



# Release Notes for Cisco onePK Version 1.1, Controlled Availability

Last Updated: December 17, 2013

One Platform Kit (onePK) is a cross-platform application programming interface (API) and software development kit (SDK) that enables you to develop applications that interact directly with Cisco networking devices. It provides you with the ability to access networking services using a set of controlled APIs that share the same programming model and style.



Note

The software supported by this release (see the Compatibility Matrix) is subject to Controlled Availability for onePK. Use of onePK outside of the Controlled Availability program is not supported. Please contact your account representative to ensure your organization is registered as part of the onePK Controlled Availability program prior to using this feature.

Language support for onePK version 1.1 includes C and Java, and Python.

- [Compatibility Matrix, page 4](#)
- [How to Get Access to onePK, page 5](#)

## Compatibility Matrix

The table below shows minimum hardware and software requirements for Cisco onePK 1.1.

Element	Software Version
<b>Cisco Devices</b>	
Cisco ASR 1000 and ISR 4400 Series Routers	Cisco IOS XE 3.11.0S (Special)
Cisco ISR G2	Cisco IOS Release 15.4(1)T (Special)

Element	Software Version
Nexus 3000 Series Switches	Cisco NX-OS Release 6.0(2)U2(1)
Nexus 5000 and 6000 Series Switches	Cisco NX-OS Release 6.0(2)N3(1)
Cisco ASR 9000 Series Routers	Cisco IOS XR Release 5.1.1
<b>Development Workstation and Process Hosting</b>	
Workstation operating system	Any POSIX Linux environment with a minimum kernel version of 2.6.x
Glibc	2.15
GNU Compiler Collection (GCC)	Version 4.5.x
GNU Make	3.4.6+
Java	1.6.0_31
Eclipse	Indigo Service Release 2 or later
Maven	3.0.3
Python	2.7
<b>All-in-One Virtual Machine</b>	
Virtualization software	<ul style="list-style-type: none"> <li>• VMWare ESXi 5 or later</li> <li>• VMWare Workstation 9 or later</li> <li>• VMWare Fusion 5 or later</li> <li>• Oracle VirtualBox 4.2 or later</li> </ul>
Workstation hardware minimum requirements	<ul style="list-style-type: none"> <li>• 2 virtual CPUs</li> <li>• 4 GB RAM</li> <li>• 20 GB disk space</li> </ul>

## How to Get Access to onePK

Cisco onePK 1.1 is currently in Controlled Availability. To participate in this program, please register at <http://developer.cisco.com/web/onepk-developer/web-form>.

### Downloading onePK

Once you have been accepted into the onePK Controlled Availability program, you will gain access to the onePK section of the [Cisco Developer Network](#). There you will find:

- Software downloads for the onePK SDK
- Technical documentation
- Sample use cases
- onePK-centric blogs and forums

## Downloading Cisco IOS and Nexus Software Releases

The software required to run onePK applications on your router or switch is subject to the onePK Controlled Availability program. You can find links to this software on Cisco's DevNet portal

## What's New in onePK

### onePK New APIs/Service Sets

- Policy API enhancements

### Additional Language support

- Python

### Additional Platform support

- ISR G2 8xx and 19xx
- ISR 4451
- Nexus 3000, 5000, and 6000
- ASR9000

## Service Set Coverage by Platform

The following table shows which service sets are available on a given platform.

*Table 1 Service Set Platform Coverage*

Service Set	Nexus 3000	Nexus 5000 & Nexus 6000	ISR G2	ISR 4400	ASR 1000	ASR 9000
Data Path			X			X
Policy	X	X	X	X	X	X
Routing	X	X	X	X	X	X
Element	X	X	X	X	X	X
Discovery	X	X	X	X	X	X
Utility	X	X	X	X	X	X
Developer	X	X	X	X	X	X
Location					X	
MediaTrace (PathTrace)			X			
Identity (SANET)			X			

## Service Set Coverage by Language

The following table shows which service sets are supported by a given language.

Table 2 Service Set Language Coverage

Service Set	C	Java	Python
Data Path	X		
Policy	X	X	X
Routing	X	X	X
Element	X	X	X
Discovery	X	X	X
Utility	X	X	X
Developer	X	X	X
Location	X	X	
MediaTrace (PathTrace)	X	X	
Identity (SANET)	X	X	

### Important Notes

#### TCP and SOCKET Transport is No Longer Supported

The onePK SDK now requires that all onePK applications use TLS to communicate with network devices. Cisco routers and switches will reject attempts to establish unencrypted connections.

#### Some Header File Names and Data Structures Have Changed

Due to ongoing product development, it has become necessary to make some minor changes:

- **datapath\_policy.h** has been changed to **onep\_datapath\_policy.h**
- **onep\_types\_backward.h** now includes **typedef**, which contains old-to-new name mappings. Your application code does not need to change, but you will need to add **typedef** to your application. For example: `typedef onep_network_element_t network_element_t`

#### Transient and Persistent Table Type Support in Policy Maps

Policy maps and class maps can either be transient or persistent. Transient policy maps and class maps conform to the following behavior:

- Data added or changed does not show up in running configuration.
- Data added or changed cannot be written to start up configuration.
- Data added or changed is removed when the application exits.
- Data added or changed cannot be removed via CLI.

Persistent policies persist across application session exit. They will also persist across network element reloads, provided they have been written to the startup configuration by the network administrator.

All policy maps and class maps are transient by default. For tables that only support persistent behavior, the application has to explicitly set the policy map and class map as persistent.

Table Name	IOS-XR	IOS	NX-OS (Nexus 3000 & 6000)
ONEP_POLICY_TABLE_TYPE_L2_ANY	Transient		
ONEP_POLICY_TABLE_TYPE_L3_IPV4	Transient		Transient
ONEP_POLICY_TABLE_TYPE_L3_IPV6			
ONEP_POLICY_TABLE_TYPE_L2_L3	Transient		
ONEP_POLICY_TABLE_TYPE_L3_DS	Transient		
ONEP_POLICY_TABLE_TYPE_QOS_INGRESS	Persistent	Transient	Transient
ONEP_POLICY_TABLE_TYPE_QOS_EGRESS	Persistent	Transient	
ONEP_POLICY_TABLE_TYPE_PBR	Persistent		
ONEP_POLICY_TABLE_TYPE_TRAFFIC_INGRESS	Persistent		
ONEP_POLICY_TABLE_TYPE_TRAFFIC_EGRESS	Persistent		
ONEP_POLICY_TABLE_TYPE_MAC			Transient
ONEP_POLICY_TABLE_TYPE_DATAPATH		Transient	

### Process Hosting and Application Developer Toolkit Not Supported in Version 1.1

The onePK SDK and documentation make references to Process Hosting and the Application Developer's Toolkit (ADT). These features are still in Early Field Trial and not part of this Controlled Availability release.

### Public and Private APIs in Java

This version of onePK includes a number of classes, members and methods in the API that are marked as public, but have an `@pad.exclude` annotation in the Javadoc. This annotation has the effect of preventing the appearance of these classes, members and methods in the generated Javadoc, but they will still appear in an IDE, for example Eclipse, when viewing the classes in question, as the `@pad.exclude` annotation does not affect the code itself.

Such classes, members or methods are *not* intended to be part of the public API. Cisco reserves the right to change the visibility of, or remove, any such class, member or method, that has an `@pad.exclude` annotation, in future releases.

### Support for Interface Statistics and Events Varies by Platform

The following table lists platform support for Interface Statistics APIs and events.

Interface Statistics onePK API	NXOS API	IOS API	XR API	NXOS Event	IOS Event	XR Event
onep_interface_statistics_get_reliability()	No	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_transmit_load()	No	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_transmit_rate_pps()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_transmit_rate_bps()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_transmit_cum_byte()	Yes	Yes	Yes	No	No	No
onep_interface_statistics_get_transmit_cur_cum_byte()	No	No	No	No	No	No
onep_interface_statistics_get_receive_load()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_receive_rate_pps()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_receive_rate_bps()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_receive_cum_byte()	Yes	Yes	Yes	No	No	No
onep_interface_statistics_get_receive_cur_cum_byte()	No	No	No	No	No	No
onep_interface_statistics_get_receive_broadcast()	Yes	Yes	Yes	Yes	Yes	Yes
onep_interface_statistics_get_receive_multicast()	Yes	Yes	Yes	No	No	No
onep_interface_statistics_get_receive_unicast()	Yes	Yes	Yes	No	No	No
onep_interface_statistics_get_receive_runt()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_receive_giant()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_receive_throttle()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_in_unknown_protocol_drop()	No	Yes	No	No	No	No
onep_interface_statistics_get_in_packet_drop()	No	Yes	No	Yes	Yes	Yes
onep_interface_statistics_get_in_error()	Yes	Yes	Yes	Yes	Yes	Yes
onep_interface_statistics_get_in_error_crc()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_in_error_frame()	No	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_in_error_overrun()	Yes	Yes	Yes	No	Yes	Yes
onep_interface_statistics_get_in_error_ignore()	Yes	Yes	Yes	No	No	No
onep_interface_statistics_get_out_packet_drop()	No	Yes	No	Yes	Yes	Yes
onep_interface_statistics_get_out_error()	Yes	Yes	Yes	Yes	Yes	Yes
onep_interface_statistics_get_out_error_underrun()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_out_error_reset()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_out_buffer_fail()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_out_buffer_swap()	No	Yes	No	No	Yes	Yes
onep_interface_statistics_get_tail_drops ()	No	No	No	No	No	No
onep_interface_statistics_get_voq_drops()	No	No	No	No	No	No
onep_interface_statistics_get_replication_engine_drops()	No	No	No	No	No	No

### Python SDK Does not Support Bidirectional Authentication

The Python SDK supports only unidirectional authentication. The SDK does not support client certificates at this time.

### Events Tutorial Requires EVMED Feature on NX-OS Devices

To configure a Nexus 3000 to run an Events tutorial, you must first enable the EVMED feature, which allows developers to use Embedded Event Manager (EEM) to monitor Application-, CLI-, and Interface Statistics-based events and then schedule actions on the occurrence of those events:

```
router(config)# feature evmed
router(config)# show feature |i evmed
evmed enabled
```

## Open Caveats

The following caveats are open in Cisco onePK 1.1.

### Severity 2 Caveats

#### CSCuj20715

Symptoms: Prior to adding an Application Route, if the "redistribute" command exists, the application-added routes are not redistributed to OSPF.

Conditions: The application-added routes will not be redistributed to OSPF and are not advertised.

Workaround: Negating the "redistribute" command and adding back the command fixes the issue.

#### CSCuj28981

Symptoms: The onep\_interface\_get\_description API does not work for VLAN interfaces.

Conditions: Occurs when you try to get interface description for VLAN interfaces via the onePK API on a Nexus 5000 or Nexus 6000.

Workaround: There is no workaround.

#### CSCuj97430

Symptoms: OnePK interface API returns failure, or OnePK ART route injection API does not update the routing table correctly.

Conditions: This situation is more likely to occur in large-scale environments. Examples are injecting large number of routes into ART table or invoking a large number of interface queries API in short period of time.

Workaround: The clear ip route \* command will recover missing ART route in the routing table. There is no workaround for other failure cases.

#### CSCul53892

Symptoms: The XOS Interface Config API does not offer the functionality equivalent of CLI "no ip address", and "no ipv6 address", which results in clients being unable to clear all ipv4 or ipv6 addresses in one call.

Conditions: This always occurs.

Workaround: Applications can iterate all addresses on the box, and remove them one by one.

#### CSCul58706

SymptomS: IPV6 ANYCAST and Linklocal address set is not supported.

Conditions: This always occurs.

Workaround: There is no workaround.

#### CSCul88004

Symptoms: DPSS packet inject fails.

Conditions: Occurs when the onePK application name contains spaces.

Workaround: Rename the application to eliminate any spaces and non-ASCII printable characters.

#### **CSCul98461**

Symptoms: The onep\_server may crash if polling interface property by com.cisco.onep.interfaces.InterfaceProperty.

Conditions: Occurs when a large number of interfaces is polled.

Workaround: There is no workaround.

#### **CSCum04858**

Symptoms: Memory leak is observed when adding routes on NX-OS routers using onePK route add APIs.

Conditions: Memory leak is seen when programming routes using onePK API when an application connects and disconnects several times and does route add during each session. Memory leak does not seem to increase depending on the number of routes programmed during the application session.

Workaround: Minimize application connection flaps to reduce memory leak.

#### **CSCum07119**

Symptoms: Router generates tracebacks or crashes depending on platforms when **show application ip route** command is used concurrently with application route deletion.

Conditions: Occurs when **show application ip route** command is issued when Java SDK is handling route replace operations.

Workaround: Do the following:

- 1 Use **show ip route** to display the application routes and not **show application ip route**.
- 2 Use **onePK GET ROUTE API** to get the status of application added route.
- 3 Use **show application ip route** only when there is no route delete in progress.

#### **CSCum07351**

Symptoms: The onep process crashes.

Conditions: Occurs with the following configuration:

```
config terminal
onep
service set vty
no service set vty
```

If you enter a command, like **show onep status**, the device will crash.

Workaround: The situation can be avoided if service discovery listeners are disabled before enabling or disabling service sets via configuration. However, if a service discovery listener is active during the configuration change, then you must reload device.

#### **CSCum10672**

Symptoms: ACL is incorrectly applied on the connected interface between a Nexus 5000 and Linux server running the ACL tutorial.

Conditions: Occurs when running ACLTutorial.py with a Nexus 5000.

Workaround: There is no workaround.



**Severity 3 Caveats****CSCue03027**

Symptoms: When entering the command `copy running-config bootflash:config/xyxy2`, the following message is displayed: Permission denied. Cannot overwrite existing file.

Conditions: Occurs when saving running-config to a file using the VTY service set when that file already exists and was created without the VTY service set.

Workaround: Write copy running-config to a new file when using this command from the VTY service set.

**CSCuf52747**

Symptoms: Some platforms do not return an IPv6 address when the interface is down.

Symptoms: For devices running Cisco IOS, the interface must be up. For devices running Cisco IOS-XR and the Cisco Nexus 3000, the IPv6 address is returned regardless of interface state.

Workaround: There is no workaround.

**CSCui25370**

Symptoms: CLI commands added by a container-based onePK application may not be removed when a container fails to install following a reboot.

Conditions: A container-based onePK application adds CLI commands to the router using the onePK CLI extensions feature. Upon reboot of the router the container fails to install properly. The CLI commands added prior to rebooting still remain even though the container is no longer installed.

Workaround: Installing and uninstalling the container for a second time will clean up the remaining CLI commands.

**CSCui38291**

Symptoms: IF-RESET events are not triggered.

Conditions: While testing with interface stats eventing, `ONEP_IF_STAT_RESETS` is not triggering an event when the interface is reset.

Workaround: Removing the polling interval of 60 did show the triggered event.

**CSCui90913**

Symptoms: cthrift error: -65553 'required parameter missing'.

Conditions: Error is seen in some instances of running a Java onePK application against Location Service Set.

Workaround: There is no workaround.

**CSCui96317**

Symptoms: OnePK applications cannot perform routing operations on platforms running NXOS.

Conditions: Occurs when the application name is 32 characters or longer.

Workaround: Change the application so it uses a shorter string as its onePK application name when connecting to NXOS platforms. For using onePK routing APIs on NXOS network elements, it is recommended to keep the application name to 21 characters or less.

**CSCuj08205**

Symptoms: A call to `onep_routing_l3_ucast_next_hop_get_ec` to get the next-hop EC after an async operation unexpectedly returns 0.

Conditions: After a route operation, the expected error codes returned by `onep_routing_l3_ucast_next_hop_get_ec` are:

1 - ROUTING\_ROUTE\_EC\_SUCCESS

4 - ROUTING\_ROUTE\_EC\_FAILURE

After a call to the async API, the `next_hop_ec` is returned as 0 instead of 1 or 4

Workaround: Use the sync API.

### **CSCuj09996**

Symptoms: If `onep` is enabled through a container, the status field does not reflect correct information. It shows that `onep`

was enabled by config instead of being enabled automatically by a container-based application.

Conditions: Information is shown in output from the `show onep status` command:

```
Router# show onep status
Status: enabled by: Config <-----
Version: 1.1.0
Transport: tcp; Status: disabled
Transport: tls; Status: disabled
Transport: tipc; Status: running; Port: 15003
Session Max Limit: 10
```

Workaround: Use the `show run` command to determine if `onep` is enabled via configuration. If `onep` is running but not enabled via configuration, then it was started by a local application.

### **CSCuj61067**

Symptoms: Delete punt flow does not delete the service node.

Conditions: Occurs when a onePK application is using Policy Service Set to capture flows.

Workaround: This causes no functionality issues. There is no workaround.

### **CSCuj84320**

Symptom: On the ASDR9000 RSP440 (Intel X86 RP), IPv4 addresses are returned in host order instead of network order. Also for linklocal prefix returned in 0 for IPv6 Linklocal addresses.

Conditions: This occurs under all conditions.

Workaround: There is no workaround.

### **CSCul10252**

Symptoms: When a Cisco Discovery Protocol (CDP) neighbor is deleted from a device, its topology service set should delete the node in its graph, and a notification should be provided for the event. Currently, no `NODES_DELETE` event notifications are being provided.

Conditions: Occurs when a CDP neighbor is deleted from a device.

Workaround: There is no workaround.

### **CSCul22974**

Symptoms: Python topology graph does not report two links to the same node.

Conditions: This always occurs.

Workaround: There is no workaround.

#### **CSCul30348**

Symptom: The `onep_dpss_redirect_flow()` API will only redirect the flow in the DPSS MP and not be able to redirect a flow in the Cisco ASR 9000.

Conditions: When a oneP application calls `onep_dpss_redirect_flow()`.

Workaround: Call `onep_dpss_unbypass_flow()` after `onep_dpss_redirect_flow()` and `dpss_mp` will handle all packet redirection instead of the platform.

#### **CSCul33377**

Symptoms: Timeout takes more time than the configured timeout value

Conditions: Occurs when using the `onep_vty_set_timeout` API.

Workaround: Use a timeout value larger than the actual timeout value required .

#### **CSCul36476**

Symptoms: VTY timeout not working over TLS for reload command in `vtty_write()` Only applicable to "reload" command through `vtty_write()` when timeout is set.

Conditions: Occurs when the VTY service set is used with the reload command and a timeout has been set.

Workaround: Use a session connect handler when reloading the router.

#### **CSCul39365**

Symptom: If an ACL is applied to both ingress and egress directions on an interface on a Cisco ASR9000 Series Aggregation Services Router, and if the application removes the ACL from interface by removing it first from one direction and then from the other direction, then the second ACL removal fails.

Conditions: This problem occurs when the same ACL is applied in both egress and ingress directions on an interface and the ACL is removed from the interface one at a time from each direction.

Workaround: Remove the ACL by specifying `ONEP_DIRECTION_BOTH` in `onep_acl_remove_from_interface` API. This will remove the ACL from both the directions.

#### **CSCul43749**

Symptoms: Router state added by the OpenFlow application is not removed upon deactivation of the k9sec PIE.

Conditions: This affects IOS-XR. OpenFlow creates state on the router, such as policy maps. While OpenFlow is running the "install deactivate k9sec pie" command is done to deactivate the k9sec PIE. OpenFlow and the onep process are both shutdown, however the state previously created by OpenFlow remains on the router.

Workaround: Unconfigure OpenFlow before deactivating the k9sec PIE.

#### **CSCul53513**

Symptoms: When setting out-of-bounds values for `setMTU`, no errors are being thrown for values that are out of range.

Conditions: This always occurs.

Workaround: There is no workaround. This has no functional or negative impact. When out-of-bound ranges are passed in they are ignored by the API.

#### **CSCul53643**

Symptoms: The clientkey with passphrase connection is not allowed

Conditions: Occurs with two-way TLS connections

Workaround: Generate clientrequest without passphrase. One way to do it is to enter the following command on any Linux host:

```
/usr/bin/openssl req -new -days 365 -nodes -out clientreq.pem -keyout clientkey.pem
```

### **CSCul62362**

Symptoms: A TTransportException: Cannot write to null outputStream is thrown when using ACL.getName()

Conditions: Occurs when invoking ACL.getName() on certain platforms that do not support this capability.

Workaround: Avoid using this API.

### **CSCul63893**

Symptoms: Some interface stats are incorrectly reported as unsupported:

In Packet Drop: NOT\_SUPPORTED

In Unknown Protocol: NOT\_SUPPORTED

Out Packet Drop: NOT\_SUPPORTED

Conditions: Occurs only when using the Python SDK. These stats are correctly reported by the C and Java SDKs.

Workaround: There is no workaround.

### **CSCul66860**

Symptoms: Dynamic memory increases on a router with each call of onep\_topology APIs.

Conditions: Occurs when running the show processes memory command.

Workaround: Some of the dynamic memory is used by the history archive. You can free this memory up by entering the clear onep history all command. There is no way to clear the remaining dynamic memory.

### **CSCul69749**

Symptoms: OneP Routing Service Set can only add/delete/replace 215 application routes on NX-OS platforms, and 409 application routes on Cisco XR platforms. If a batch update exceeds those numbers, the remaining routes will not get updated.

Conditions: Occurs when an application sends more than 215 application routes on NX-OS or 409 routes on Cisco XR. These numbers are memory dependent, and they will be lower when more than one nexthop is specified or if there are IPv6 routes.

Workaround: Use a smaller number of routes in each API call than MAX batchsize

### **CSCul84640**

Symptom: Interface delete notifications are not sent back to the client.

Conditions: Occurs when a filter is set on an interface type,

Workaround : The client must cache the interface handle, name and type. Register to listen on delete notifications without any filters and compare handles in the application context to filter on the type.

### **CSCul88050**

Symptoms: For CDP events, the IPv6 address is showing incorrectly.

Conditions: When CDP events are notified to applications, the IPv6 address shows only the first 64-bit address.

Workaround: The application can use the "interface get address APIs" to obtain the correct IPv6 addresses after receiving the CDP events.

#### **CSCul88343**

Symptoms: "Permission denied" errors occur in log file for dpss\_mp, and applications fail to initialize correctly.

Conditions: Occurs when user has started onePK application using DPSS as the root user or another user that is not in the same group as the GROUP setting in the dpss.conf file.

Workaround: Take the following actions:

- 1 Stop dpss\_mp and any onePK apps using the DPSS.
- 2 As root user, remove the file /dev/shm/dpss\_pak\_buf (or maybe some other name if it was modified in dpss.conf).
- 3 Remove all files matching /tmp/.dpss\* (may have to be done as root user)
- 4 (following points apply to AiO VM, can be logically extended to any manual deployment)
- 5 edit dpss.conf to set user/group to onepk/cisco
- 6 as root user, restart dpss\_mp
- 7 run apps as user/group cisco/cisco
- 8 At this point, apps should be able to run successfully. Any user within the same group as the group mentioned in the dpss.conf file should be adequate.
- 9 It is recommended that the dpss.conf file in the AiO VM be updated to ensure that the user is "onepk" and the group is set to "cisco".

#### **CSCul89805**

Symptoms: Interface description gets truncated to 199 characters in total.

Conditions: This always occurs.

Workaround: Use an interface description less than 200 characters.

#### **CSCul91512**

Symptoms: The valid range of an ACE sequence number for an L3 ACL is 1 - 4294967295, but, when its set beyond 2147483647 and the add ACE to ACL API is executed, error "'i' format requires -2147483648 \<= number \<= 2147483647" is reported.

Conditions: Occurs when "add ace to ACL" API using an ACE with a sequence number larger than 2147483647.

Workaround: There is no workaround.

#### **CSCul92371**

Symptoms: Application applies an ACL to an interface using invalid direction parameter. An error is returned but the ACL loses its assigned ACE configuration and cannot be deleted.

Conditions: Occurs when applying an ACL to an interface using invalid direction parameter.

Workaround: Always verify direction parameter is correct before applying ACL to interface.

#### **CSCul92398**

Symptoms: While applying TCP flags to Layer 3 access control list, the TCP mask list is not taken into account.

Conditions: Occurs on the Nexus 3000.

Workaround: There is no workaround. If TCP flag value and TCP mask are provided, the TCP values will reflect for an ACL irrespective of which TCP mask value is provided. To apply TCP flags for an ACL, make sure to provide the same set of flags in TCP flag value.

#### **CSCul93604**

Symptoms: While adding an ACE to ACL, when the port address is set to more than 32767, it gives an error saying 'h' format requires -32768 \<= number \<= 32767

Conditions: Occurs when the port address exceeds 32767.

Workaround: There is no workaround. Keep the port address in the range of 0-32767.

#### **CSCul95003**

Symptoms: OnePK Python SDK does not have the set interface description API implemented. There is no way to set the interface description.

Conditions: Occurs only with Python.

Workaround: Use the CLI or OnePK Python SDK VTY service set to set interface description.

#### **CSCul96090**

Symptoms: Results of `onep_topology_graph_get_edge_list` and `onep_topology_graph_get_node_list` are empty for graphs created from LLDP

Conditions: Occurs when you use these topology APIs to get LLDP information.

Workaround: Use show command output instead of `onep` applications for LLDP. Use CDP for topology `onePK` APIs instead of LLDP.

#### **CSCul97847**

Symptoms: Missing `EDGES_DELETE` notifications with Topology APIs on Nexus 3000 platform.

Conditions: Occurs when a CDP-enabled link is shut down at either end, or CDP is removed from the link at either end, or globally at either end of the links.

Workaround: There is no workaround. This is not seen in all setups, and issue is reported by only one user.

#### **CSCul98631**

Symptoms: Control characters are seen in `onep` topology service set notifications, specifically at the end of node connector names for the peer side of the edge.

Conditions: Occurs with most (not all) Python-based event notifications that use the API `onep.topology.NodeConnector` name.

Workaround: Ignore the last character of the node connector name. Since this is not a consistent problem, using some tool that recognizes the character and strips it off would be better.

#### **CSCum00859**

Symptoms: A Nexus 3000 may crash if the `onep` process is stopped while `onePK` applications are running.

Conditions: Occurs when a user tries to execute 11 `onep` applications and then disables the `onep` process.

Workaround: Do not stop the `onep` process while any application sessions are active.

#### **CSCum01124**

Symptoms: Memory leak occurs over a period of time when the sample Routing application is extracting several thousand OSPF, RIP, or static routes.

Conditions: This can be observed by running the show system internal onep memstat command. You need to have more routes to see visible leak over time.

Workaround: There is no workaround.

### **CSCum06508**

Symptoms: Removing a L2 ACE from an L2 ACL causes a failure.

Conditions : Occurs when you remove a L2 ACE while leaving the ACL in effect.

Workaround: Remove the entire L2 ACL and reconstruct a new one with a new L2 ACE.

### **CSCum07103**

Symptoms: When interface events are reported for "shut" interface, the lineproto state is reported as ONEP\_IF\_STATE\_ADMIN\_DOWN instead of ONEP\_IF\_STATE\_OPER\_DOWN.

Conditions: This is specific to "shut" events for interfaces.

Workaround: There is no workaround. The application should treat the ONEP\_IF\_STATE\_ADMIN\_DOWN state as ONEP\_IF\_STATE\_OPER\_DOWN state for lineproto events notifications.

### **CSCum07407**

Symptoms: Two-rate three- color policing works fine. But one-rate two-color policing gives error as "Error occurred in the operation. Illegal or inappropriate argument. pir must be between 8000 and 80000000000 in bits". Also it does not allow to configure cir in percentage and gives error as "Error occurred in the operation. Illegal or inappropriate argument. cir must be between 8000 and pir in bits"

Conditions: Occurs when configuring police action for policy-map.

Workaround: There is no workaround.

### **CSCum06935**

Symptoms: An invalid configuration shows up (65535) as the qos-group.

Conditions: Occurs when configuring a Python class-map to match on qos-group 1022 and then submitting the class-map.

Workaround: Use qos-group \<1 - 127\> for matching.

### **CSCum07556**

Symptoms: When an IPv6 prefix is configured with IPv6 recursive next hop, it can lead to a crash.

Conditions: Occurs when an IPv6 prefix is configured with IPv6 recursive next hop.

Workaround: Use V6mappedV4 as a next hop address or do not use recursive IPv6 next hop when configuring IPv6 application route.

### **CSCum08139**

Symptoms: Odd behavior occurs when multiple applications are run against multiple devices, such as applications not getting packets when they expect to.

Conditions: Occurs when multiple applications attempt to register for packets via the Policy service set across multiple devices.

Workaround: A limited number of scenarios are validated to work in the current release. The valid scenarios are:

- 1 Multiple applications on the same Linux host registering with the same network element on different targets (limitation enforced via Policy Service Set).
- 2 Multiple Linux hosts, where each host runs multiple applications conforming to (1).
- 3 Single application registering for packets on multiple targets across multiple network elements.
- 4 Multiple applications, each running on a separate Linux host, where each application conforms to (3).

No other scenarios (e.g multiple applications on a single host registering for packets across multiple network elements) is currently validated to function correctly.

#### **CSCum09504**

Symptoms: Using the onePK Python SDK, calling `onep.aaa.User.User.send_accounting_record(self, action, acctRecord)` results in the disconnection of the onep session and the error message "onep.core.event.EventManager:Event channel listener error".

Conditions: Occurs when the `acctRecord` argument of `onep.aaa.User.User.send_accounting_record(self, action, acctRecord)` is a list containing one or more instances of `onep.aaa.IntAttribute.IntAttribute` or `onep.aaa.BinaryAttribute.BinaryAttribute`.

Workaround: There is no workaround.

#### **CSCum10040**

Symptoms: Output of CPU utilization shows garbage characters.

Conditions: Occurs when running `ElementProcessTutorial.py`.

Workaround: There is no workaround.

© 2013 Cisco Systems, Inc. All rights reserved.