Appendix B

CCNA ICND2 200-105 Exam Updates

Over time, reader feedback allows Pearson to gauge which topics give our readers the most problems when taking the exams. To assist readers with those topics, the authors create new materials clarifying and expanding on those troublesome exam topics. As mentioned in the Introduction, the additional content about the exam is contained in a PDF document with this book's updates at http://www.ciscopress.com/title/9781587205798.

This appendix is intended to provide you with updated information if Cisco makes minor modifications to the exam upon which this book is based. When Cisco releases an entirely new exam, the changes are usually too extensive to provide in a simple update appendix. In those cases, you might need to consult the new edition of the book for the updated content.

This appendix attempts to fill the void that occurs with any print book. In particular, this appendix does the following:

- Mentions technical items that might not have been mentioned elsewhere in the book
- Covers new topics if Cisco adds new content to the exam over time
- Provides a way to get up-to-the-minute current information about content for the exam

Always Get the Latest at the Book's Product Page

You are reading the version of this appendix that was available when your book was printed. However, given that the main purpose of this appendix is to be a living, changing document, it is important that you look for the latest version online at the book's companion website. To do so, follow these steps:

Step 1. Browse to http://www.ciscopress.com/title/9781587205798.

Step 2. Click the **Updates** tab.

Step 3. If there is a new Appendix B document on the page, download the latest Appendix B document.

NOTE The downloaded document has a version number. Comparing the version of the print Appendix B with the latest online version of this appendix, you should do the following:

- **Same version:** Ignore the PDF that you downloaded from the companion website.
- **Website has a later version:** Ignore this Appendix B in your book and read only the latest version that you downloaded from the companion website.

Technical Content

The version for this appendix is version 2.0. The version history is as follows:

Version 1.0: No technical content; the appendix was a placeholder to inform readers to check the website for future updates.

Version 2.0: Added the short planned section about the APIC-EM Path Trace ACL Analysis tool, which had not been released when the book originally published.

Table B-1 lists the major headings in this appendix and the chapter after which to read the content contained in this appendix.

Table B-1 Topics and When to Best Read Them

Topic	Chapter
The APIC-EM Path Trace ACL Analysis Tool	28

The APIC-EM Path Trace ACL Analysis Tool

Chapter 28 of this book mentions one particular exam topic that talks about a particular feature of the APIC-EM controller. That feature was not yet available in APIC-EM by the time the book went to print. This section of Appendix B completes the material about that exam topic by describing that one product feature, a feature described in the exam topic as "APIC-EM Path Trace ACL Analysis tool." To that end, this section first describes the APIC-EM Path Trace tool, and then the specific ACL Analysis tool that is integrated as part of the Path Trace application.

APIC-EM Functions and Applications

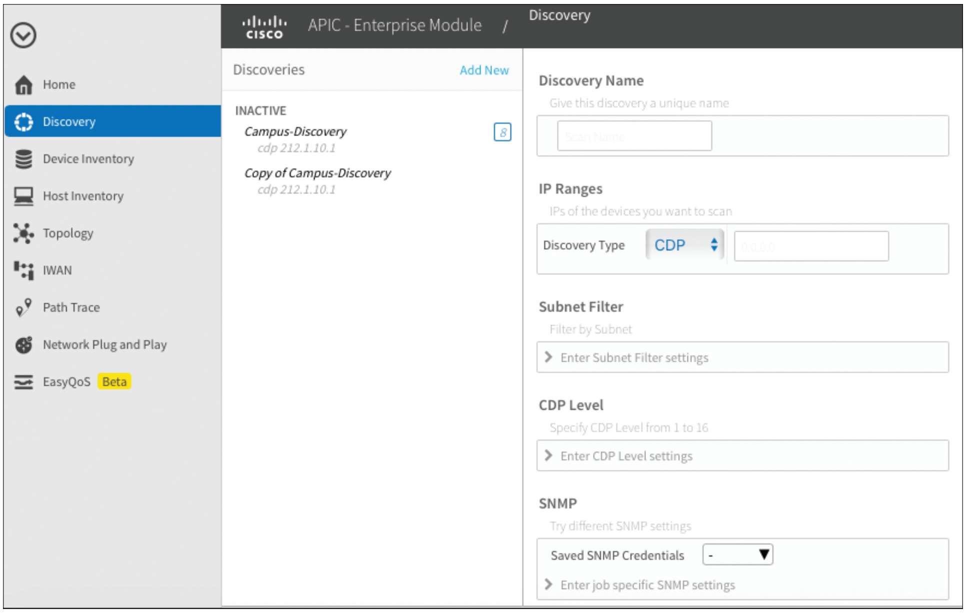
As discussed in Chapter 28, the Application Policy Infrastructure Controller Enterprise Module (APIC-EM) centralizes the control and programmability of the enterprise WAN. APIC-EM allows for centralized control of the enterprise networking devices through northbound APIs and through applications that run as part of APIC-EM itself. Although APIC-EM today does not remove traditional control plane functions out of the networking devices—protocols such as routing protocols and spanning tree—APIC-EM does allow for more centralized operation and programmability of the network.

When this appendix was written in June 2016, Cisco listed four available APIC-EM applications in addition to the core features of the APIC-EM product. To the end user, the core features and applications all appear as another option from the user interface. However, the separately identified applications are

- Enterprise Service Automation (ESA)
- Intelligent WAN (IWAN)
- Plug and Play (PnP)
- Path Trace

Figure B-1 shows an image of the APIC-EM user interface for perspective. The list on the left shows the built-in functions like discovery, device inventory, host inventory, and topology, with the list of applications below. In this case, the Discovery option is selected, filling the right side of the screen with details of how you could go about telling APIC-EM to discover the devices in the network.

NOTE Cisco states that APIC-EM works on regular update cycles, with new releases expected every three months. This material is based on APIC-EM version 1.2, released around June 1, 2016. The features and functions of APIC-EM will, of course, develop further over time.



INACTIVE

Campus-Discovery
cdp 212.1.10.1

Copy of Campus-Discovery
cdp 212.1.10.1

Discovery Name
Give this discovery a unique name
Scan Name

IP Ranges
IPs of the devices you want to scan
Discovery Type: CDP 0.0.0.0

Subnet Filter
Filter by Subnet
> Enter Subnet Filter settings

CDP Level
Specify CDP Level from 1 to 16
> Enter CDP Level settings

SNMP
Try different SNMP settings
Saved SNMP Credentials: -
> Enter job specific SNMP settings

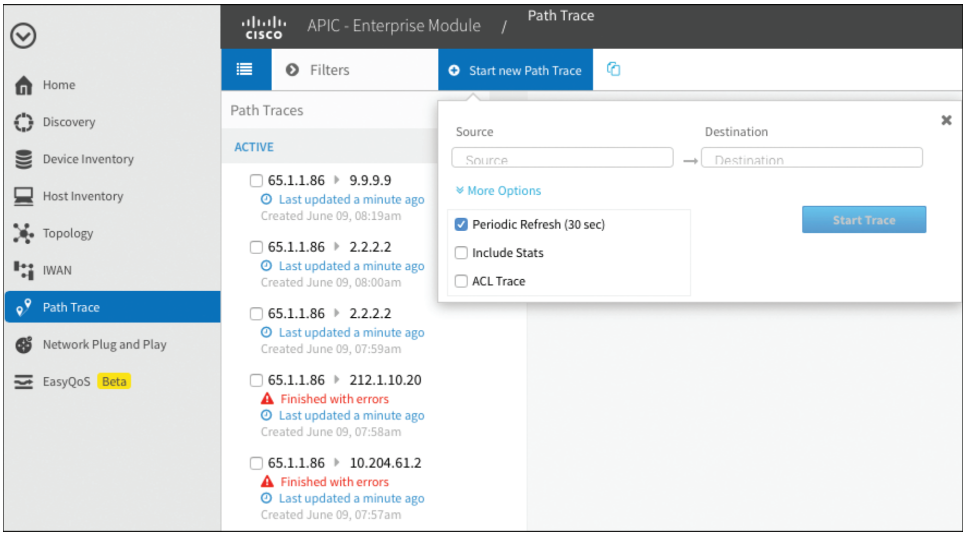
Figure B-1 *APIC-EM GUI with Four Applications Shown*

APIC-EM Path Trace Application

The APIC-EM Path Trace app provides a wonderful function for the operation and troubleshooting of networks. Path Trace takes as input from the user a source and destination IP address. Path Trace then analyzes the current forwarding tables in the devices in the network, determines where packets would flow between those two addresses, and shows that path on a network topology map.

Figure B-2 shows an example of the Path Trace app GUI in which you can enter the source and destination IP address. Of particular note:

- The user interface lists prior attempts on the left.
- The window at the top lets you enter the source and destination IP address (with a drop-down list of known addresses, not shown in the figure).
- Note in the small window that the ACL Trace box is not checked.



□ 65.1.1.86 ▸ 9.9.9.9
Last updated a minute ago
Created June 09, 08:19am

□ 65.1.1.86 ▸ 2.2.2.2
Last updated a minute ago
Created June 09, 08:00am

□ 65.1.1.86 ▸ 2.2.2.2
Last updated a minute ago
Created June 09, 07:59am

□ 65.1.1.86 ▸ 212.1.10.20
Finished with errors
Last updated a minute ago
Created June 09, 07:58am

□ 65.1.1.86 ▸ 10.204.61.2
Finished with errors
Last updated a minute ago
Created June 09, 07:57am

☑ Periodic Refresh (30 sec)
□ Include Stats
□ ACL Trace

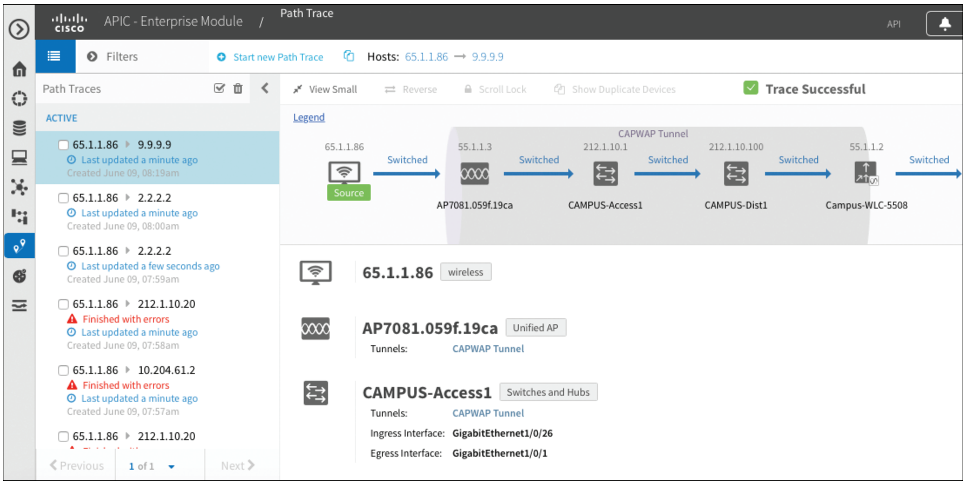
Figure B-2 *Path Trace GUI with Start New Path Trace Window Open*

The GUI input process is pretty basic. You set the source and destination address by typing or choosing from the list. You can choose a few other options, such as enabling or disabling a periodic refresh of the trace results. You also can supply the transport layer protocol and ports (by clicking the More Options button), which supplies information Path Trace needs to find the path used by some load balancing logic.

After you click the Start Trace button, the Path Trace app looks at the forwarding tables and then draws a window with the following:

- The devices in the path
- Notations about the forwarding logic used to forward over each link (for example, switched for a Layer 2 switching decision and routed for a Layer 3 routing decision)
- Going down the page, a list of the devices in the path

Figure B-3 shows a sample Path Trace result.



Hosts: 65.1.1.86 → 9.9.9.9

65.1.1.86 ▸ 9.9.9.9
Last updated a minute ago
Created June 09, 08:19am

65.1.1.86 ▸ 2.2.2.2
Last updated a minute ago
Created June 09, 08:00am

65.1.1.86 ▸ 2.2.2.2
Last updated a few seconds ago
Created June 09, 07:59am

65.1.1.86 ▸ 212.1.10.20
Finished with errors
Last updated a minute ago
Created June 09, 07:58am

65.1.1.86 ▸ 10.204.61.2
Finished with errors
Last updated a minute ago
Created June 09, 07:57am

65.1.1.86 ▸ 212.1.10.20

65.1.1.86 wireless

AP7081.059f.19ca Unified AP
Tunnels: CAPWAP Tunnel

CAMPUS-Access1 Switches and Hubs
Tunnels: CAPWAP Tunnel
Ingress Interface: GigabitEthernet1/0/26
Egress Interface: GigabitEthernet1/0/1

Figure B-3 *Sample Results from an APIC-EM Path Trace*

NOTE Figure B-3 and the rest of the figures in this section have collapsed the menu on the left; note that it shows the Path Trace icon in a darker color, which means that the user had first selected the Path Trace app.

This particular screenshot shows a path trace with five consecutive Layer 2 forwarding actions at the top of the window. Path Trace provides a horizontal view of the network path, with arrowed lines pointing in the direction of flow. In this case, the word Switched over each link refers to the fact that each uses a Layer 2 switching choice to forward the message.

Additionally, notice the gray rectangle with the words CAPWAP Tunnel at the top. The host on the far left is a wireless host whose traffic flows to the wireless LAN controller (WLC) with hostname Campus-WLC-5508. That traffic is encapsulated in a CAPWAP tunnel from the ingress AP to the WLC, as noted with that gray section.

The APIC-EM Path Trace application can identify a variety of forwarding actions by different devices, as referenced as a *device protocol*. Some of these device protocols sit within the scope of what you have learned for CCNA R&S, and some do not. However, to give you a sense of the kinds of device protocols APIC-EM's Path Trace application might list, Table B-2 lists some of the protocols you have read about in this book.

Table B-2 APIC-EM Path Trace Device Protocols

Indicator	Meaning
HSRP	When HSRP is used, Path Trace determines the active router and then shows the path going through that router (rather than through the standby router).
OSPF, EIGRP, BGP, static, connected	When a device performs routing, APIC-EM checks the source of the route that was used and lists that routing source. For example, if the route used to forward a packet is learned with OSPF, Path Trace lists "OSPF."
SVI	When a Layer 3 switch forwards out an SVI interface, Path Trace can list the device protocol as SVI.
Switched	When a Layer 2 switch makes a Layer 2 forwarding decision.

APIC-EM Path Trace ACL Analysis Feature (ACL Trace)

This topic exists in the book because of one specific exam topic on the ICND2 and CCNA R&S exams. Quoting from the ICND2 200-105 exam topics:

> 4.5 Verify ACLs using the APIC-EM *Path Trace ACL Analysis tool*

Because of the timing and other factors, the exam topic uses a different phrasing than some other references, such as the APIC-EM product documentation. The exam topic uses the phrase *ACL Analysis tool* and the longer *Path Trace ACL Analysis tool*. The APIC-EM user interface for version 1.2 (which is the first version to include the feature) uses the term *ACL Trace* as noted in the check box shown previously in Figure B-2. The APIC-EM configuration guide adds yet another phrase, *ACL-Based Path Trace*, and with some references to the shorter *ACL Trace*. So be ready to be flexible in your reading about this feature.

This appendix uses *ACL Trace* for the rest of the discussion.

ACL Trace is not a separate APIC-EM app but rather a feature of the Path Trace application. Because of that, to navigate to the ACL Trace feature, an APIC-EM user must first choose the Path Trace icon/option on the left side of the APIC-EM user interface.

So, terminology aside, what does ACL Trace do? It takes the path determined by the Path Trace app and analyzes any ACLs in that path. That analysis compares the packet you described in the Path Trace input window with the ACLs in the path. So, the Path Trace tool determines the path, and the ACL Trace feature determines whether any ACLs in that path would filter your packet.

The rest of this section works through the logic details of ACL Trace along with some examples.

ACL Trace and Matching on Source/Destination Address Only

Ignoring the user interface for a moment, imagine that you used Path Trace with the ACL Trace option. You created a new Path Trace with a source IP address of 10.1.1.1 and a destination of 10.2.2.2, and checked the box for ACL Trace. That is basically all you have to do from the user interface to perform an ACL Trace. Then you start the trace, and it completes. What kind of information do you think would be provided by a tool called ACL Trace? Well, here's what you get:

Path Trace info: The same figure of the network normally given by Path Trace, including the reasons why packets were forwarded over a particular link, with a list of devices below.

New ACL Trace info: Overlaid on that same diagram you get icons (ACL Trace indicators) sitting on each device. These tell you whether an ACL would filter your packet. It also supplies icons that enable you to drill down to see more information.

In other words, as a feature of Path Trace, ACL Trace just adds more info to the same kind of output Path Trace already supplies. Figure B-4 shows a sample close-up from the Path Trace results page, with just the part of the page with the horizontal sequence of devices. Note that the figure shows the usual Path Trace info of reasons why the message was forwarded—in this case, a Layer 2 switched link, then a route learned by OSPF, and then a connected route. Additionally, the figure shows check boxes over two devices and a red box with a X in it over a router.

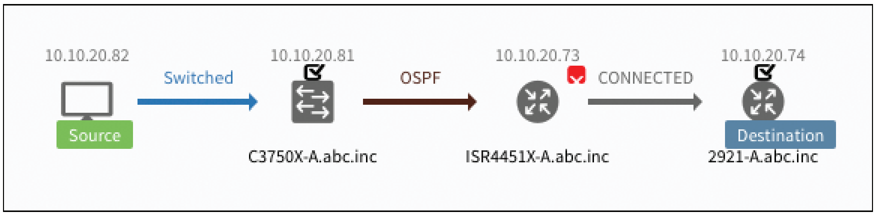


Figure B-4 *Path Trace Indicators: No ACL and ACL Denies Packet*

The image shows two examples of ACL Trace indicators; that is, icons that tell you the big idea that ACL Trace has determined for each node. These indicators are icons that describe four answers about what ACL Trace expects ACLs in the network to do to that packet. Table B-3 covers the meaning of the indicators.

Table B-3 APIC-EM ACL Trace Indicators (Icons)

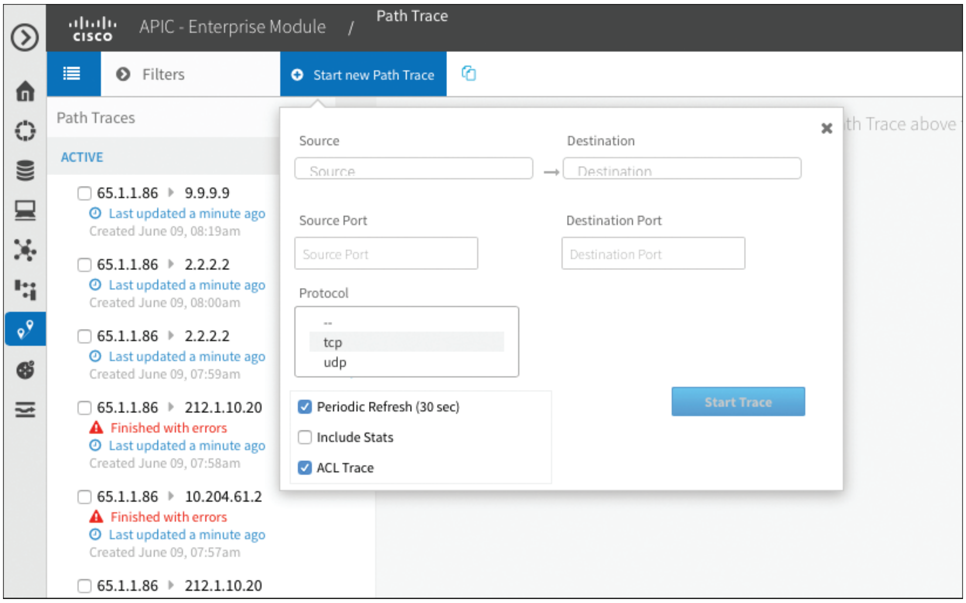
Indicator Icon	Indicator Description	Meaning
☑	Check box, no fill color	No ACL is present
✅	Check box, green fill	An ACL is present and permits the packet
☒	Box with X and red fill	An ACL is present and it does deny the packet
⚠	Triangle with ! and yellow fill	An ACL is present and might or might not be denying the packet

The text now walks through two examples of using an ACL Trace. The first results in a definitive answer, so the ACL Trace indicator shows a red box with an X in it, with some devices noted as having no ACL configured at all. The second example demonstrates a case in which ACL Trace is unsure, showing the yellow triangle with an exclamation point to note that there is ambiguity. The good news is that in both cases, more detail is easily available to show you exactly why ACL Trace made its choice.

ACL Trace Example with Transport Protocol and Port Numbers

The Path Trace application user interface at which the user creates a new path trace event lists a check box with the words ACL Trace beside it, as shown in Figure B-5. Checking the ACL Trace box tells Path Trace to also perform the ACL Trace function. Simple enough. (Note that to navigate to this page, the user would click the path trace icon on the menu on the left, and then click Start New Path Trace at the top of the window to make the window shown in the figure appear.)

Figure B-5 also shows the fields that appear if the user clicks a button labelled More Options. Those additional options are the Source Port, Destination Port, and Protocol fields. The figure shows the Protocols field with its drop-down menu listed, with two protocols, UDP and TCP.



□ 65.1.1.86 ▸ 9.9.9.9
Last updated a minute ago
Created June 09, 08:19am

□ 65.1.1.86 ▸ 2.2.2.2
Last updated a minute ago
Created June 09, 08:00am

□ 65.1.1.86 ▸ 2.2.2.2
Last updated a minute ago
Created June 09, 07:59am

□ 65.1.1.86 ▸ 212.1.10.20
Finished with errors
Last updated a minute ago
Created June 09, 07:58am

□ 65.1.1.86 ▸ 10.204.61.2
Finished with errors
Last updated a minute ago
Created June 09, 07:57am

□ 65.1.1.86 ▸ 212.1.10.20

Source: Source → Destination: Destination
Source Port: Source Port
Destination Port: Destination Port
Protocol: -- / tcp / udp

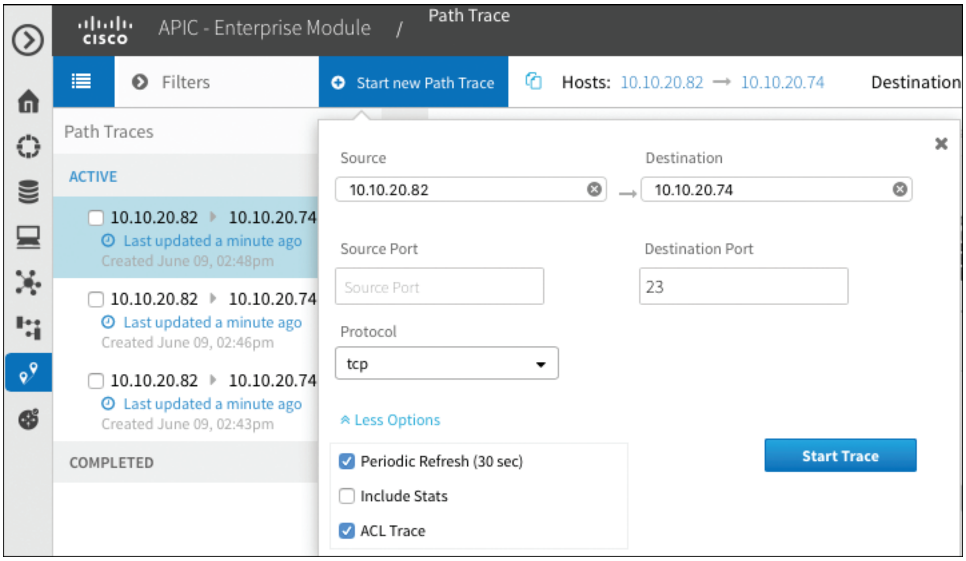
☑ Periodic Refresh (30 sec)
□ Include Stats
☑ ACL Trace

Figure B-5 *Input of Parameters for a Sample Path Trace ACL Trace Attempt*

So what do these extra options mean for Path Trace and ACL Trace? Most forwarding logic ignores the transport protocol and ports. However, many ACLs examine the transport protocol and ports, so the ACL Trace feature can make good use of the information. In fact, by using these fields, you could test ideas like this:

- Packets from address X to address Y's well-known service that uses TCP port 80 (HTTP)
- Packets from address X to address Y's well-known service that uses TCP port 23 (Telnet)
- Packets from address X's well-known service that uses TCP port 80 (HTTP) to address Y

This next example uses the logic shown in the second item in the list, as shown in Figure B-6. It basically adds the details into the Path Trace window to path that item, with source address 10.10.20.82 and destination address 10.10.20.74.



Hosts: 10.10.20.82 → 10.10.20.74 Destination

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 02:48pm

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 02:46pm

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 02:43pm

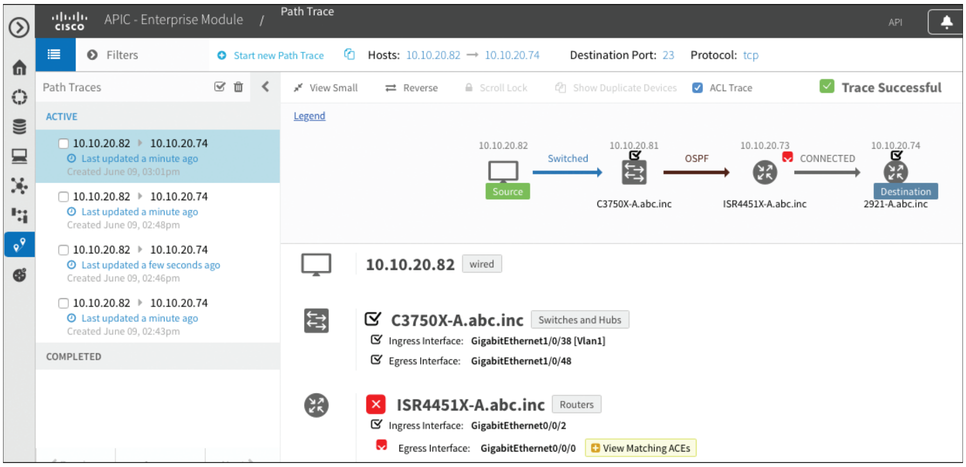
Source: 10.10.20.82 → Destination: 10.10.20.74
Source Port: Source Port
Destination Port: 23
Protocol: tcp

☑ Periodic Refresh (30 sec)
□ Include Stats
☑ ACL Trace

Figure B-6 *Input into the Path Trace Window to Create an ACL Trace with TCP Port 23*

After you add the input in Figure B-6 and click the Start Trace button, the Path Trace app does the work. It uses the usual southbound APIs to discover forwarding table information, as well as ACL configuration, to perform both the Path Trace and the ACL Trace features. Figure B-7 shows the results of this particular attempt.

NOTE Figure B-4 is a close-up view of the image in Figure B-7.



Hosts: 10.10.20.82 → 10.10.20.74   Destination Port: 23   Protocol: tcp

☑ ACL Trace   ☑ Trace Successful

ACTIVE

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 03:01pm

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 02:48pm

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a few seconds ago
Created June 09, 02:46pm

□ 10.10.20.82 ▸ 10.10.20.74
Last updated a minute ago
Created June 09, 02:43pm

COMPLETED

10.10.20.82 wired

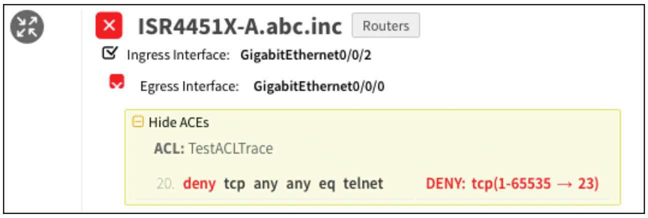
☑ C3750X-A.abc.inc Switches and Hubs
☑ Ingress Interface: GigabitEthernet1/0/38 [Vlan1]
☑ Egress Interface: GigabitEthernet1/0/48

☒ ISR4451X-A.abc.inc Routers
☑ Ingress Interface: GigabitEthernet0/0/2
☒ Egress Interface: GigabitEthernet0/0/0 View Matching ACEs

Figure B-7 *Path Trace ACL Trace Results*

To interpret Figure B-7, first look at the horizontal network map at the top. Of those four device icons, the left-most icon, representing a host, has no ACL Trace indicator. However, the other three icons, which represent networking devices, have ACL Trace indicators. The second and fourth icons have the box with a check mark, indicating that ACL Trace found no ACLs on those devices. However, the third device reading left to right has an ACL Trace indicator of a box with an X in it, telling us that the ACL would filter packets.

Moving down in Figure B-7, note the same ACL Trace indicator icon next to the name of the router that would filter the packet (hostname ISR4451X-A). That section also lists a clickable item labelled View Matching ACEs. ACEs, or Access Control Entries, are the individual lines in an ACL. Figure B-8 shows a close-up view of what the app displays next when you click that View Matching ACEs button. It shows the specific ACL (named TestACLTrace) and the specific ACE (**deny tcp any any eq telnet**) that would match the packet and, in this case, block the packet. (Note that the icon and the mentions of the word **deny** show up in red when viewing this output in color.)



☒ ISR4451X-A.abc.inc Routers
☑ Ingress Interface: GigabitEthernet0/0/2
☒ Egress Interface: GigabitEthernet0/0/0

⊟ Hide ACEs
ACL: TestACLTrace
20. deny tcp any any eq telnet   DENY: tcp(1-65535 → 23)

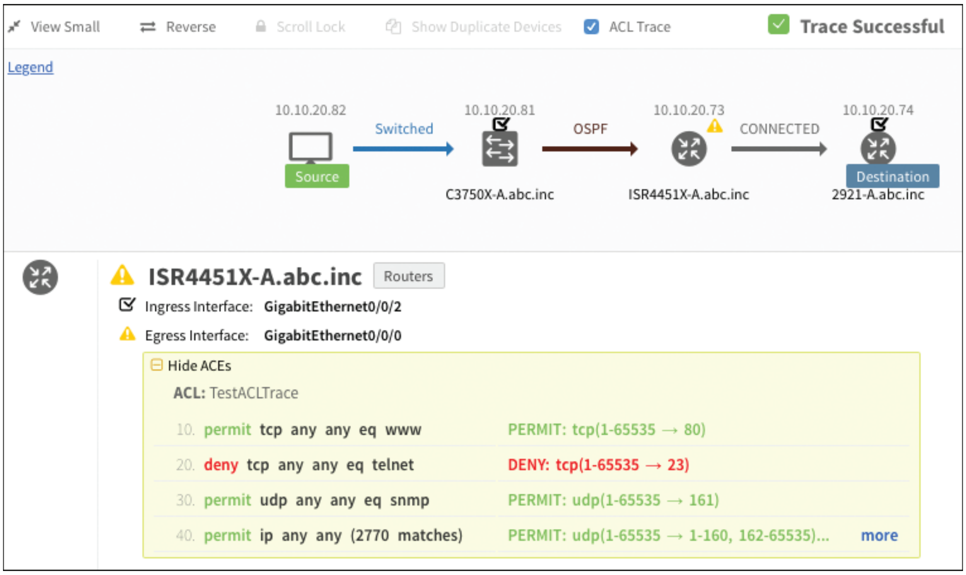
Figure B-8 *Close-up of Single Router After Clicking View Matching ACLs*

ACL Trace Example with No Transport Layer Details

In the previous example, the ACL Trace feature gave a definite answer: The ACL on one router would filter the packet. ACL Trace might not be able to make such a confident answer, even if you supply the transport layer details as done in the previous example. However, because so many ACLs match on transport layer port numbers, if you choose to leave out transport layer details when using ACL Trace, ACL Trace might not be able to tell whether the packet would be blocked. When that happens, ACL Trace tells you just that: the packet might be filtered but might not be filtered. This next example shows just such a case.

This next example repeats the previous example, except that instead of the user adding a transport protocol of TCP and a destination port of port 23, the user leaves those fields blank. In other words, this next example is like the input shown in Figure B-6, but with the transport protocol and destination port fields blank.

Figure B-9 shows the results of this next ACL Trace, with that same ACL located on Router ISR4451X-A. That ACL matches packets based on several transport port numbers: TCP port 80 (HTTP), TCP port 23 (Telnet), and UDP port 161 (SNMP). However, ACL Trace cannot tell which ACE would be matched, because all three of those ACEs match the source and destination IP addresses supplied to ACL Trace. So, as you see in both the top and center of the figure, the Path Trace app lists the little triangle with an exclamation point in it. (It's yellow in the color user interface.)



ISR4451X-A.abc.inc Routers

Ingress Interface: GigabitEthernet0/0/2

Egress Interface: GigabitEthernet0/0/0

ACL: TestACLTrace

```
10. permit tcp any any eq www        PERMIT: tcp(1-65535 → 80)
20. deny tcp any any eq telnet       DENY: tcp(1-65535 → 23)
30. permit udp any any eq snmp       PERMIT: udp(1-65535 → 161)
40. permit ip any any (2770 matches) PERMIT: udp(1-65535 → 1-160, 162-65535)...
```

Figure B-9 *Results That Show ACL Might or Might Not Be Blocked*

Although you might think that an answer of "it may be blocked, but maybe not" might not be very useful, the beauty of this feature comes through in the bottom half of the window. Figure B-9 shows the output with the View Matching ACEs button already clicked. Note that ACL Trace lists all the ACEs that could match a packet per the criteria you added in the Path Trace window, which in this case is four different ACEs. (In the color GUI, it lists the permit items in green and the deny items in red.) Then you can make your own determination about which types of packets will be permitted and denied by this ACL.