transport.

Yang Data Models can be found here:

https://github.com/YangModels/yang/tree/master/vendor/cisco/xe/1721

There are two feature modules available for raw socket that belong to the main Cisco-IOS-XE-native model. They are:

- Cisco-IOS-XE-rawsocket.yang

This module contains a collection of YANG definitions for Raw Socket Transport Configuration commands.

This module has the following corresponding Cli commands:

```
# encapsulation raw-tcp
# encapsulation raw-udp
# raw-socket packet-length <length>
# raw-socket packet-timer <timer>
# raw-socket special-char <value>
# raw-socket tcp server <port> <ip>
# raw-socket tcp idle-timeout <value>
# raw-socket tcp client <dest-ip> <dest-port>
# raw-socket tcp idle-timeout <timeout>
# raw-socket tcp tcp-session <value>
# raw-socket tcp dscp <value>
# raw-socket udp connection <dest-ip> <dest-port> <local_port>
```

- Cisco-IOS-XE-rawsocket-oper.yang

This module contains a collection of YANG definitions for Raw Socket Transport operational data.

This module has the following corresponding Cli commands:

```
# show raw udp statistics
# show raw tcp statistics
# show raw tcp session
# show raw udp session
# show raw tcp session local
# show raw udp session local
```

The following is a list of the Dependent Modules:

- Cisco-IOS-XE-native
- Cisco-IOS-XE-features
- ietf-inet-types
- Cisco-IOS-XE-interfaces
- Cisco-IOS-XE-ip
- Cisco-IOS-XE-vlan
- ietf-yang-types @ (any revision)
- cisco-semver

# Digital IO for IOx container applications

Release 17.2.1 provides support for IOx container applications to be able to access the digital IO. There is a new CLI that has been added to the alarm contact command.

```
Router(config)# alarm contact ?
  <0-4>         Alarm contact number (0: Alarm port, 1-4: Digital I/O)
  attach-to-iox  Enable Digital IO Ports access from IOX
Router (config)# alarm contact attach-to-iox
```

Enabling the **attach-to-iox** command will provide complete control of all Digital IO ports to IOx. The ports will be exposed as four character devices /dev/dio-[1-4] to IOX applications. You can use read/write functions to get/set values of the Digital IO ports.

If you wish to update the mode, you can write the mode value to the character device file. This is accomplished by IOCTL calls to read/write the state, change mode, and read the true analog voltage of the port. Following this method, you can attach analog sensors to the IR1101. All ports are initially set to Input mode with voltage pulled up to 3.3v.

The following are examples of IOCTL calls:

### Read Digital IO Port

```
cat /dev/dio-1
```

### Write to Digital IO Port

```
echo 0 > /dev/dio-1
echo 1 > /dev/dio-1
```

### Change mode

```
echo out > /dev/dio-1
echo in > /dev/dio-1
```

### List of IOCTLs supported

```
DIO_GET_STATE = 0x1001
DIO_SET_STATE = 0x1002
DIO_GET_MODE = 0x1003
DIO_SET_MODE_OUTPUT = 0x1004
DIO_SET_MODE_INPUT = 0x1005
DIO_GET_THRESHOLD 0x1006
DIO_SET_THRESHOLD = 0x1007
DIO_GET_VOLTAGE = 0x1009
```

### Read State using IOCTL

```
import fcntl, array
file = open("/dev/dio-1","rw")
state = array.array('L',[0])
fcntl.ioctl(file, DIO_GET_STATE, state)
print(state[0])
```

### Change mode using IOCTL

```
import fcntl
file = open("/dev/dio-1","rw")
fcntl.ioctl(file, DIO_SET_MODE_OUTPUT, 0)
```

# L2 Sticky Secure MAC Addresses

This is a new feature for the IR1101, however, it been present in IOS-XE for some time.

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

## Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.

- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

**Note** If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.
- shutdown—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the errdisable recovery cause psecure-violation global configuration command, or you can manually re-enable it by entering the shutdown and no shut down interface configuration commands. This is the default mode.
- shutdown vlan—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

## Command Line Interface

Under switch interface, add port-security cli.

```
Router(config-if)#switchport port-security ?
  aging        Port-security aging commands
  mac-address  Secure mac address
  maximum      Max secure addresses
  violation    Security violation mode
  <cr>         <cr>
Router(config-if)#switchport port-security mac-address sticky
```

# Signed Application Support

Cisco Signed applications are now supported on the IR1101. In order to install a signed application, signed verification has to be enabled on the device. Signed verification can be enabled by following the following instructions.

```
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Router(config)#
Router(config)#app-hosting signed-verification
Router(config)#
Router(config)#exit
```

After enabling the signed verification, follow the instructions in the Installing and Uninstalling Apps section under IOx Application Hosting in order to install the application.