paloalto NETWORKS | CYBERSECURITY PARTNER OF CHOICE

Palo Alto Networks Firewall: Troubleshooting

Student Guide

EDU-330

PAN-OS® 11.1

Courseware Version A

June 2024

Palo Alto Networks Technical Education

Palo Alto Networks, Inc.

https://www.paloaltonetworks.com

Table of Contents

Typographical Conventions 11
How to Use This Lab Guide 12
Browsers 12
1.1 Lab: Tech Support Files 14
Lab Objectives 14
Lab Scenario 14
1.1.1 Connect to Your Student Firewall 14
1.1.2 Apply a Baseline Configuration to the Firewall 14
1.1.3 Validate the Basic Functionality of the System 18
1.1.4 Use the Web Interface to Get a Tech Support File 20
1.1.5 Decompress the Contents of the Tech Support File 21
1.1.6 Explore the Tech Support File 23
1.1.7 Clean Up Your Lab Environment 26
1.2 Lab: Use the CLI to Export a Tech Support File 27
Lab Objectives 27
Lab Scenario 27
1.2.1 Use the CLI to Generate a Tech Support File 27
1.2.2 Use the CLI to Export a Tech Support File 28
1.2.3 Validate the Exported Tech Support File 30
1.2.4 Reference Information 31
1.2.5 Clean Up Your Lab Environment 31
1.3 Lab: CLI Fundamentals 32
Lab Objectives 32
Lab Scenario 32
1.3.1 Import, Load, and Commit a Configuration File 32
1.3.2 Confirm the Current Device Configuration 43
1.3.3 Explore Options for Changing Other Device Settings 44
1.3.4 Change the Current Device Configuration 47
1.3.5 Clean Up Your Lab Environment 50

1.4. Lab (Optional): Use the CLI to Modify Policy Objects 51
Lab Objectives 51
Lab Scenario 51
1.4.1 Review the Existing Policy Configuration 52
1.4.2 Use the CLI to Examine a Configuration and Discover Options for How to Modify It 55
1.4.3 Modify Object Parameters 62
1.4.4 Review Changes and Commit the Configuration 66
1.4.5 (Optional) Test URL Filtering Profile Changes 70
1.4.6 Reference Information 71
1.4.7 Clean Up Your Lab Environment 72
2. Lab: Tracing Data-Plane Packet Flow 73
Lab Objectives 73
Lab Scenario 73
2.1 Open the Packet-Diagnostics File 75
2.2 Trace the First Packet Through the Firewall 76
2.3 Trace the Second Packet 81
2.4 Trace the Content Inspection of a Packet 81
2.5 Identify Firewall-Generated Packets 83
2.6 Identify Other Dropped Packets and the Session End 85
2.7 Clean Up Your Lab Environment 86
2.8 Reference Information 87
3. Lab: Packet Capture 88
Lab Objectives 88
Lab Scenario 88
3.1 Load a Configuration and Test Baseline Functionality 88
3.2 Configure a Packet Filter 89
3.3 Test Session Marking 90
3.4 Configure Capture Stages 91
3.5 Clear Marked Sessions 94
3.6 Turn On Packet Capture and Capture Packets 94
3.7 Analyze the Pcaps 97

3.8 Add a Security Policy Configuration to Drop Traffic 98
3.9 Reconfigure the Filter 99
3.10 Capture a New Session and Download the Pcaps 101
3.11 Analyze the Pcaps 103
3.12 Clean Up Your Lab Environment 107
3.13 Reference Information 107
4. Lab: Flow Basic 109
Lab Objectives 109
Lab Scenario 109
4.1 Load the Lab Config File and Start the FTP Server 109
4.2 Verify External Connectivity to the FTP Server 109
4.3 Verify the Problem with the Internal Client 110
4.4 Examine Firewall Traffic Logs and Threat Logs 112
4.5 Configure the Capture Filter 113
4.6 Check Counters 115
4.7 Configure Packet Capture and Enable Flow Basic 119
4.8 Run Packet Capture and Flow Basic Diagnostic Logging 122
4.9 Interpret the Flow Basic Log and Pcaps 124
4.10 Implement a Solution and Verify 128
4.11 Check Logs and Enable Logging for Increased Visibility 129
4.12 Clean Up Your Lab Environment 132
5.1 Lab: Host-Inbound VPN Traffic—Case A 134
Lab Objectives 134
Lab Scenario 134
5.1.1 Apply a Baseline Configuration to the Firewall 135
5.1.2 Verify the Problem 135
5.1.3 Check Routing and Security Policy Rules 136
5.1.4 Stop! Try a Top-Down Approach Instead 138
5.1.5 Check the Health of the VPN Tunnel 140
5.1.6 Initiate the VPN Connection from the Remote Network 142
5.1.7 Troubleshoot the VPN Connection as the Responder 143

5.1.8 Check Proxy ID Settings and Correct the Problem 144
5.1.9 Verify the Solution 145
5.1.10 Clean Up Your Lab Environment 146
5.2 Lab: Host-Inbound VPN Traffic—Case B 147
Lab Objectives 147
Lab Scenario 147
5.2.1 Apply a Baseline Configuration to the Firewall 147
5.2.2 Verify the Problem with SFTP Access to the Web Server 147
5.2.3 Review the Traffic and System Logs 148
5.2.4 Check the High-Level Health Indicators for the Tunnel 150
5.2.5 Troubleshoot as the Responder 150
5.2.6 Reset the Pre-Shared Key and Verify Functionality 151
5.2.7 (Optional) Cause the Firewall to Initiate the Connection 153
5.2.8 Clean Up Your Lab Environment 156
5.3 (Optional) Troubleshoot VPN connectivity independently – Case C 157
Lab Objectives 157
Lab Scenario 157
5.3.1 Apply a Baseline Configuration to the Firewall 157
5.3.2 Troubleshoot and Resolve 158
6.1 Lab: Transit Traffic—App-ID and Torrents 159
Lab Objectives 159
Lab Scenario 159
6.1.1 Apply a Baseline Configuration to the Firewall 159
6.1.2 Attempt to Download Torrent File 160
6.1.3 Examine Traffic Logs and App-ID Results 163
6.1.4 Enable Traffic 165
6.1.5 Set the Matching Policy Rule to “Deny” and Test 167
6.1.6 Create a Policy Rule to Block Torrents 168
6.1.7 Add File Blocking to the Security Profile Setting 169
6.1.8 Clean Up Your Lab Environment 170
6.2 Lab: Transit Traffic—Blocking Tor 172

Lab Objectives 172
Lab Scenario 172
6.2.1 Lab Challenge and Checklist 172
6.2.2 Lab Solution: Security Policy to Block Tor App-ID 174
6.2.3 Lab Solution: Use Application Filters 175
6.2.4 Lab Solution: Block Risky URL Categories 177
6.2.5 Lab Solution: Deny Unknown Applications 177
6.2.6 Lab Solution: Blocking Untrusted and Expired Certificates with a Decryption Profile 178
6.2.7 Lab Solution: Create Decryption Profile for Decrypted Traffic 179
6.2.8 Lab Solution: Use an External Dynamic List (EDL) 180
6.2.9 Clean Up Your Lab Environment 181
6.3 (Optional) Troubleshoot Internet connectivity 182
Lab Objectives 182
Lab Scenario 182
6.3.1 Check Running Services 182
6.3.2 Troubleshoot and Resolve 182
7. Lab: System Services 183
Lab Objectives 183
Lab Scenario 183
7.1 Check Running Services 183
7.2 Review the Logs for a Specific Service 185
7.3 Change the Debug Level for a Service 187
7.4 Restart a Service 191
7.5 Restart a Service and Monitor a Data-Plane Session 192
7.6 Investigate the Event 194
7.7 Clean Up Your Lab Environment 196
8. Lab: SSL Decryption 197
Lab Objectives 197
Lab Scenario 197
8.1 Apply a Baseline Configuration to the Firewall 197
8.2 Verify the Functionality of SSL Decryption 198

8.3 Create a Tag and a Dynamic Address Group 201
8.4 Create a Decryption Policy Rule 204
8.5 Create Custom Vulnerability Signatures 207
8.6 Configure a Log Forwarding Profile 214
8.7 Configure a Vulnerability Protection Profile to Generate Alerts 220
8.8 Add the Log Forwarding Profile to a Security Policy Rule 225
8.9 Test the Configuration and Confirm Results 226
8.10 Clean Up Your Lab Environment 234
9. Lab: No User-ID Names in Logs 235
Lab Objectives 235
Lab Scenario 235
9.1 Apply a Baseline Configuration to the Firewall 235
9.2 Diagnose and Fix the Problem 236
9.3 Reference Information 236
9.4 Lab Solution: Enable User-ID on the Correct Zone 238
9.5 Lab Solution: Fix the LDAP Server Profile 239
9.6 Lab Solution: Verify the Solution with Traffic Logs 240
9.7 Clean Up Your Lab Environment 241
10. Lab: Troubleshooting GlobalProtect 242
Lab Objectives 242
Lab Scenario 242
10.1 Apply a Baseline Configuration to the Firewall 242
10.2 Install the GlobalProtect Agent 243
10.3 Connect to the External Gateway 244
10.4 Export GlobalProtect Certificate 245
10.5 Disconnect the Connected User 255
Locate Information about the Client 255
Bonus Lab 257
Lab Objectives 257
Detailed Lab Steps 257
Apply a Baseline configuration to the Firewall 257

Modify Authentication Settings..... 258
Save the Configuration..... 261
Commit Your Changes and Verify Fix..... 262

Typographical Conventions

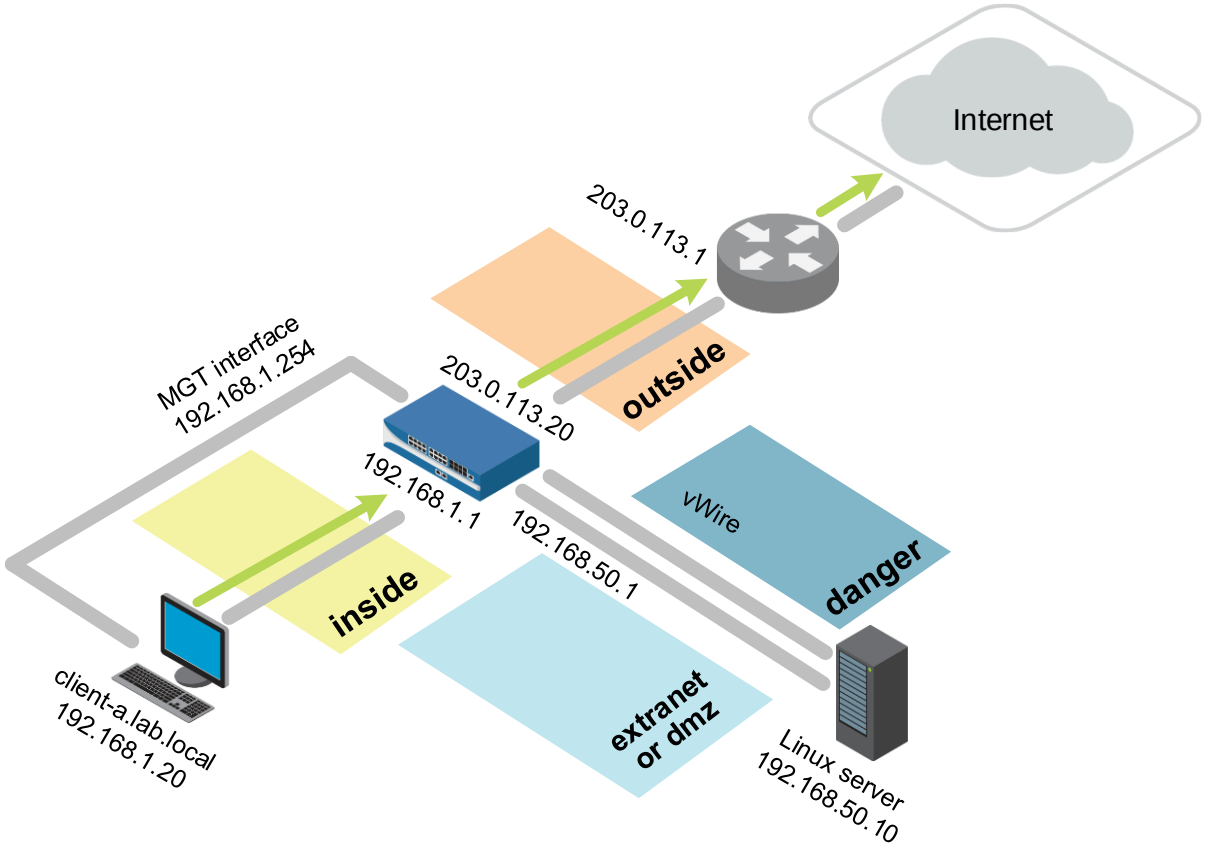
This guide uses the following typographical conventions for special terms and instructions.

Convention	Meaning	Example
Bolding	Names of selectable items in the web interface	Click **Security** to open the **Security Policy Rule** window.
`Consolas` font	Text that you enter and coding examples	Enter the following command: **`a:\setup`** The **`show arp all`** command yields this output: **`username@hostname> show arp <output>`**
Calibri 11 pt. gray font	Lab step results and explanations	A new zone should appear in the web interface.
Click	Click the left mouse button	Click **Administrators** under the **Device** tab
Right-click	Click the right mouse button	Right-click the number of a rule you want to copy, and select **Clone Rule**
< > (text enclosed in angle brackets)	Denotes a variable parameter. Actual value to use is defined in the Lab Guide document.	Type **`ping source <value> host <value>`** and press **Enter**.

How to Use This Lab Guide

The Lab Guide contains exercises that correspond to modules in the Student Guide. Some lab exercises include step-by-step, task-oriented instructions that you should follow precisely to complete the exercise and be able to answer various questions about the actions prescribed. Some lab exercises include sections that provide only a description of a problem or a situation and ask you to solve the problem or address the situation by use of the methodologies, skills, and procedures that you must apply and, if necessary, acquire without step-by-step guidance.

The following diagram provides a basic overview of the lab environment:



Browsers

You will use two different browsers for these lab exercises:

- Configuration Browser - use this application to configure the firewall.
- Testing Browser - use this application to test features once you have configured the firewall.

There are two browsers available in the lab environment:

- Chromium

- Firefox

Note: For all lab exercises, we recommend always using the Chromium browser as the configuration browser when accessing the FireWall WebUI and Firefox as the testing browser. Chromium has been shown to produce fewer errors than other browsers like Firefox when navigating the FireWall WebUI. Please note that the FireWall WebUI requires a lot of memory, and having more than three browser windows or tabs open at the same time can consume the client's entire memory and consequently slow down the lab. We also recommend you restart your browser at least once a day.

The detailed lab guide sections will let you know which browser to use for each task.

1.1 Lab: Tech Support Files

Lab Objectives

- Load a configuration and validate the basic functionality of your lab environment
- Use the web interface to generate and download a Tech Support File
- Review the structure and contents of Tech Support Files

Lab Scenario

The Tech Support File is a compressed archive of system logs:

- You need to generate a Tech Support File to open a tech support case.
- You want to know what is inside the Tech Support File.

Subsequent modules and lab activities will provide more detailed information about many of these files. For now, your goal simply should be to develop a familiarity with the names of various files that you might see when you troubleshoot difficult problems.

1.1.1 Connect to Your Student Firewall

1. Launch the Configuration browser and connect to **`https://192.168.1.254`**.
2. On the "Your connection is not private" page, click **ADVANCED** and then click **Proceed to 192.168.1.254 (unsafe)**.
3. Log in to the Palo Alto Networks firewall. Use the following credentials:

Parameter	Value
Username	**`admin`**
Password	**`Pal0Alt0!`**

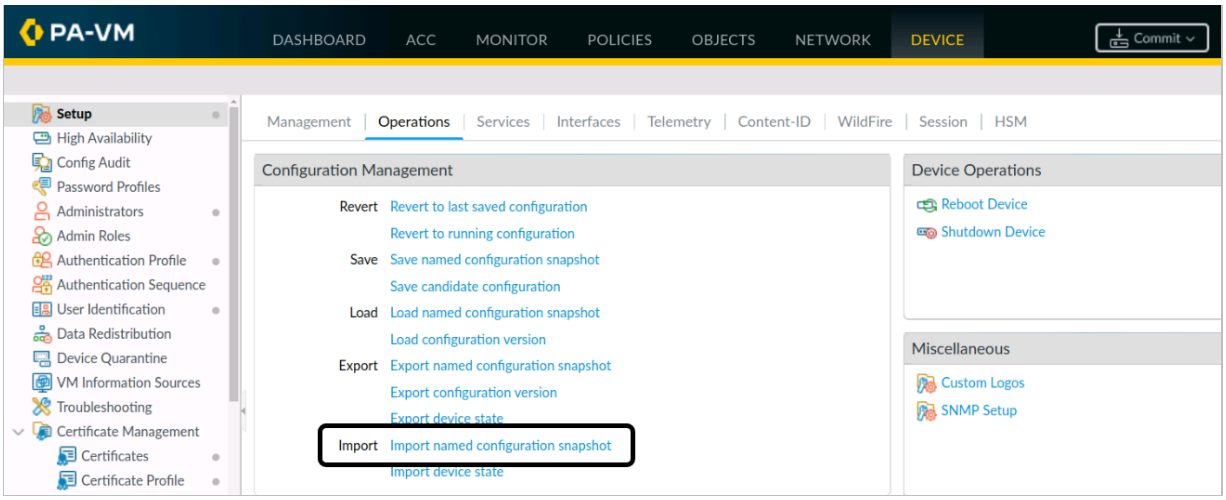
4. Check the check box **Do not show again** and click **Close** at the bottom of the Welcome window.
5. Click **Remind Me Later** at the Telemetry Data Collection window.

   Note: This step will need to be repeated whenever you are asked to log in to the firewall because this feature will not be used during the labs.

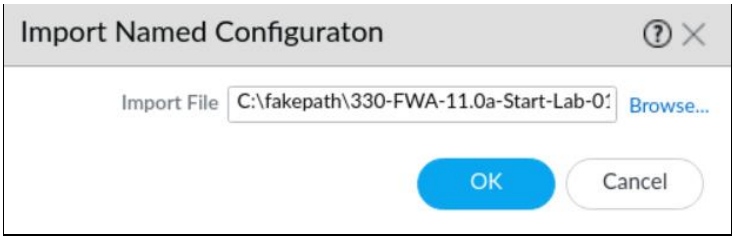
1.1.2 Apply a Baseline Configuration to the Firewall

6. In the Palo Alto Networks firewall web interface, select **Device > Setup > Operations**.

7. Click **Import named configuration snapshot**:



8. Click **Browse**.
9. Navigate to the **Desktop\Lab-Files\EDU-330\firewall-config-files** folder,
10. Select **330-FWA-11.1a-Start-Lab-01.xml**.
11. Click **Open**, and then click **OK**:

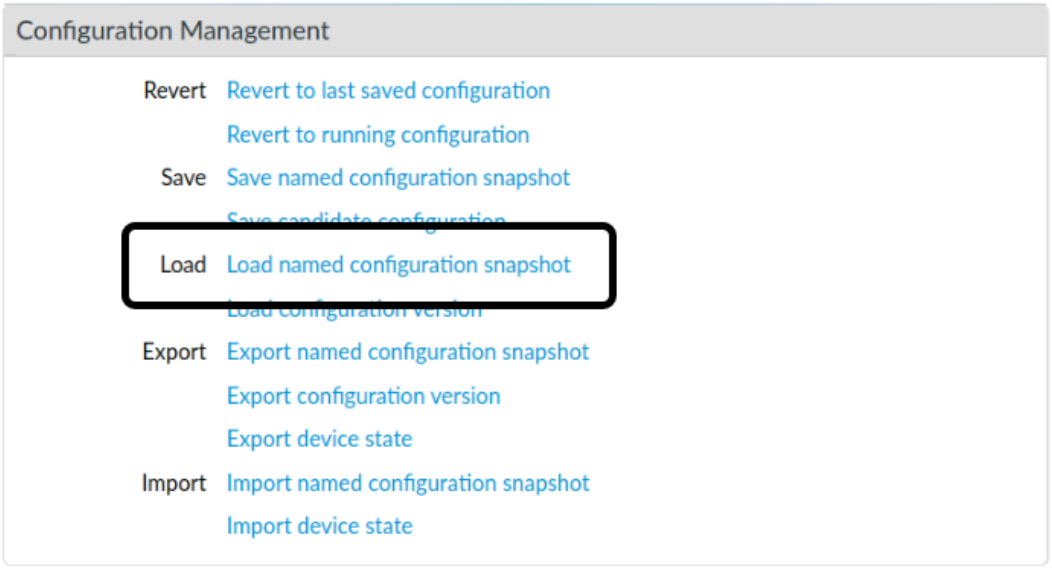


The firewall will display a notification that the imported file is saved.

12. Click **Close**.

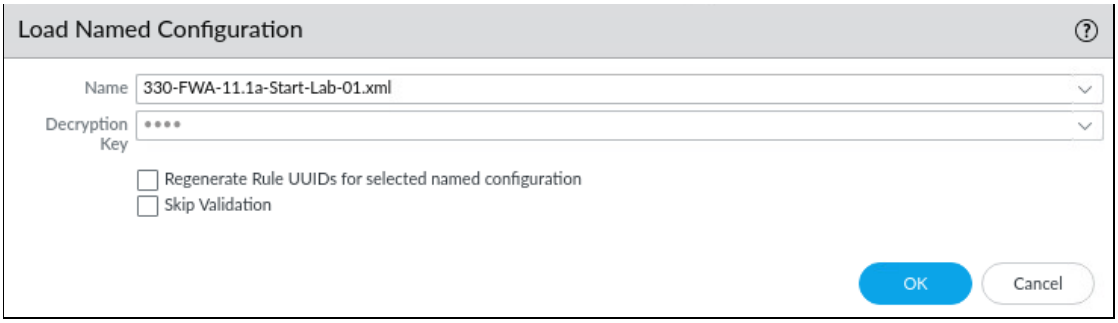
Now, you need to load the configuration you just imported.

13. Click **Load named configuration snapshot**:



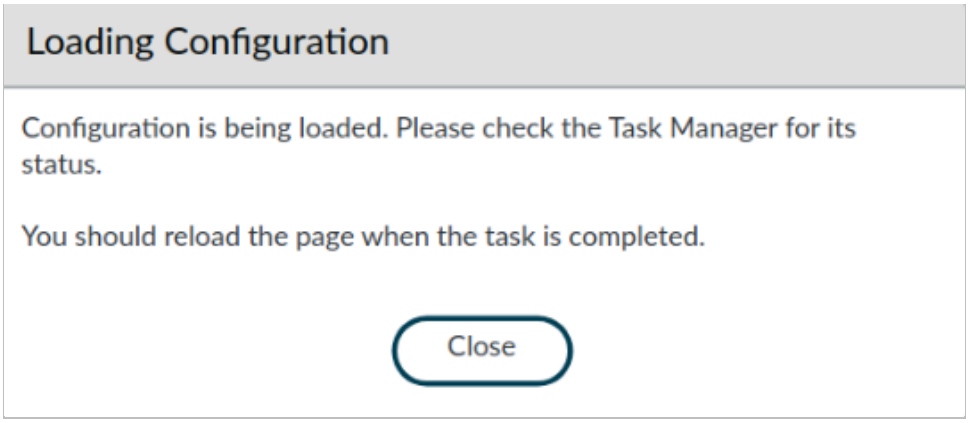
14. In the **Load Named Configuration** window, use the drop-down list next to the **Name** field to select **330-FWA-11.1a-Start-Lab-01.xml**.

    Leave all options in the bottom half of the dialog window unselected.



15. Click **OK**.

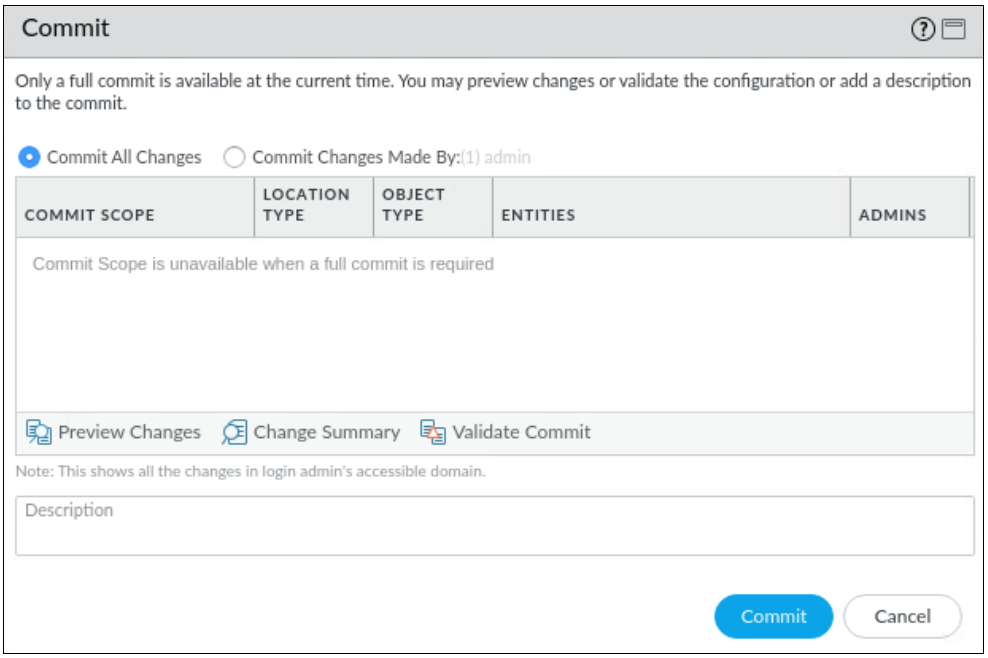
    The firewall will display the **Loading Configuration** prompt:



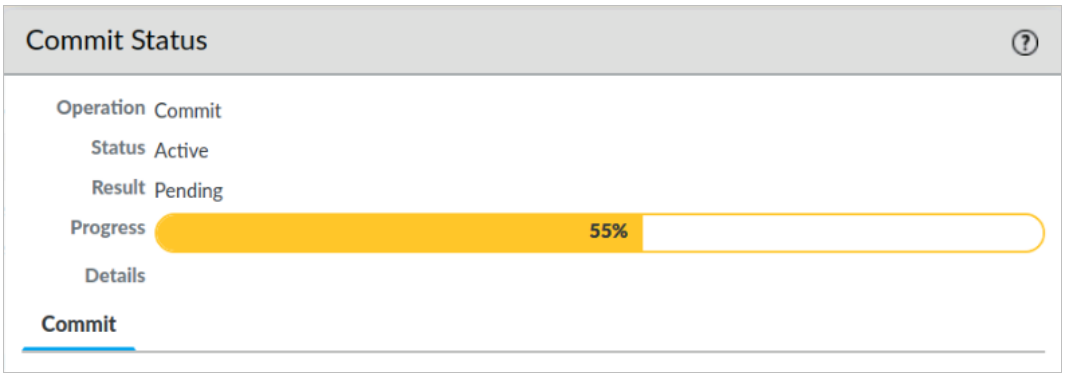
16. Click **Close**.

17. Click **Commit** at the upper right of the web interface:

The **Commit** dialog window is displayed:



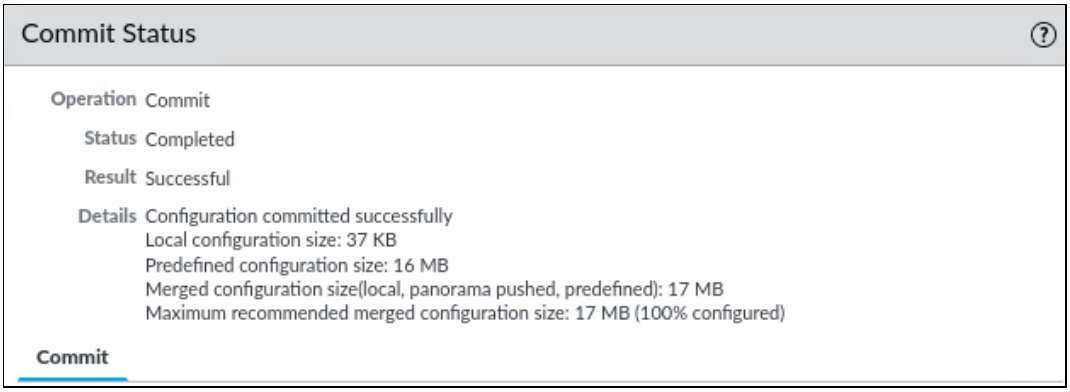
18. Add a brief description that includes the following: what was changed, why was the change made, who made the change, and the date and time.
19. Click **Commit** and wait until the commit process is complete:



It may take a few moments for the status to change to Completed.

If you receive a message regarding the deprecated algorithm used to generate the API KeyGen, ignore it. This message will have no effect on the labs.

There is a Bonus Lab at the end of this guide that will show you how to address this issue.



Verify the **Result** reported is "Successful" and the **Details** include "Configuration committed successfully."

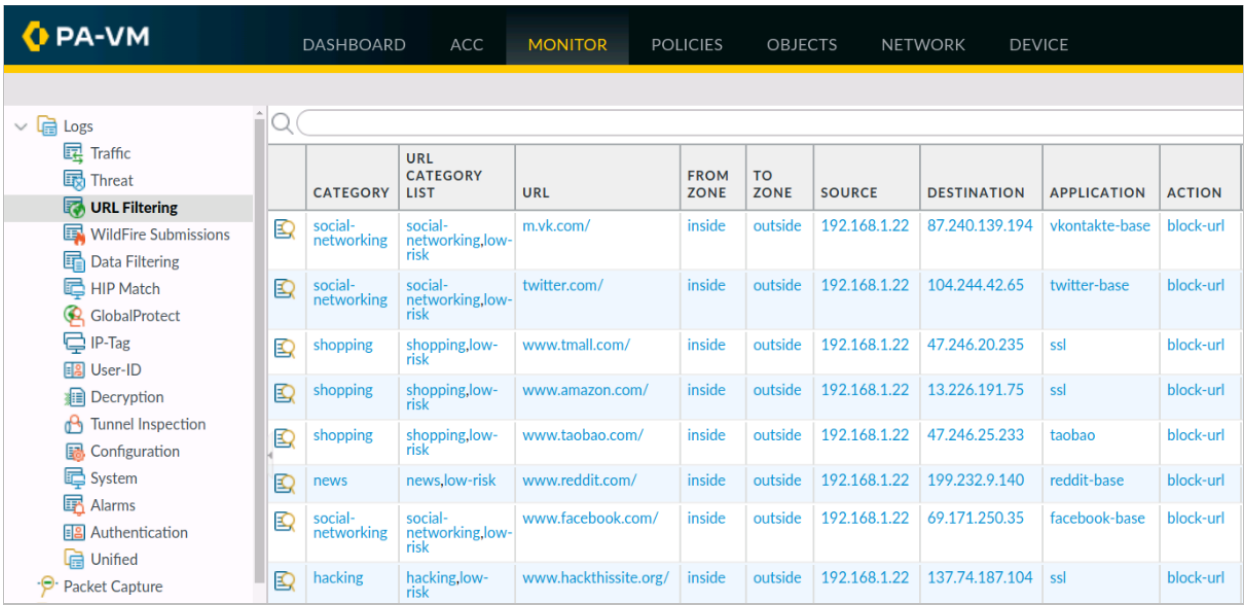
20. Click **Close**.

1.1.3 Validate the Basic Functionality of the System

If your system fails any of the functional tests that follow, work with your instructor to diagnose and troubleshoot the problem.

21. On your student desktop, open the Terminal and ping an external address such as **8.8.8.8**.
    Press Ctrl+C to stop the Ping after several seconds.
22. In the Terminal window, type **cd Common <ENTER>.**
23. Then type **./url-traffic.sh**.
    The **url-traffic.sh** script generates HTTP and HTTPS traffic from a list of about 50 popular websites. The process could take 7 to 10 minutes to complete. Ignore any errors in the script.
24. While **url-traffic.sh** is running, go to the web interface of the firewall.
25. Select **Monitor > Session Browser**.
    Verify that the firewall is creating sessions and that you can monitor traffic.
26. After one or two minutes, select **Monitor > Logs > Traffic**.
    Verify that the firewall is creating Traffic logs. You should be able to see new log entries created in which the application detected is "web-browsing," "ssl," or some other application that is not "dns" or "paloalto-updates."
27. After a minute or two, select **Monitor > Logs > URL Filtering**.

You should find that sessions that include URLs related to “social-networking,” “news,” and “shopping” have been blocked based on one or more of the existing Security policies. Look at the URL Category List and Action columns to verify:

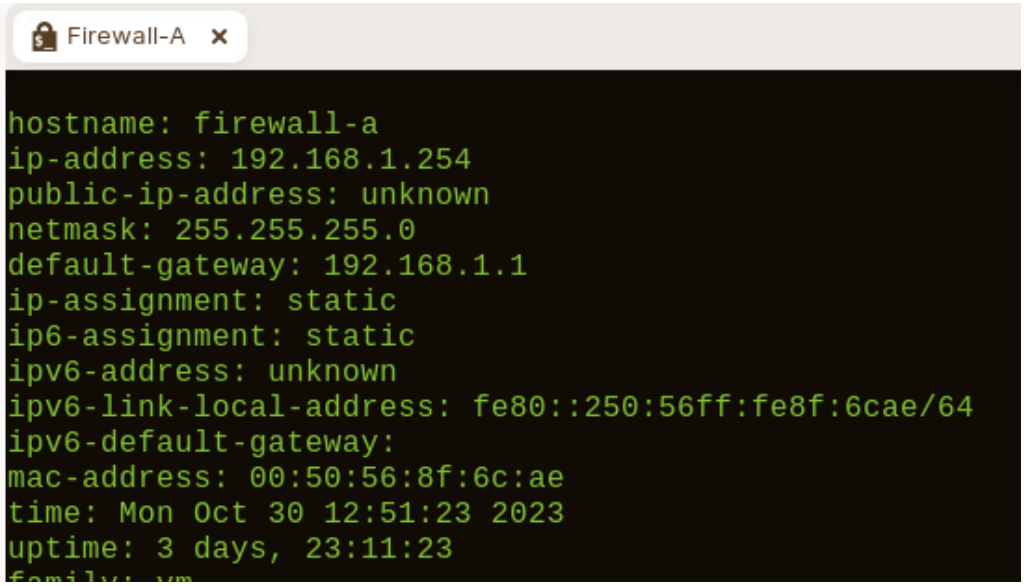


CATEGORY	URL CATEGORY LIST	URL	FROM ZONE	TO ZONE	SOURCE	DESTINATION	APPLICATION	ACTION
social-networking	social-networking,low-risk	m.vk.com/	inside	outside	192.168.1.22	87.240.139.194	vkontakte-base	block-url
social-networking	social-networking,low-risk	twitter.com/	inside	outside	192.168.1.22	104.244.42.65	twitter-base	block-url
shopping	shopping,low-risk	www.tmall.com/	inside	outside	192.168.1.22	47.246.20.235	ssl	block-url
shopping	shopping,low-risk	www.amazon.com/	inside	outside	192.168.1.22	13.226.191.75	ssl	block-url
shopping	shopping,low-risk	www.taobao.com/	inside	outside	192.168.1.22	47.246.25.233	taobao	block-url
news	news,low-risk	www.reddit.com/	inside	outside	192.168.1.22	199.232.9.140	reddit-base	block-url
social-networking	social-networking,low-risk	www.facebook.com/	inside	outside	192.168.1.22	69.171.250.35	facebook-base	block-url
hacking	hacking,low-risk	www.hackthissite.org/	inside	outside	192.168.1.22	137.74.187.104	ssl	block-url

28. On the student desktop, double-click the **Remmina** shortcut.
29. Double-click the **Firewall-A** configuration in the **Remmina Remote Desktop Client** window.

    Note: The login credentials have been pre-configured for you. There is no need to enter them.

30. At the firewall CLI, type the command:
    `show system info <ENTER>`:



Press the **spacebar** to display more data until the CLI returns you to the prompt.

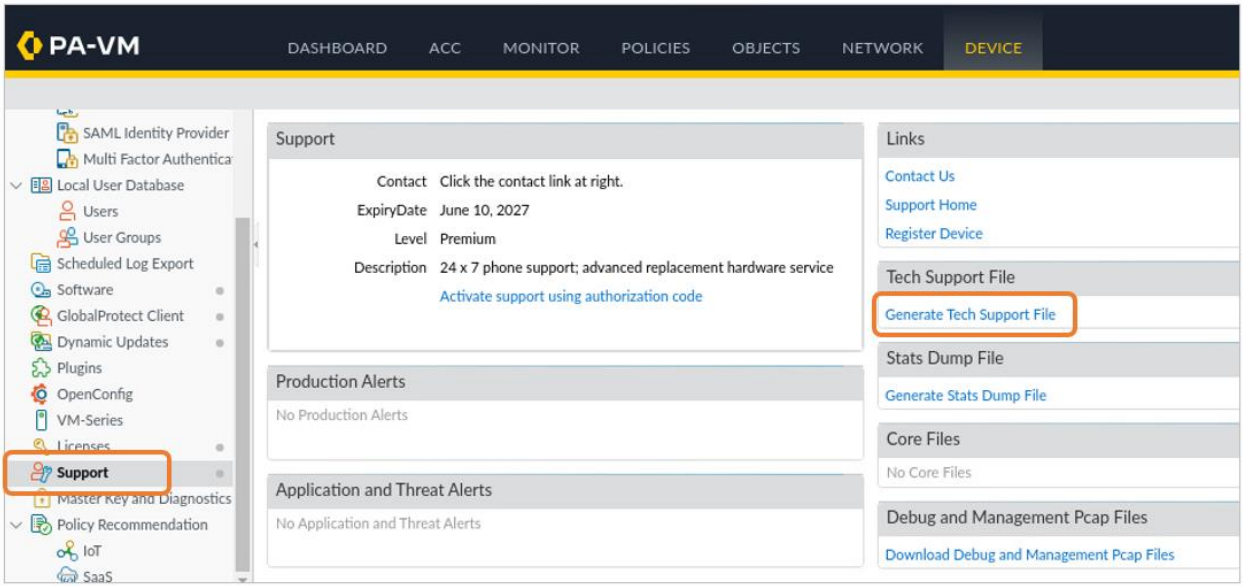
31. Review the available information and then type:
    `exit` and press **Enter**.

    The SSH session and the **Remmina** window should close.

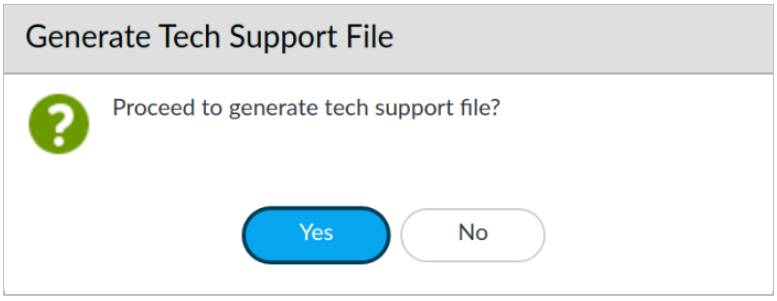
32. Close the Remmina Remote Desktop Client window.

1.1.4 Use the Web Interface to Get a Tech Support File

33. Select **Device** > **Support** to display the support page:

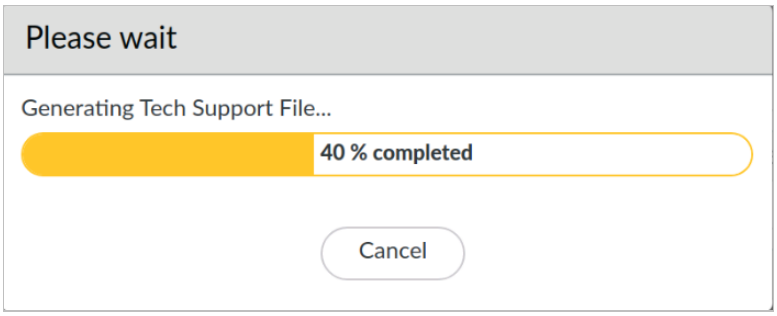


34. Click **Generate Tech Support File**.
35. When prompted to proceed, click **Yes**:



36. Monitor the progress.

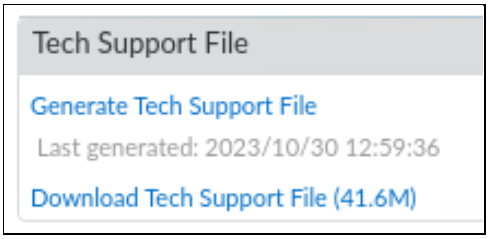
    The firewall typically will take two minutes to complete the generation of the file.



After the file is generated, the firewall will update the **Tech Support File** section of the support page with the following information:

- The time that the most recent Tech Support File was generated
- A download link that includes an indication of the size of the file in parentheses

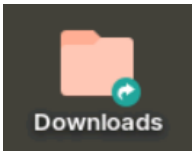
37. Click the link for **Download Tech Support File (<size>)**.



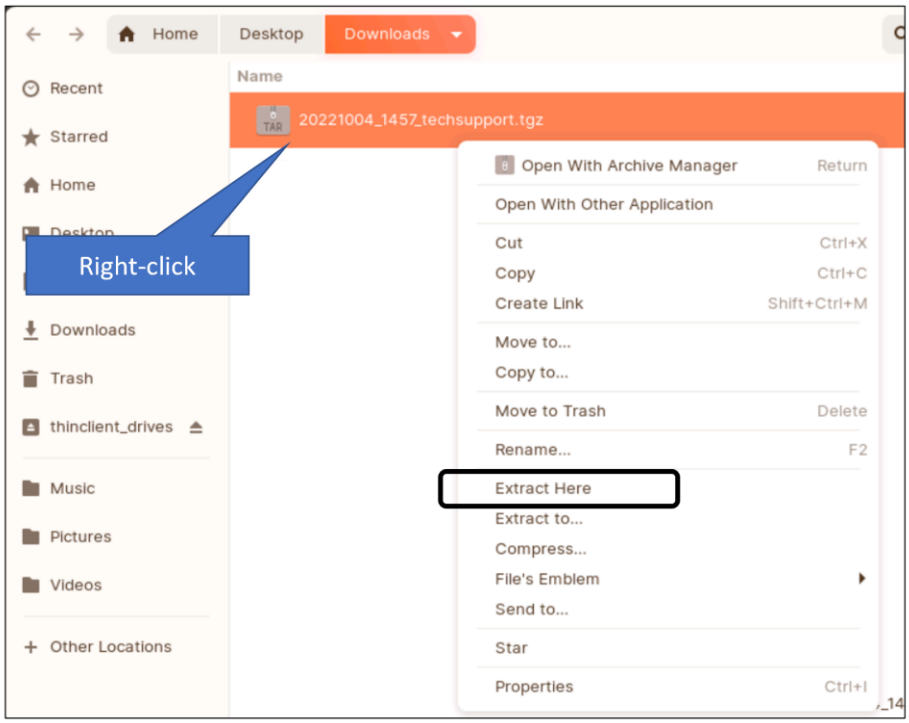
38. Save the file to the Downloads folder.

1.1.5 Decompress the Contents of the Tech Support File

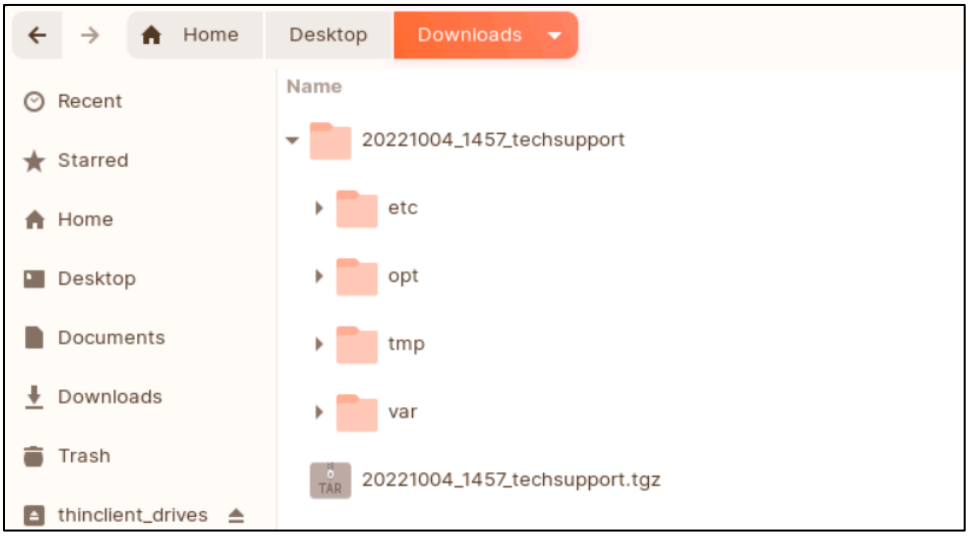
39. After the download is complete, open the Downloads folder on the Desktop:



40. Right-click the **<YYYYMMDD_HHMM>_techsupport.tgz** file and select **Extract Here**:



41. When the tgz file has been extracted, expand the folder to see the contents:

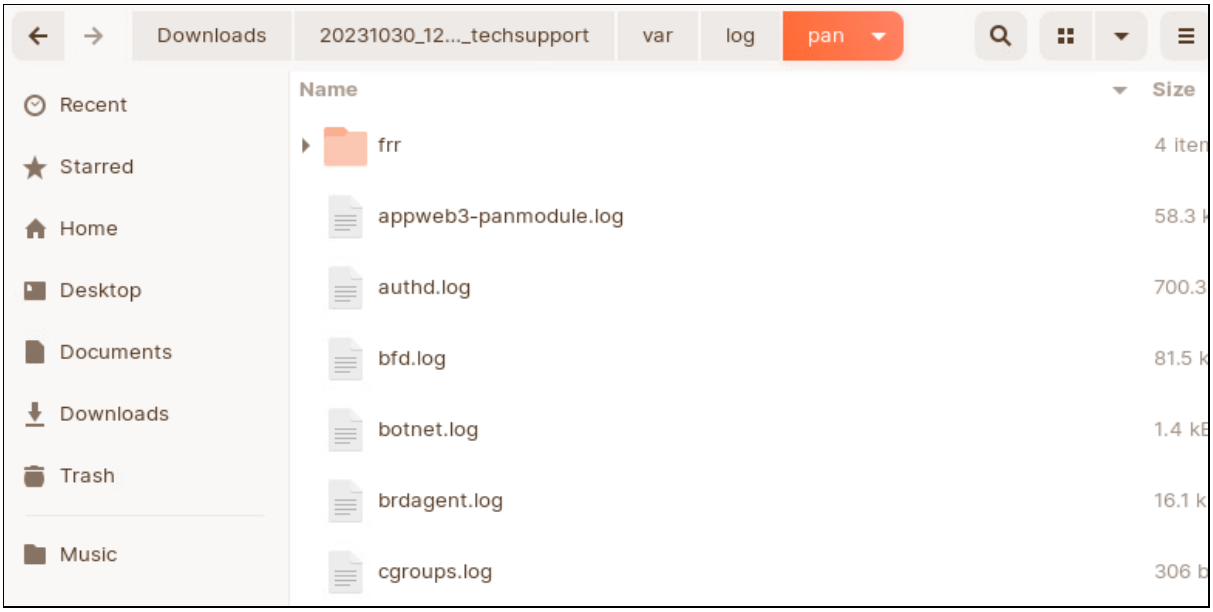


You should see four subfolders: etc, opt, tmp and var.

1.1.6 Explore the Tech Support File

42. Open the **var** > **log** > **pan** folder.

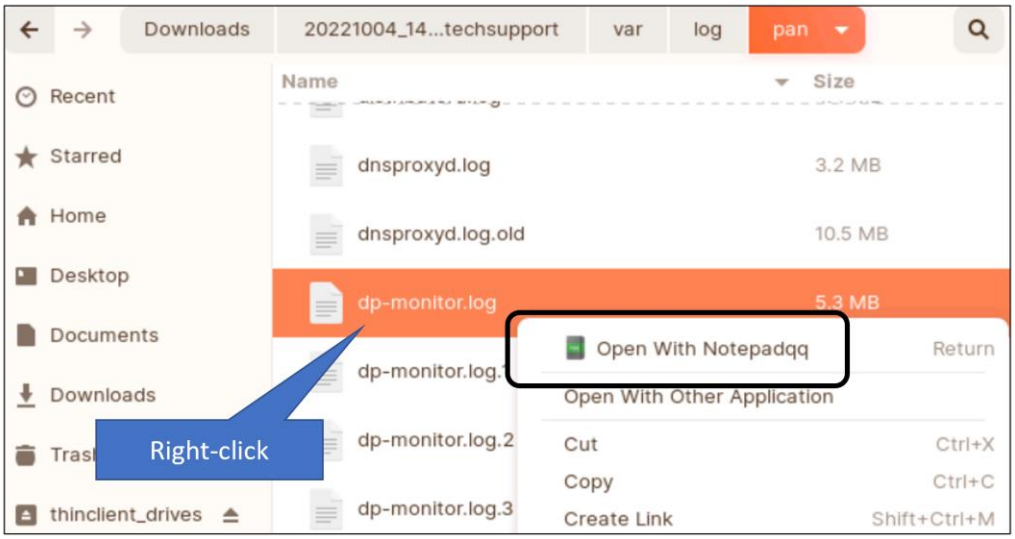
    The firewall writes logs for most PAN-OS® functions to the **/var/log/pan** folder:



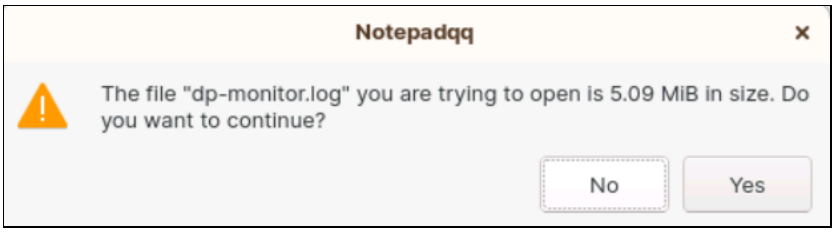
43. Look through the names of all the files in this directory.

    Notice that most of the logs are specific to individual services. The most important system services and corresponding logs that you should be aware of will be described in more detail in a section of the course dedicated to system services.

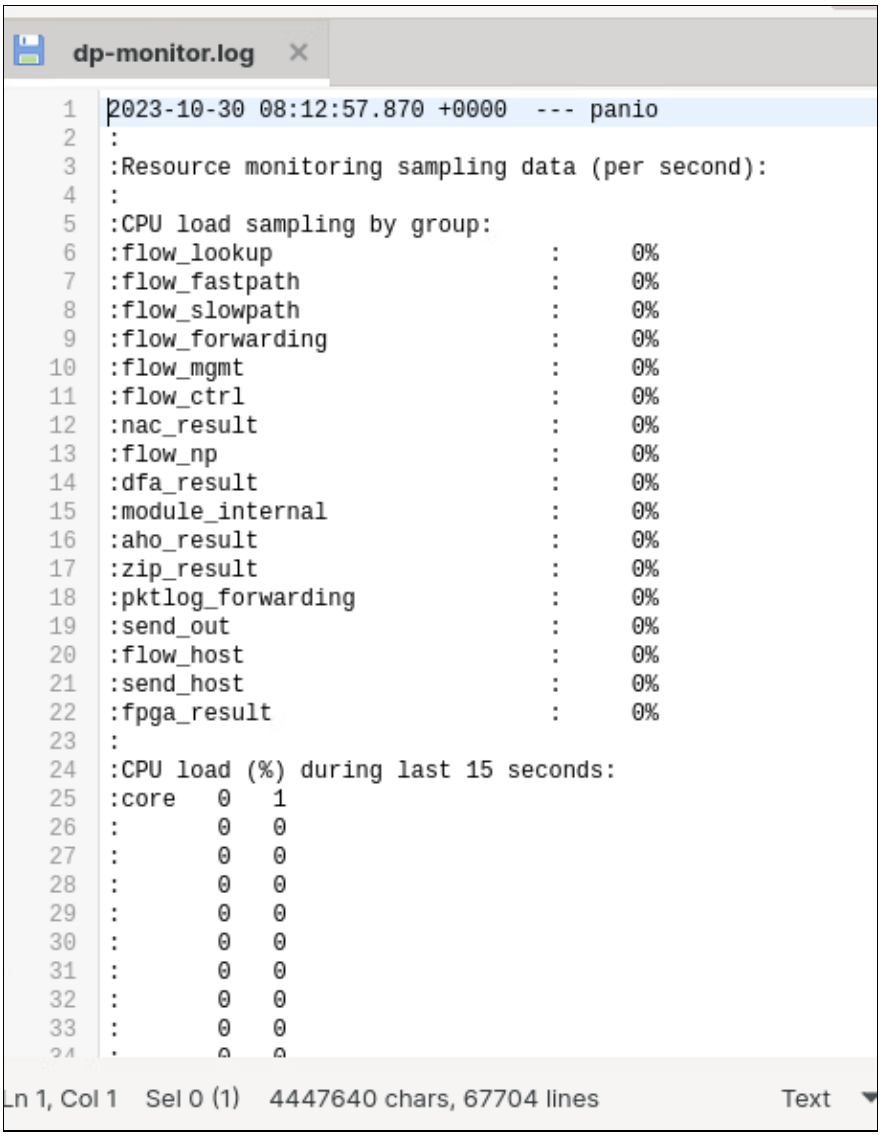
44. Open the **dp-monitor.log** file with **Notepadqq** by right-clicking and selecting **Open with Notepadqq**:



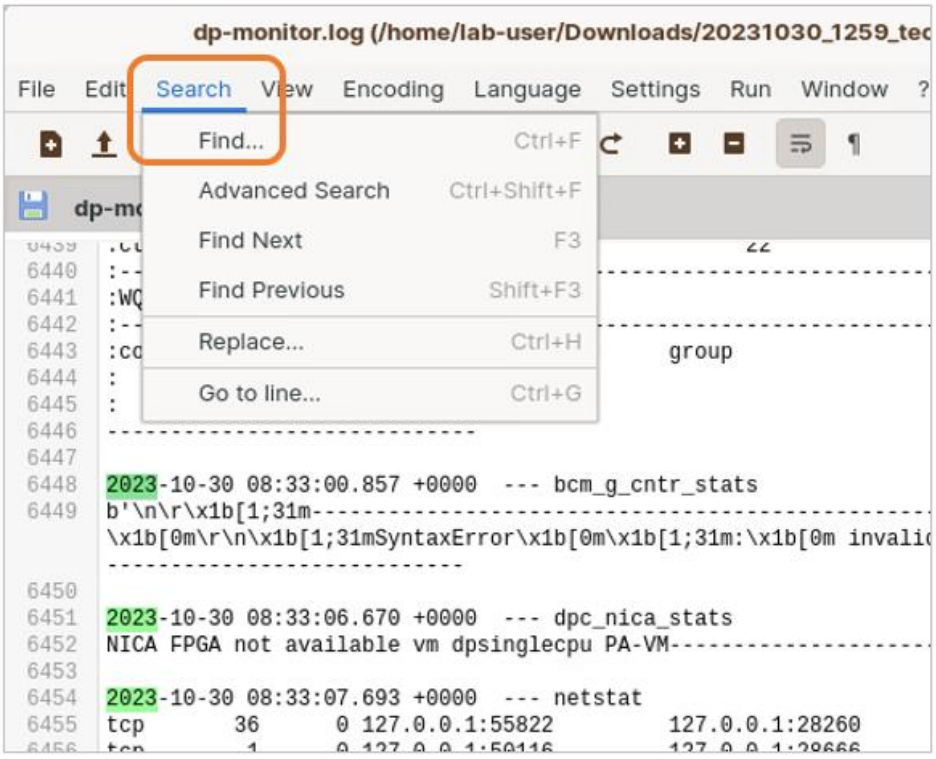
    If Notepadqq warns you about the size of the file, click Yes.



45. In **Notepadqq**, scroll through the **dp-monitor.log** file.
    What kind of information do you see? How often is some of it repeated?

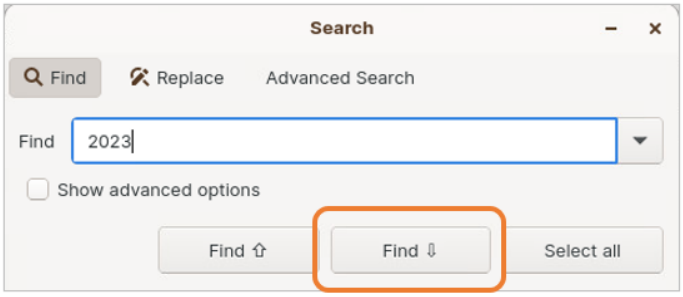


46. Find a date stamp within the **dp-monitor.log** file. Double-click the **year** of the date stamp (**20242025**, etc.), select **Search**, then **Find ↓** from the dropdown menu.



47. Then click **Find ↓** repeatedly to navigate down through the file.

    How frequently are "cpu" and "panio" statistics logged?



48. Go back to the top of the file and search for "**Global counters**" (case sensitive).
49. Click **Find** (or F3) repeatedly to navigate down through the file.

    Do you notice anything interesting about the interval between the logging for counters? Are they the same interval for all counters? Why might they be different?

50. Find and open the **mp-monitor.log** file and review it in the same way. Search for the current calendar **year** of the timestamp (such as 2024 or 2025) and repeat the search to navigate from heading to heading.

    Alternatively, you can search for " ---- " (space, four hyphens, space).

51. Explore other areas of the **Tech Support File** folder structure.
    - Find configuration and log files for various helper services in the **\opt** folder.
    - Find a copy of the running configuration in **\opt\pancfg\mgmt\saved-configs**.

- Find a summary file of the various commands that the firewall executes during the generation of the Tech Support File in the **\tmp\cli** folder.

52. Open **\tmp\cli\techsupport_<fw-name>_<YYYYMMDD>_<HHMM>.txt** in **Notepadqq** and review the various commands used to generate system-status information and the associated output of each command.

1.1.7 Clean Up Your Lab Environment

53. Close Notepadqq program when you are finished.
    Is there anything that you expected to find in the **Tech Support File** that you did not find?
54. Close any open **File Manager** windows.

Stop. This is the end of the lab.

1.2 Lab: Use the CLI to Export a Tech Support File

Lab Objectives

- Use the firewall CLI to generate a Tech Support File
- Use the firewall CLI to export a Tech Support File

Lab Scenario

The Tech Support File is a compressed archive of system logs:

- You want to see how the Technical Support File is generated.
- You need to export a Tech Support File because the web interface is not available.

Subsequent modules and lab activities will provide more detailed information about many of these files. For now, your goal simply should be to develop a familiarity with the names of various files that you might see when you troubleshoot difficult problems.

1.2.1 Use the CLI to Generate a Tech Support File

1. On the **Desktop**, double-click the **Remmina** shortcut.
2. Double-click the **Firewall-A** connection in the **Remmina Remote Desktop Client** window.
3. Type:
   `find command keyword support` and press **Enter**:

```
admin@firewall-a> find command keyword support

show log traffic session-end-reason equal <unknown|aged-out|decoder|tcp-reuse
|tcp-fin|tcp-rst-from-server|tcp-rst-from-client|resources-unavailable|policy-deny
|threat|decrypt-error|decrypt-unsupport-param|decrypt-cert-validation|n/a>
show log traffic session-end-reason not-equal <unknown|aged-out|decoder|tcp-euse
|tcp-fin|tcp-rst-from-server|tcp-rst-from-client|resources-unavailable|policy-deny
|threat|decrypt-error|decrypt-unsupport-param|decrypt-cert-validation|n/a>
debug techsupport duts add-search-dir <value>
debug techsupport duts set-byte-threshold <0-1073741823>
debug techsupport duts on
debug techsupport duts off
debug techsupport duts reset-config
debug techsupport duts run
request hsm support-info
request tech-support dump
request support info
request support check
scp export tech-support to <value> remote-port <1-65535> source-ip <ip/netmask>
scp export hsm-support-info from <value> to <value> remote-port <1-65535> source-ip
<ip/netmask>
tftp export tech-support to <value> remote-port <1-65535> source-ip <ip/netmask>
```

The **find command keyword <string>** command will display all available commands that contain the named string (in this case, "support"), including all arguments, parameters, and fixed values that are available as options for a command.

When you do not know the exact command you need, use of the **find command keyword <string>** command often is the best way to begin your search for the required details.

Note: The parameter "**duts**" refers to the Linux disk-usage command "**du**" and technical support, abbreviated as "**ts**." The **debug techsupport duts run** command displays disk-usage information for all log files.

To generate a Tech Support File, you do *not* need to use the **request tech-support dump** command, which is the equivalent of clicking **Generate Tech Support File** in the web interface.

When you use an **SCP** or **tftp export** command, a new Tech Support File will be generated automatically. (In the lab, you will use SCP.)

1.2.2 Use the CLI to Export a Tech Support File

The following steps will generate and export a Tech Support File to the Downloads folder of your student desktop.

4. In the firewall CLI, type:
scp export tech-support to lab-user@192.168.1.20:./Downloads/ and then press **Enter**:

```
admin@firewall-a> scp export tech-support to lab-user@192.168.1.20:./Downloads/
Group 'batch' suspend
Collecting command output...
configure
save config to techsupport-saved-currcfg.xml
exit
show admins all
show clock
show system software status
show jobs pending
show jobs processed
show system info
show system files
[. . .]
```

Notice the various commands that the firewall runs to package the Tech Support File for export. You can use most of these commands individually on a live system to retrieve targeted information.

After the firewall completes the packaging of the Tech Support File, the last several lines of CLI output should look similar to the following example:

```
[. . .]
Measuring disk usage...
Group 'batch' resume
```

```
Finish generating tech support.
The authenticity of host '192.168.1.20 (192.168.1.20)' can't be established.
ECDSA key fingerprint is 5f:73:e3:8b:78:8c:cd:63:a6:59:ff:1b:2d:06:5a:0b.
Are you sure you want to continue connecting (yes/no)? yes [Enter]
```

After the Tech Support File is completed, the firewall will connect to the student desktop. The student desktop will return an SSH key that the firewall may not have seen before.

IMPORTANT: You **may** be prompted to confirm that you want to continue connecting; if not, skip to step 6.

5. Type:
   `yes` and press **Enter**.
   The firewall may display a warning that the new key has been permanently added to the list of known hosts. Then the firewall will prompt you for the password for the lab-user account on the student desktop:

```
[. . .]
Warning: Permanently added '192.168.1.20' (ECDSA) to the list of known hosts.
lab-user@192.168.1.20's password:Pal0Alt0! [Enter]
```

6. Type:
   `Pal0Alt0!` and press **Enter**.
   The firewall will transfer the file to the student host, report completion statistics, and return you to the CLI prompt:

```
[. . .]
PA_01234567890123456_ts_90.0_20190317_0509.tar.gz        100%   22MB  16.2MB/s
00:00
admin@firewall-a>
```

Take note of the name of the **`PA_*.tar.gz`** file that is transferred. You will need to find this file in the Downloads folder of your student desktop.

7. Use the window controls of the **Remmina** CLI window to scroll back through the command output. Look for and find instances of the script **`/usr/local/bin/remove-private-info.sh`**. This script removes private information from log files.
   Note: This will not be visible from the desktop. It's only visible in the firewall's command output.
   Did this script run?
   Does it run for more than one file?
   Were any errors reported by this script?
   Why is running a script like this important?

8. Continue to scroll up and look through all the commands that the firewall executed to package the Tech Support File.

9. Find the command **`show system software status`**.
10. Use your mouse to select all four words of this command. After the text is selected, **right-click** and copy and paste it into the active command prompt. Press **Enter** to execute the command.

    You should see output that is similar to the following:

```
admin@firewall-a> show system software status

Slot 1, Role mp
----------------------------------------

Type     Name                            State    Info
Group    all                             running
Group    base                            running
Group    batch                           running
Group    batch_secondary                 running
Group    chassis                         running
Group    data_plane                      running
Group    dataplane_zone                  running
Group    dsms                            running
Group    fips                            running
Group    grp_plugins                     running
Group    ha_ssh                          running
Group    management_zone                 running
Group    mgmt_services                   running
Group    ntlm-grp                        running
Group    service_zone                    running
Group    services                        running
Group    supervisor                      running
Group    tasks                           running
Group    third_party                     running
Process  all_task                        running  (pid: 6772)
Process  authd                           running  (pid: 5613)
Process  bfd                             running  (pid: 6972)
[ . . . ]
```

    The output shows key firewall processes, process groups, current running states, and the process ID ("pid") for individual processes that are in the "running" state.

    We will cover commands like the one shown above in other modules and activities.

11. Test two or three other commands that look interesting and useful to you in your environment.

1.2.3 Validate the Exported Tech Support File

12. On the student desktop, open the **Downloads** folder.
13. Right-click and select **Extract Here** to decompress the contents of the **Tech Support File**.

    Use the techniques that you used in the previous lab activity to locate and decompress the contents of the Tech Support File.

14. (Optional) Briefly explore the contents of the **Tech Support File**.

    Review the contents of the subfolder **\var\log\pan**.

    Use **Notepadqq** to view log files.

1.2.4 Reference Information

Search the LIVE Community for the article “How to Generate and Upload a Tech Support File Using the WebGUI and CLI.”

1.2.5 Clean Up Your Lab Environment

15. Select the **Remmina** session on the student desktop.
16. Exit the CLI by typing **`exit`** and pressing **Enter**.
17. Close Notepadqq program.
18. Close the Remmina Remote Desktop Client window.
19. Close any open **File Manager** windows.

Stop. This is the end of the lab.

1.3 Lab: CLI Fundamentals

Lab Objectives

- Explore the CLI, including configuration mode and the configuration hierarchy
- Use CLI tools to find commands and use them correctly
- Change a setting in the device configuration and commit the change

Lab Scenario

In this lab, you will step through the procedures that enable you to:

- Use the CLI to load and commit a new configuration
- Review existing configuration settings
- Modify device settings and commit the candidate configuration

After you make configuration changes by using the CLI, you will see how your changes are reflected in the web interface.

1.3.1 Import, Load, and Commit a Configuration File

You will use the CLI of the firewall to import a configuration file, load the configuration file so that it becomes the candidate configuration, and then **Commit** the configuration so that it becomes the running configuration.

1. Double-click the **Remmina** shortcut on the student desktop if it is not already open.
2. In the **Remmina Remote Desktop Client** window, double-click the **Firewall-A** entry.
3. Type the command:
   `find command keyword upload` and press **Enter**.

   Try to find the command that you will need to use to upload (or import) the **330-FWA-11.1a-Start-Lab-1.3.xml** configuration file:

```
admin@firewall-a> find command keyword upload
show log data action equal <alert|allow|deny|drop|drop-all|reset-client|reset-se[...]
show log data action not-equal <alert|allow|deny|drop|drop-all|reset-client|rese[...]
debug swm load-uploaded image <value>
debug dataplane set blocked-forward upload yes
debug dataplane set blocked-forward upload no
debug wildfire upload-log show channel <public|private>
debug wildfire upload-log log max-size <1-50>
debug wildfire upload-log log extended-log <yes|no>
debug wildfire upload-log log disable
debug wildfire upload-log log enable
[. . .]
```

Scan the list of commands for anything related to configuration files. Nothing seems to be related to configuration files.

4. Type the command:
   `find command keyword import` and press **Enter**:

```
admin@firewall-a> find command keyword import
show routing protocol bgp policy virtual-router <value> import
set management-server logging <on|off|import-start|import-end>
set system setting shared-policy <enable|disable|import-and-disable>
set system setting template <enable|disable|import-and-disable>
request certificate import-scep-ca-cert certificate-name <value> scep-profile <v[...]
scp import idp-metadata profile-name <value> max-clock-skew <value> validate-met[...]
scp import configuration from <value> remote-port <1-65535> source-ip <ip/netmask>
scp import ui-translation-mapping from <value> remote-port <1-65535> source-ip <[...]
scp import private-key from <value> remote-port <1-65535> source-ip <ip/netmask>[...]
scp import keypair from <value> remote-port <1-65535> source-ip <ip/netmask> pas[...]
scp import logdb from <value> remote-port <1-65535> source-ip <ip/netmask>
:
```

There is a lot of text to sort through. Remember that CLI output that exceeds the line-height of the current window will be displayed automatically through the Linux **`less`** program.

If you are using a computer with a large display and are working in full-screen mode, Remmina may be able to display all the command output that is referenced above on a single screen. If this is the case for you, to complete subsequent lab steps as written, type **`find command`** (with no other parameters) and press **Enter**. This command will generate enough output for you to be able to use the following steps to practice using the **`less`** program to find what you need.

5. Type:
   `/configur` and press **Enter**.

   This **`less`** program searches the data displayed for the string "configur".

   You also can search for "configuration," but searching for "configur" will enable you to find instances of "configure" and "configuration" in one search.

```
[. . .]
scp import global-protect-clientless-vpn from <value> remote-port <1-65535> sour[...]
scp import global-protect-client from <value> remote-port <1-65535> source-ip <i[...]
tftp import anti-virus from <value> file <value> remote-port <1-65535> source-ip[...]
tftp import wildfire from <value> file <value> remote-port <1-65535> source-ip <[...]
tftp import device-state from <value> file <value> remote-port <1-65535> source-[...]
/configur [Enter]
```

Note: Make sure to type **`/configur`** before all the output text of the **`find command keyword <keyword>`** command is displayed.

The CLI automatically uses **`less`** as a paging function for long outputs. You can type **n** to search for the next occurrence of the search string (or type **N** to search for a prior occurrence). However, unlike with *explicit* use of the **`less`** command to display a file, after the display output has reached the end of the data to display, the CLI automatically exits the **`less`** command.

After you return to the CLI command prompt **username@<firewall-name>**, you no longer can use **less** commands to navigate the output data.

```
scp import idp-metadata profile-name <value> max-clock-skew <value> validate-met[...]
scp import configuration from <value> remote-port <1-65535> source-ip <ip/netmas[...]
scp import ui-translation-mapping from <value> remote-port <1-65535> source-ip <[...]
scp import private-key from <value> remote-port <1-65535> source-ip <ip/netmask [...]
[. . .]
tftp import pandb-url-database from <value> file <value> remote-port <1-65535> s[...]
tftp import global-protect-client from <value> file <value> remote-port <1-65535[...]
tftp import configuration from <value> file <value> remote-port <1-65535> source[...]
tftp import certificate from <value> file <value> remote-port <1-65535> source-i[...]
```

In the output, the command you need is the command that begins with **scp import configuration**. But, try one more time to limit the output of the find command to narrow in on the task you need to accomplish.

6. After you return to the main CLI prompt, type:
   find command keyword configuration and press **Enter**:

```
admin@firewall-a> find command keyword configuration
debug software resource subsystem <value> plane <value> slot <0-64> show configuration
scp import configuration from <value> remote-port <1-65535> source-ip <ip/netmask>
scp export configuration from <value> to <value> remote-port <1-65535> source-ip <ip/netmask>
tftp import configuration from <value> file <value> remote-port <1-65535> source-ip <ip/netmask>
tftp export configuration from <value> to <value> remote-port <1-65535> source-ip <ip/netmask>
```

In the following step, you will use the select-and-paste function of **Remmina** to copy the first part of the command to the command line.

7. Use your mouse to select the text **scp import configuration from** in the CLI output from the previous find command. Then **right-click** and copy and paste it into the active command prompt. Press **Enter** to execute the command.

```
scp import configuration from
```

8. If the cursor is positioned immediately after the "**m**" in "**from**," press **Tab** twice.
   If there already is a space after the "**m**," press **Tab** once.

   This procedure will provide suggestions and help text that may help you to understand the "<value>" that the parameter "from" requires, as you saw in the prior **find command** output.

```
admin@firewall-a> scp import configuration from [Tab]
  <value>  Source (username@host:path)
```

Notice that the command requires the value of the **scp host**. The CLI accepts host values in the form of username@host:path. You can display exactly which parameters are *required*, such as **from**, and which ones are *optional*, such as **remote port**, by returning to the base form of the command and its initial options and pressing **Tab**.

9. Delete the parameter "**from**" from the current command line and press **Tab**.

   If there is not a space after the parameter "**configuration**" when you press **Tab**, you have two options: Add a **space** and then press **Tab**, or press **Tab** twice:

```
admin@firewall-a> scp import configuration [Tab]
+ remote-port   SSH port number of remote host
+ source-ip     Set source address to specified interface address
* from          Source (username@host:path)
```

   Notice that some of the available options are marked with a plus sign (**+**) or an asterisk (*****). An option marked with a plus sign is optional. An option marked with an asterisk is an option that *must* be used to execute the command.

   You have found the correct command to import the configuration. Now, you need to find the IP address of student desktop. You can get this information from the student desktop itself by running **ifconfig -a** from a terminal window. You can reference the lab topology diagram. You also can use the CLI of the firewall, which can tell you the IP addresses of all open connections to the web interface.

   You now should be connected to the firewall twice, once each through the CLI and the web interface. Try to find a command that will show you the IP addresses of all connected admins.

10. Type:
    find command keyword admin and press **Enter**:

```
admin@firewall-a> find command keyword admin
delete admin-sessions username <value>
show config list admins partial shared-object <excluded> device-and-network excl[...]
show config list changes partial shared-object <excluded> device-and-network  <e[...]
show config list change-summary partial admin  [ <admin1> <admin2>... ]
show log alarm admin equal <value>
show log alarm ack_admin equal <value>
show log auth clienttype equal <unknown|Admin UI|CLI|GlobalProtect Portal|Global[...]
show admins all
show admins local
debug list-admin-history
debug device-server test admin-override-password <value>
request authentication unlock-admin user <value>
request commit-lock remove admin <value>
[. . .]
```

    The first command, **delete admin-sessions username <value>**, references admin sessions, but **delete** is not the action you want. You want to **show**.

11. Use your mouse to select the text **show admins all** and **right-click** and copy and paste it into the active command prompt.
12. Press **Enter** to execute the command.

```
admin@firewall-a> show admins all

admin
```

```
panorama
```

A list of admins that now are configured for the firewall in the lab is displayed. However, the command produces no IP addresses. Perhaps we do not have the correct command.

The following steps will illustrate a key limitation of using **find command**. If you know this limitation, you can take measures to work around it. For example, the parameter **all** may not be required.

Could a simpler version of the **`show admins all`** command produce more or different information?

13. Type:
`show admins`, press the **spacebar**, and then press **Tab**.
Or type **sh[Tab] ad[Tab] [Tab]**.

This procedure shows you all the available options for the **`show admins`** command:

```
admin@firewall-a> show admins [Tab]
+ all      All administrators
+ local    All local administrators
  |        Pipe through a command
  <Enter>  Finish input
```

Note that the **`all`** parameter is marked with a plus sign ("**+**"), which indicates that it is optional. Also note that the option **`<Enter> Finish input`** is listed.

14. Execute the **`show admins`** command, *without* the **`all`** parameter:

```
admin@firewall-a> show admins [Enter]

Admin               From   Client Session-start        Idle-for   Session-expiry
------------------------------------------------------------------------------------
* admin     192.168.1.20      CLI 04/26/2023 13:48:47  00:00:00s  05/26/2023 13:48:47
  admin     192.168.1.20      Web 04/25/2023 15:21:43  00:15:54s  05/25/2023 15:21:43
```

Note that the IP address of your student desktop is **`192.168.1.20`**.

Now that you have the IP address of your workstation, you can import the configuration file **330-FWA-10.1-Start-02** from IP address 192.168.1.20.

15. Type:
`scp import configuration from lab-user@192.168.1.20:./Desktop/Lab-Files/EDU-330/firewall-config-files/330-FWA-11.1a-Start-Lab-1.3.xml`
and press **Enter**:

Note: Password is `Pal0Alt0!`

```
admin@firewall-a> scp import configuration from lab-user@192.168.1.20:./Desktop/Lab-
Files/EDU-330/firewall-config-files/330-FWA-11.1a-Start-Lab-1.3.xml
lab-user@192.168.1.20's password:

330-FWA-11.1a-Start-Lab-1.3.xml saved
```

Use the output from the **scp import configuration** command to confirm that the file was received and that it was saved.

Next, you will use the CLI to *load* the configuration file **330-FWA-11.1a-Start-lab-1.3.xml** and thereby make it the current candidate configuration.

First use the CLI to **find** the required **load** command.

16. In the CLI, type:
find command keyword load and press **Enter**:

```
admin@firewall-a> find command keyword load
show cloud-appid cloud-app-data app-metadata payload
show log data action equal <alert|allow|deny|drop|drop-all|reset-client|reset-se[...]
show log data action not-equal <alert|allow|deny|drop|drop-all|reset-client|rese[...]
[. . .]
debug set-content-download-retry attempts <1-3>
debug syslog-ng reload
debug swm load image <value>
debug swm load-uploaded image <value>
debug dataplane set blocked-forward upload yes
[. . .]
set session distribution-policy session-load
request plugins upload name <value> path <value>
request plugins download file <value> sync-to-peer <yes|no>
request system software download scp-profile <value> sync-to-peer <yes|no> to-ve[...]
request system software download scp-profile <value> sync-to-peer <yes|no> versi[...]
request system software download scp-profile <value> sync-to-peer <yes|no> file [...]
[. . .]
```

Scan the first word of each command returned and note that no commands for loading configurations are returned.

In the following steps, you will use the same **find** command in configuration mode.

17. Type:
conf[Tab] and press **Enter**.

This procedure will execute the command required to enter configuration mode:

```
admin@firewall-a> configure
Entering configuration mode
[edit]
admin@firewall-a#
```

Note the change in the prompt from **>** to **#**.

The **#** prompt confirms that you are in configuration mode.

18. Type:
find command keyword load and press **Enter**:

```
admin@firewall-a# find command keyword load
```

```
load config key <value>|<default> regenerate-rule-uuid-all <yes|no> skip-validate
<yes|no> from <value>

load config key <value>|<default> regenerate-rule-uuid-all <yes|no> skip-validate
<yes|no> version <value>|<1-1048576>

load config key <value>|<default> regenerate-rule-uuid-all <yes|no> skip-validate
<yes|no> last-saved

load config key <value>|<default> regenerate-rule-uuid-all <yes|no> skip-validate
<yes|no> partial shared-objects <included> shared-policies
[. . .]
show deviceconfig high-availability group mode active-active virtual-address <na[...]
show deviceconfig high-availability group mode active-active virtual-address <na[...]
[. . .]
```

The first few commands and the available options seem to represent actions similar to the task of loading an existing configuration file. But, you are not sure about the **key** option.

Is the **key** option required? To what might the **key** option refer?

Use CLI suggestions and explanatory text to help you to discover the exact command that you need. Press the **spacebar** as needed, or type **q**, to return to the command prompt.

19. Type:
load conf[Tab] [Tab].

This procedure displays the available options for the **load config** command:

```
admin@firewall-a# load conf[Tab] [Tab]
+ key          key
+ regenerate-rule-uuid-all   Regenerate UUID for all rules; ignore existing UUID[...]
+ skip-validate              Skip validation for loaded config to improve load p[...]
> from         Filename
> last-saved   Last saved configuration
> partial      partial config loading
> version      Version
```

Note that no single option is required, but that the **<Enter>** option is not available.

The **from** parameter looks like the one you need to use to load a configuration file by filename.

20. Type:
load config fr[Tab] [Tab].

This procedure will list all the configuration files currently loaded on the firewall:

```
admin@firewall-a# load config fr[Tab] [Tab]
  330-FWA-11.1a-Start-Lab-01.xml          2023/04/21 04:08:07       33.0K
  330-FWA-11.1a-Start-Lab-1.3.xml         2023/04/21 04:54:48       51.5K
 [. . . ]
  running-config.xml                      2023/04/21 01:02:48       33.0K
  techsupport-saved-currcfg.xml           2023/04/21 09:26:13       33.0K
  <value>                                 Filename
```

Sometimes the CLI provides a list of parameters that consist of existing files or other values based on the *current state* of the firewall.

21. Use your mouse or touchpad to select the name of the target file **`330-FWA-11.1a-Start-Lab-1.3.xml`**, and then **right-click** and copy and paste it into the end of the current command.

    If you add a **space** and press **Tab**, you can see if the command supports any more options after specifying the filename. In this case, it does not. The only option is **`<Enter> Finish input`**.

```
admin@firewall-a# load config from 330-FWA-11.1a-Start-Lab-1.3.xml [Tab]
  <Enter>  Finish input
```

    After you add a **space** and press **Tab**, the prior command is returned to the active prompt.

22. Verify that the current command reads:
    `load config from 330-FWA-11.1a-Start-Lab-1.3.xml` and then press **Enter**.

    This command loads the named configuration:

```
admin@firewall-a# load config from 330-FWA-11.1a-Start-Lab-1.3.xml [Enter]

Config loaded from 330-FWA-11.1a-Start-Lab-1.3.xml

[edit]
```

    Verify that the output from the command indicates that the target configuration was loaded.

    In configuration mode, after you execute a command, your current position in the configuration hierarchy is reported to the screen. In this case, the position is **`[edit]`**, which is the *root* of the hierarchy.

    You now are ready to **commit** the configuration to the running state of the firewall.

23. Type:
    `find command keyword commit` and press **Enter**:

```
admin@firewall-a# find command keyword commit
check full-commit-required
commit description <value> force partial device-and-network <excluded> shared-ob[...]
commit description <value> force partial device-and-network <excluded> shared-ob[...]
commit description <value> force partial device-and-network <excluded> shared-ob[...]
commit description <value> force partial device-and-network <excluded> shared-ob[...]
commit description <value> partial device-and-network <excluded> shared-object <[...]
commit description <value> partial device-and-network <excluded> shared-object <[...]
commit description <value> partial device-and-network <excluded> shared-object <[...]
commit description <value> partial device-and-network <excluded> shared-object <[...]
show shared admin-role <name> role device webui commit
set deviceconfig setting management auto-acquire-commit-lock <yes|no>
set deviceconfig setting management disable-commit-recovery <yes|no>
set deviceconfig setting management commit-recovery-retry <1-5>
set deviceconfig setting management commit-recovery-timeout <3-30>
set deviceconfig setting management rule-fail-commit <yes|no>
set network interface ethernet <name> layer3 ipv6 dhcp-client v6-options rapid-c[...]
```

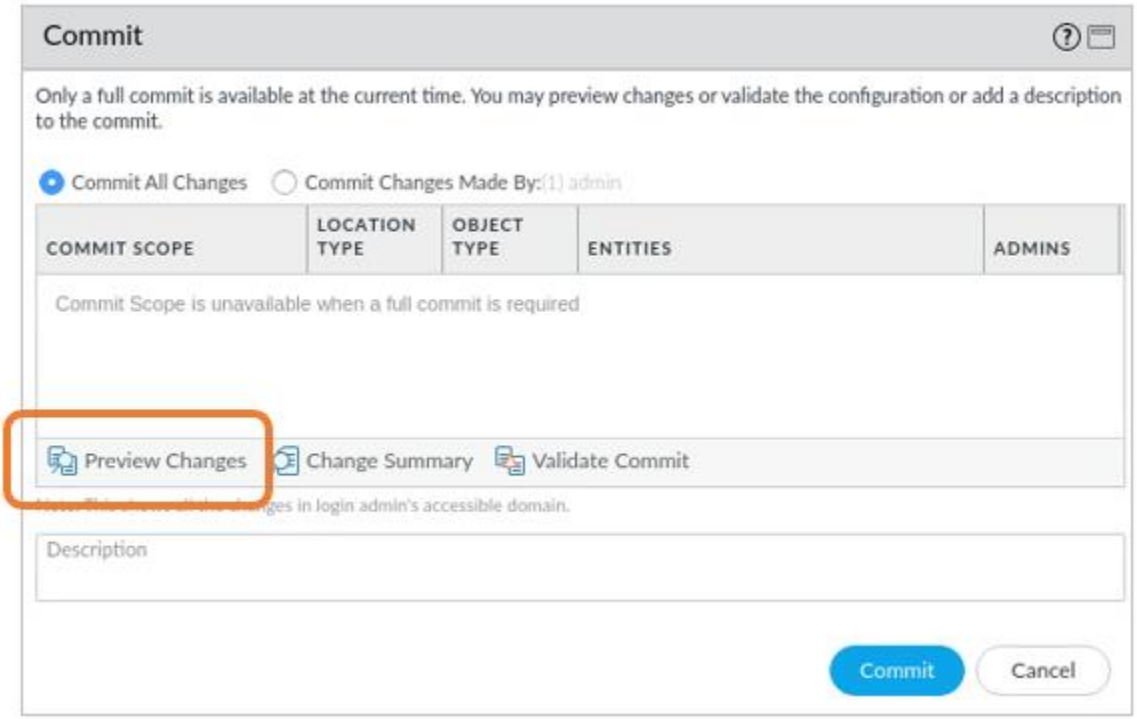
```
set network interface ethernet <name> layer3 units <name> ipv6 dhcp-client v6-op[...]
set network interface vlan ipv6 dhcp-client v6-options rapid-commit <yes|no>
set network interface vlan units <name> ipv6 dhcp-client v6-options rapid-commit[...]
set shared admin-role <name> role device webui commit
set shared admin-role <name> role device webui commit device <enable|disable>
set shared admin-role <name> role device webui commit commit-for-other-admins >[...]
set shared admin-role <name> role device webui commit object-level-changes <ena[...]
set shared admin-role <name> role device xmlapi commit <enable|disable>
[edit]
```

All but three commands in the output are concerned with setting or showing *permissions* to perform a commit operation as defined by a named admin *role*. We will focus on the options that start with "commit."

You now know that **`find command keyword <string>`** output displays the full possibilities for a command along a given path of options. When one option excludes another option, **`find command`** will list the new path of command options recursively until they are exhausted. Consequently, the **`find command`** output can appear redundant.

However, you also have seen how the **`find command`** output can be too concise. Remember that many or even all options listed after a certain parameter *may not be required*. For example, the **`commit`** command can be executed with no options, but the **`find command`** alone will not show this possibility as valid.

Note: The options that the CLI provides for starting commit jobs enable you to perform most of the same functions that you can perform using the web interface, including the ability to add a description and to execute partial commits when a full commit is not required. However, the CLI does not provide a **Preview Changes** option, which the web interface does:



24. Type:
commit, press the **spacebar**, and then press **Tab**.

This procedure shows you the options available for the **commit** command:

```
admin@firewall-a# commit [Tab]
+ description   Enter commit description
> force         force
> partial       partial
  <Enter>       Finish input
```

Note that **<Enter>** is listed as a valid option.

An attempt to submit a *partial commit* when a full commit is required will result in an error. The error message will state that a partial commit is not allowed. A partial commit on firewalls running multiple virtual systems enables you to limit the commit of the configuration to a specific virtual system only.

25. Type:
commit partial device-and-network excluded and press **Enter**.

This command attempts a *partial* commit:

```
admin@firewall-a# commit partial device-and-network excluded

Server error : Partial commit is not allowed. Full commit must be completed.
[edit]
```

26. Type:
check full-commit-required and press **Enter**.
This command tells you if a full commit is required:

```
admin@firewall-a# check full-commit-required

yes
[edit]
```

27. Type:
validate full and press **Enter**.

This command checks the candidate configuration as valid to become the running configuration.

Note: The **jobid** number displayed in your lab likely will be different from the following output:

```
admin@firewall-a# validate full

Validate job enqueued with jobid 3252
3252

[edit]
```

In operational mode, you can use the **show jobs id <number>** command to check the results of the validation.

In configuration mode, you can execute show commands that are unique to operational mode by adding **run** to the beginning of the command.

28. Type:
    run show jobs id <number>
    For the number, use the unique job ID displayed in your lab environment.
    If the job is not finished, rerun the command to monitor progress and check the results:

```
admin@firewall-a# run show jobs id 3252

Enqueued              Dequeued       ID          Type       Status Result Completed
------------------------------------------------------------------------------------
2023/04/21 08:55:18   08:55:18     3252       Validate         FIN     OK 08:55:27
Warnings:
Details:Configuration is valid

[edit]
```

29. Type:
    commit and press **Enter**:

```
admin@firewall-a# commit

Commit job 3277 is in progress. Use Ctrl+C to return to command prompt
...70%...98%........100%
Configuration committed successfully

[edit]
```

    Note: Wait until the commit has successfully completed before continuing.

30. Type:
    check pending-changes and press **Enter**.
    This command checks for any *uncommitted* changes to the candidate configuration:

```
admin@firewall-a# check pending-changes

no
[edit]
```

31. In the web interface, go to **Monitor > Logs > Configuration**.
    Look for the configuration activity that relates to the changes you have made in the CLI. Notice the correlation of the **Client**, **Command**, and **Result** columns with the actions you have taken:

RECEIVE TIME	ADMINISTRATOR	HOST	CLIENT	COMMAND	RESULT
07/08 00:09:03	admin	192.168.1.20	CLI	commit	Submitted
07/08 00:07:41	admin	192.168.1.20	CLI	commit	Failed
07/07 23:08:40	admin	192.168.1.20	CLI	commit	Submitted
07/07 23:01:38	admin	192.168.1.20	Web	commit	Submitted
07/07 22:11:20	admin	192.168.1.20	Web	edit	Succeeded
07/07 22:11:19	admin	192.168.1.20	Web	override	Failed
07/07 21:22:20	admin	192.168.1.20	Web	edit	Succeeded

Your specific log file entries may differ from the example.

32. Leave your web browser open.

1.3.2 Confirm the Current Device Configuration

You have discovered that the DNS setting for the management interface of the firewall is configured for *external* sources, which are **4.2.2.2** and **8.8.8.8**. This configuration appears to cause problems for FQDN Address objects and for Kerberos-related authentication profiles.

Change the DNS setting for the device configuration of the firewall to use **192.168.50.53** only.

33. Click to display the **Remmina** window that contains the CLI for the firewall and type: **`run ping host client-a.panw.lab`** and press **Enter**.

    This command tests if the management interface can resolve internal DNS names properly.

```
admin@firewall-a# run ping host client-a.panw.lab [Enter]
ping: client-a.panw.lab: System error
[edit]
```

    Remember that in configuration mode you can execute commands that are unique to operational mode by adding **`run`** to the beginning of the command.

    If the command prompt from the previous section is still active, you already should be in configuration mode. Verify that that command prompt sign is "**#**". If it is not, use the command **`configure`** to enter configuration mode.

34. Type:
    `find command keyword dns` and press **Enter**.

```
show deviceconfig system dns-setting
show deviceconfig system dns-setting
show deviceconfig system dns-setting servers
show deviceconfig setting dns-over-https
show network interface ethernet <name> layer3 ipv6 neighbor-discovery router-adv[...]
show network interface ethernet <name> layer3 ipv6 neighbor-discovery router-adv[...]
[. . .]
show network dns-proxy
show network dns-proxy <name>
[. . .]
set deviceconfig system dns-setting servers primary <ip/netmask>
set deviceconfig system dns-setting servers secondary <ip/netmask> [. . .]
[edit]
```

Focus your attention on the first few **show** commands. The text string "dns-setting" appears to provide a better focus on the commands that we will need.

Press the **spacebar** as needed, or type **q**, to return to the command prompt.

35. Type:
`find command keyword dns-setting` and press **Enter**.

This query produces a shorter list of commands.

```
admin@firewall-a# find command keyword dns-setting [Enter]
show deviceconfig system dns-setting
show deviceconfig system dns-setting
show deviceconfig system dns-setting servers
set deviceconfig system dns-setting
set deviceconfig system dns-setting
set deviceconfig system dns-setting servers primary <ip/netmask>
set deviceconfig system dns-setting servers secondary <ip/netmask>
set deviceconfig system dns-setting dns-proxy-object <value>
[edit]
```

36. Type:
`show deviceconfig system dns-setting` and press **Enter**.

This command shows the current DNS settings and helps to confirm that these are the settings that you need to change.

```
admin@firewall-a# show deviceconfig system dns-setting [Enter]
dns-setting {
  servers {
    primary 4.2.2.2;
    secondary 8.8.8.8;
  }
}
[edit]
```

1.3.3 Explore Options for Changing Other Device Settings

37. Type:
`edit deviceconfig` and press **Enter**.

This command changes the working context in the configuration hierarchy.

```
admin@firewall-a# edit deviceconfig [Enter]
[edit deviceconfig]
```

38. Type:
    ed[Tab][Tab]

    This procedure displays additional options for commands run from the current working context.

```
admin@firewall-a# edit [Tab]
  cluster             cluster
  high-availability   high-availability configuration
  plugins             plugins
  setting             setting
  system              system
  <Enter>             Finish input
```

    Notice that the branches at this level in the configuration hierarchy do not correspond closely to the navigation menu nor tabbed **Device > Setup** configuration pages in the web interface.

39. Type:
    edit sett[Tab] and press **Enter**.

    This command changes the working context to **[edit deviceconfig setting]**.

```
admin@firewall-a# edit setting [Enter]
[edit deviceconfig setting]
```

40. Type:
    sh[Tab][Tab]

```
admin@firewall-a# show [Tab]
  application              application
  autofocus                autofocus
  captive-portal           captive-portal
  cloud-userid             cloud-userid
  cloudapp                 cloudapp
  config                   config
  ctd                      ctd
  custom-logo              custom-logo
  dhcp-syslog-server       Syslog DHCP log collector
  dns-over-https           dns-over-https
  global-protect           global-protect
  hawkeye                  DLP cloud setting
  http2                    http2
  icmpv6-rate-limit        icmpv6-rate-limit
  inline-spyware-setting   inline-spyware-setting
  inline-url-setting       inline-url-setting
  inline-wf-setting        inline-wf-setting
  iot                      iot
  jumbo-frame              jumbo-frame
  l3-service               l3-service
  logging                  logging
[. . .]
```

Review the options listed and make basic mental correlations with the settings that you are already familiar with on the **Device** pages of the web interface.

41. Press **Alt+Backspace** to delete the command-line text and type:
    se[Tab][Tab]

    This procedure displays the options available for the **set** command.

```
admin@firewall-a# set [Tab]
+ advance-routing          Enable Advanced Routing Module
+ auto-mac-detect          Using detected VM interface MAC as PANOS interface MAC
+ gtp                      gtp
+ mobile-security-policy   mobile-security-policy
+ net-inspection           enable net inspection
+ preserve-prenat-feature  enable preserve-prenat feature
+ sctp                     sctp
+ tunnel-acceleration      to accelerate GTP-U, GRE and VxLAN traffic
> application              application
> autofocus                autofocus
> captive-portal           captive-portal
> cloud-userid             cloud-userid
> cloudapp                 cloudapp
> config                   config
[. . .]
```

    Notice that several configuration options are displayed in addition to available branches of the configuration hierarchy that are available to the **show** command.

42. Press **Alt+Backspace** to delete the command-line text and type:
    sh[Tab] ses[Tab][Tab]

    This procedure displays the options available for the **show** command, plus the **session** option.

```
admin@firewall-a# show session [Tab]
  |        Pipe through a command
  <Enter>  Finish input
```

    No options other than | and <Enter> are listed.

43. Press **Enter**.

```
admin@firewall-a# show session [Enter]
[edit deviceconfig setting]
```

    No settings are shown. In many cases, the **show** command options are available only for those settings that have been configured with values different from the system default(s) relative to the applicable context of the configuration hierarchy.

44. Type:
    set sess[Tab][Tab]

    This procedure displays options available for changing the systems session settings. Notice that *all* options listed are marked with a "+" sign to indicate that they are optional.

```
admin@firewall-a# set session [Tab]
```

```
+ accelerated-aging-enable                enable/disable accelerated session aging
+ accelerated-aging-scaling-factor        set accelerated session aging scaling[...]
+ accelerated-aging-threshold             set accelerated aging threshold in pe[...]
+ dhcp-bcast-session-on                   enable/disable session setup for DHCP[...]
+ erspan                                  enable/disable ERSPAN
+ icmp-unreachable-rate                   set maximum number of ICMP unreachabl[...]
+ ipv6-firewalling                        enable/disable IPv6 firewalling
+ max-pending-mcast-pkts-per-session      Max number of multicast packets queue[...]
+ multicast-route-setup-buffering         enable/disable multicast packet queue[...]
+ offload                                 enable/disable hardware session offloading
+ packet-buffer-protection-activate       percentage of packet buffer utilization
+ packet-buffer-protection-alert          percentage of packet buffer utilization
[. . .]
```

Most, but not all, of these session options are configurable in the web interface on the **Device > Setup > Session** page. Session options include parameters that you do not often need to change. Examples of times when you may need to change these settings include the need to troubleshoot traffic that is, or may be, offloaded to a network processor or the need to optimize session settings for traffic that includes abnormal delays or that is highly fragmented.

45. Press **Alt+Backspace** *twice* to delete the command-line text and then type:
`top` and press **Enter**.

    This command moves the working context to the top of the configuration hierarchy.

```
admin@firewall-a# top [Enter]
[edit]
```

Verify the working context is now **`[edit]`**.

1.3.4 Change the Current Device Configuration

46. Type:
`edit deviceconfig system dns-setting servers` and press **Enter**.

    This command changes the working context to the level that is closest to the settings that you need to modify.

```
admin@firewall-a# edit deviceconfig system dns-setting servers [Enter]
[edit deviceconfig system dns-setting servers]
```

Note: In subsequent steps, practice any methods that you now know, such as the use of the Tab key, arrow keys, and various key combinations, to accomplish the steps specified. Use these methods to save time or to explore the available options. Instructions that call explicit attention to opportunities to use these methods will be used less frequently than in prior steps.

47. Type:
`show` and press **Enter**.

```
admin@firewall-a# show [Enter]
servers {
  primary 4.2.2.2;
  secondary 8.8.8.8;
```

```
}
[edit deviceconfig system dns-setting servers]
```

Confirm that configuration parameters to be changed are displayed to verify that the working context of the command prompt is correct.

48. Type:
    `set [Tab]`

    Habitual use of the Tab key displays the availability and spelling of the next option or value required and, thereby, helps you to avoid errors.

```
admin@firewall-a# set [Tab]
+ primary     Primary DNS server IP address
+ secondary   Secondary DNS server IP address
  <Enter>     Finish input
```

49. Type:
    `set primary 192.168.50.53` and press **Enter**.

```
admin@firewall-a# set primary 192.168.50.53 [Enter]

[edit deviceconfig system dns-setting servers]
```

50. Type:
    `show` and press **Enter**.

```
admin@firewall-a# show [Enter]
servers {
  primary 192.168.50.53;
  secondary 8.8.8.8;
}
[edit deviceconfig system dns-setting servers]
```

51. Type:
    `commit` and press **Enter**.

```
admin@firewall-a# commit [Enter]

Commit job 25265 is in progress. Use Ctrl+C to return to command prompt
...55%...75%...99%......100%
Configuration committed successfully
```

52. Type:
    `top` and press **Enter**.

```
[edit deviceconfig system dns-setting servers]
admin@firewall-a# top [Enter]
[edit]
```

53. Type:
    `exit` and press **Enter**.

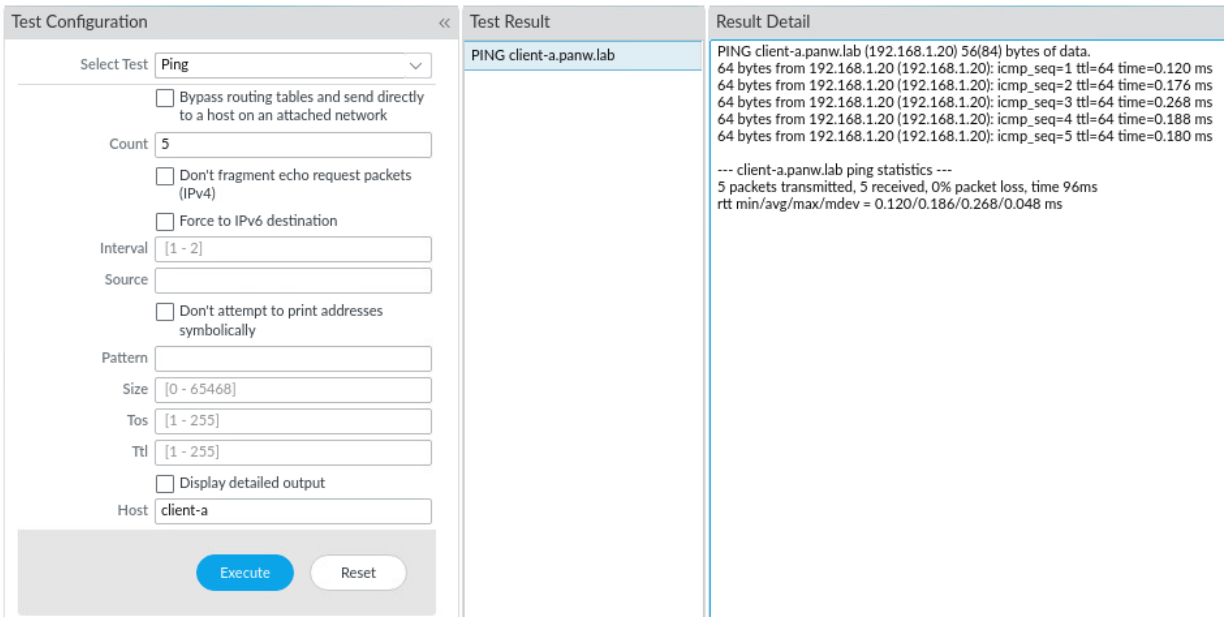
```
admin@firewall-a# exit [Enter]
Exiting configuration mode
```

54. Type:
    `ping host client-a` and press **Enter**.
55. After three or more ping results are displayed, press **Ctrl+C**:

```
admin@firewall-a> ping host client-a [Enter]
PING client-a.panw.lab (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=128 time=0.068 ms
64 bytes from 192.168.1.20: icmp_seq=2 ttl=128 time=0.192 ms
64 bytes from 192.168.1.20: icmp_seq=3 ttl=128 time=0.181 ms
64 bytes from 192.168.1.20: icmp_seq=4 ttl=128 time=0.202 ms
^C
--- client-a.panw.lab ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2998ms
rtt min/avg/max/mdev = 0.068/0.160/0.202/0.056 ms
```

Notice in the heading for the statistics summary that the domain name **panw.lab** has been appended automatically to the hostname. The firewall is configured with the correct domain name for the DNS lookup.

56. (Optional) Use the CLI to ping the FQDN **`client-a.panw.lab`**.
    Based on the results of the prior use of ping, you can infer that the firewall is configured with the correct domain name for the DNS lookup. If the domain-name configuration for the firewall needed to be changed, where in the CLI would you go to change it?
57. Use the CLI to ping an external domain name such as www.paloaltonetworks.com to verify the DNS resolution for external sites.
58. Go to the web interface and use the **Device** > **Troubleshooting** page to ping **`client-a`** and review the results:



After you select **Ping** for the test type, add the name or IP address of the **Host** to ping (**`client-a`**), and click **Execute**, an item (**PING <destination>**) will appear in the middle column after the test is complete. You then must click the results item in the **Test Result** column to display the detailed results of the test in the **Result Detail** column.

1.3.5 Clean Up Your Lab Environment

59. Close all open tabs and windows for the configuration browser.

    Leave open the **configuration** browser and the connection to the web interface of the firewall.

60. If you have an open CLI connection to the firewall, leave the connection open for the next lab.

Stop. This is the end of the lab.

1.4. Lab (Optional): Use the CLI to Modify Policy Objects

Lab Objectives

- Correlate available system status information in the CLI with the same or similar information as displayed by the firewall in the web interface
- Use the CLI to make one or more policy modifications and commit changes
- Verify CLI changes using the web interface

Lab Scenario

In this lab, you will review the current configuration in the web interface and then use the CLI to display various elements of the same configuration.

Subsequent lab steps will demonstrate the procedures required to:

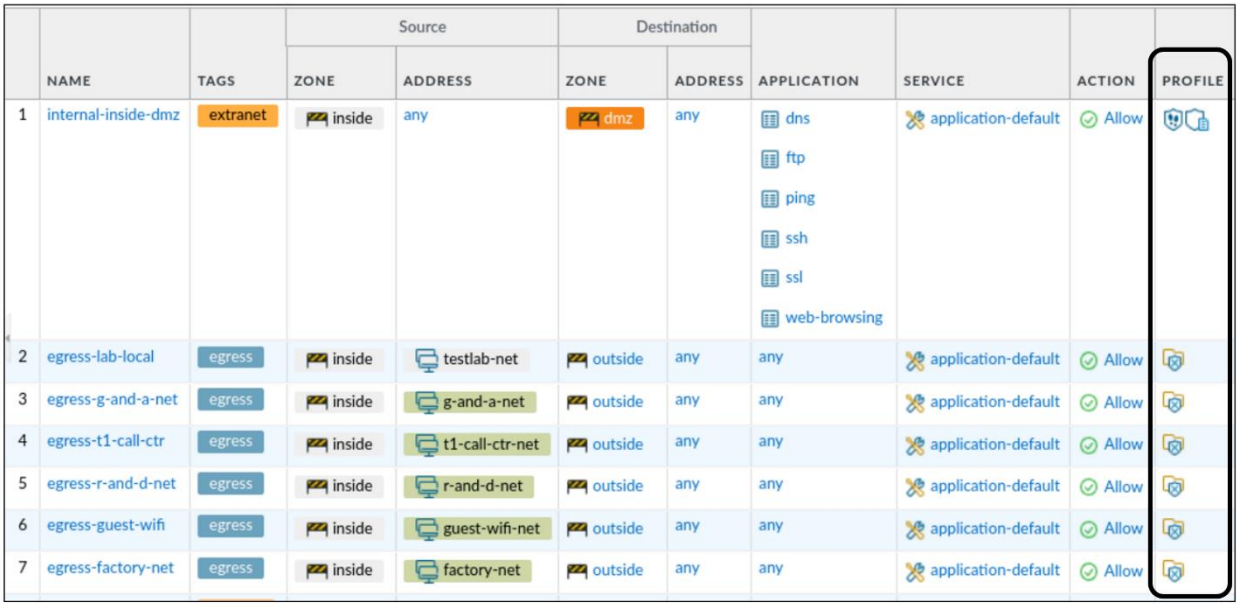
- Display configuration data
- Change the running configuration
- Commit the candidate configuration

After you make configuration changes by using the CLI, you will see how your changes are reflected in the web interface.

Correlation of the information available in the web interface with the way that configuration and status information is displayed in the CLI likely will help you better understand how the CLI works and how you can use it.

1.4.1 Review the Existing Policy Configuration

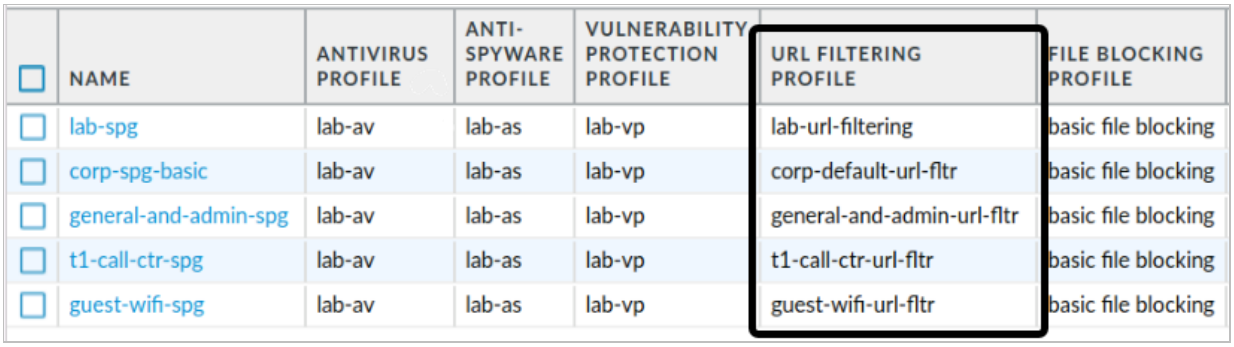
1. Click **Policies > Security**:



	NAME	TAGS	Source ZONE	Source ADDRESS	Destination ZONE	Destination ADDRESS	APPLICATION	SERVICE	ACTION	PROFILE
1	internal-inside-dmz	extranet	inside	any	dmz	any	dns ftp ping ssh ssl web-browsing	application-default	Allow	
2	egress-lab-local	egress	inside	testlab-net	outside	any	any	application-default	Allow	
3	egress-g-and-a-net	egress	inside	g-and-a-net	outside	any	any	application-default	Allow	
4	egress-t1-call-ctr	egress	inside	t1-call-ctr-net	outside	any	any	application-default	Allow	
5	egress-r-and-d-net	egress	inside	r-and-d-net	outside	any	any	application-default	Allow	
6	egress-guest-wifi	egress	inside	guest-wifi-net	outside	any	any	application-default	Allow	
7	egress-factory-net	egress	inside	factory-net	outside	any	any	application-default	Allow	

Policy rules 2 through 7 enable external access for a series of subnets. Notice that each rule includes a Security profile configuration that references a Security profile group.

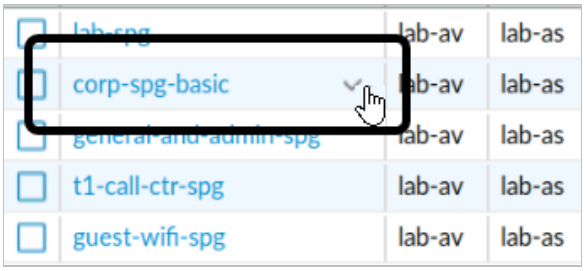
2. Click **Objects > Security Profile Groups**:



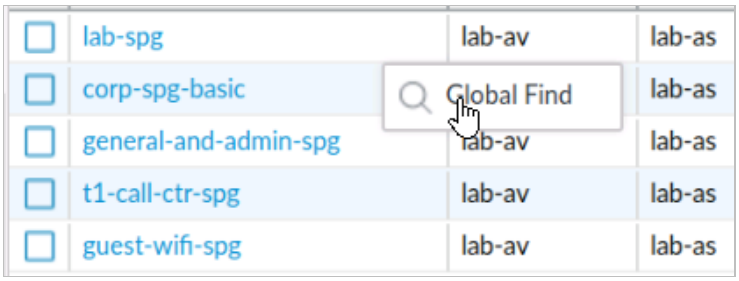
NAME	ANTIVIRUS PROFILE	ANTI-SPYWARE PROFILE	VULNERABILITY PROTECTION PROFILE	URL FILTERING PROFILE	FILE BLOCKING PROFILE
lab-spg	lab-av	lab-as	lab-vp	lab-url-filtering	basic file blocking
corp-spg-basic	lab-av	lab-as	lab-vp	corp-default-url-fltr	basic file blocking
general-and-admin-spg	lab-av	lab-as	lab-vp	general-and-admin-url-fltr	basic file blocking
t1-call-ctr-spg	lab-av	lab-as	lab-vp	t1-call-ctr-url-fltr	basic file blocking
guest-wifi-spg	lab-av	lab-as	lab-vp	guest-wifi-url-fltr	basic file blocking

Notice that each Security profile group includes a different URL Filtering profile.

Hover your mouse or touchpad pointer over the name of the **corp-spg-basic** group:

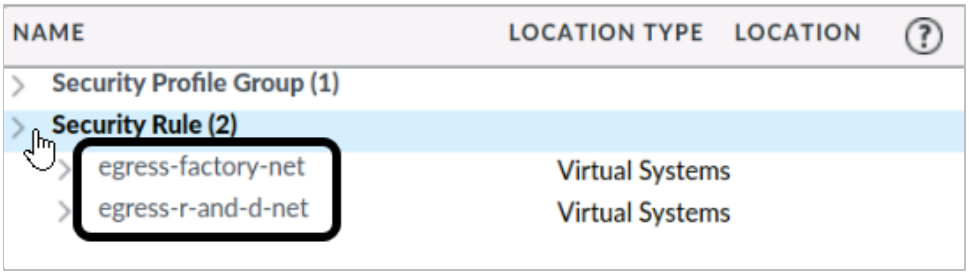


3. Click the **drop-down icon**, then click **Global Find**:



This action will populate the search box in the upper right and after a few moments return Global Find search results for the name of the selected element.

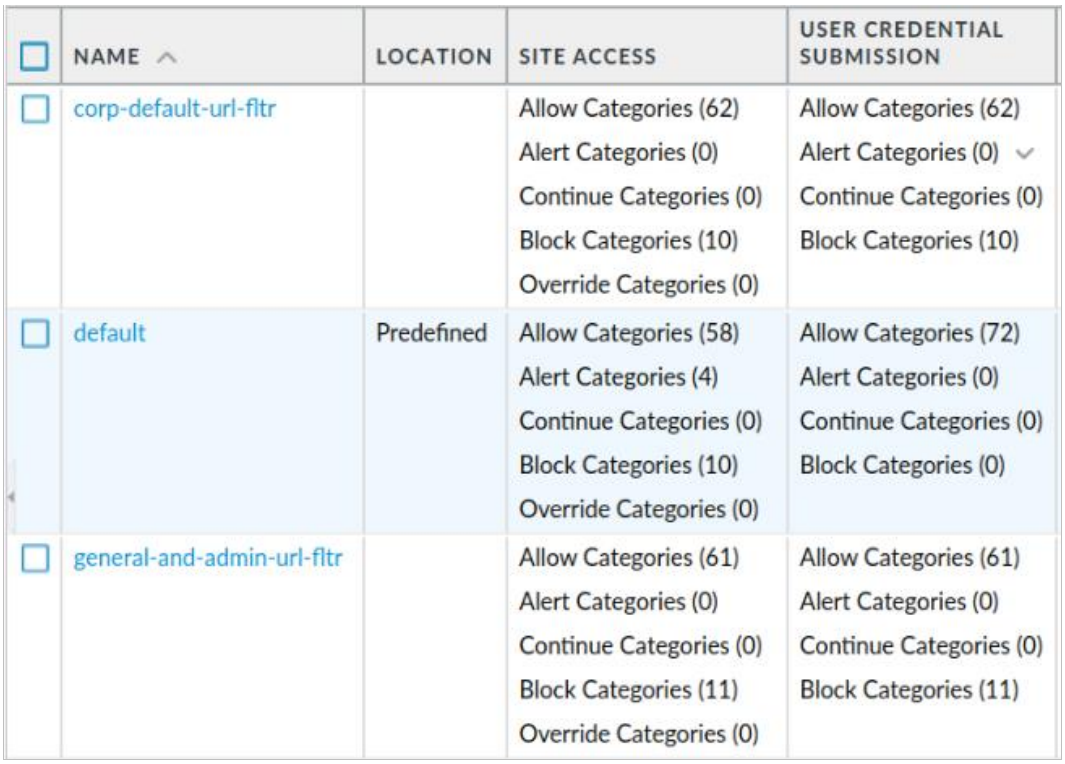
4. Click > to display a list of each **Security Rule** in which the target configuration element is used:



The corp-spg-basic Security profile group is used in two Security rules. The other groups listed on the Security profile groups page also are used in various Security rules.

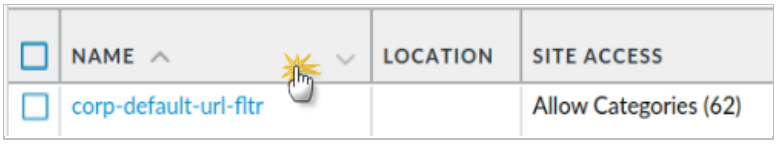
5. (Optional) Repeat the steps required to use the Global Find feature to discover the Security rules in which the other Security profile groups are used.

6. Click **Objects > Security Profiles > URL Filtering**:



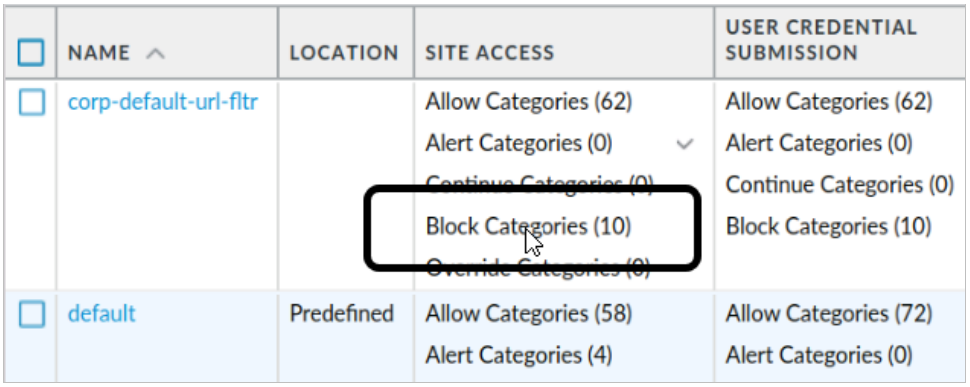
NAME	LOCATION	SITE ACCESS	USER CREDENTIAL SUBMISSION
corp-default-url-fltr		Allow Categories (62) Alert Categories (0) Continue Categories (0) Block Categories (10) Override Categories (0)	Allow Categories (62) Alert Categories (0) Continue Categories (0) Block Categories (10)
default	Predefined	Allow Categories (58) Alert Categories (4) Continue Categories (0) Block Categories (10) Override Categories (0)	Allow Categories (72) Alert Categories (0) Continue Categories (0) Block Categories (0)
general-and-admin-url-fltr		Allow Categories (61) Alert Categories (0) Continue Categories (0) Block Categories (11) Override Categories (0)	Allow Categories (61) Alert Categories (0) Continue Categories (0) Block Categories (11)

7. Click the **Name** column header to sort the table of URL filter configurations by name.



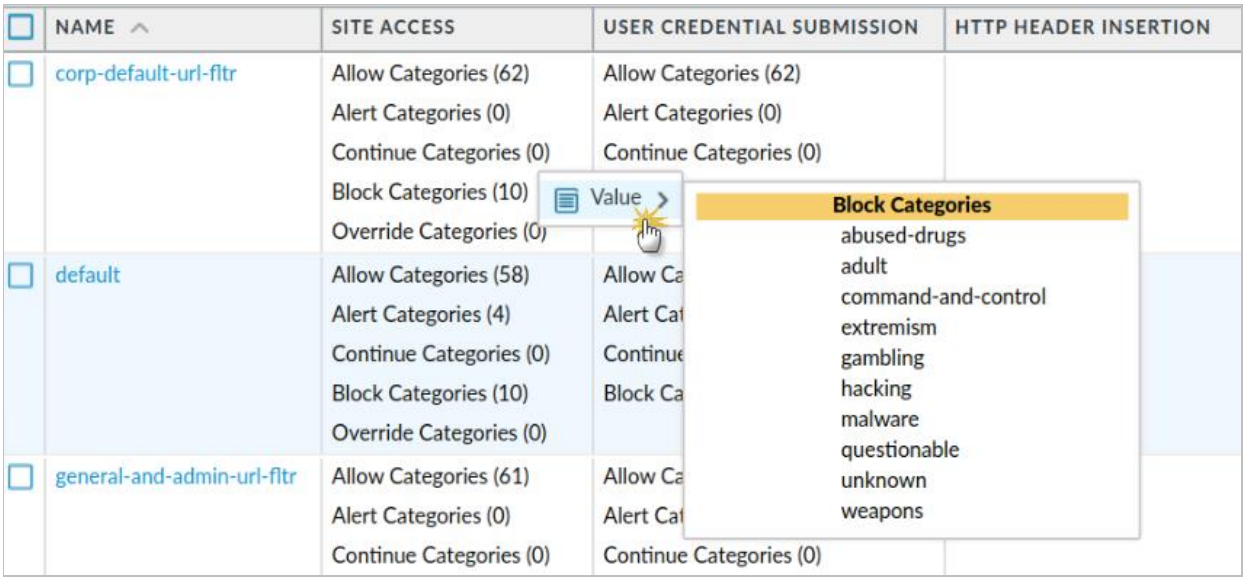
NAME	LOCATION	SITE ACCESS
corp-default-url-fltr		Allow Categories (62)

8. Hover your pointer over the text **Block Categories** in the **Site Access** column of the first URL Filtering profile on the page:



NAME	LOCATION	SITE ACCESS	USER CREDENTIAL SUBMISSION
corp-default-url-fltr		Allow Categories (62) Alert Categories (0) Continue Categories (0) Block Categories (10) Override Categories (0)	Allow Categories (62) Alert Categories (0) Continue Categories (0) Block Categories (10)
default	Predefined	Allow Categories (58) Alert Categories (4)	Allow Categories (72) Alert Categories (0)

9. Click the **drop-down icon**, then hover your mouse pointer over **Value**:



Scan the **Block Categories** listed. Each URL Filtering profile in the configuration includes a unique set of URL categories that are to be blocked.

10. Hover your mouse pointer over the text **Allow Categories** in the **Site Access** column of the first URL Filtering profile on the page, click the **drop-down icon**, then hover your pointer over **Value**. What categories are allowed?
11. Review the **Block Categories** and **Allow Categories** that are specified in one or more of the other URL Filtering profiles.

1.4.2 Use the CLI to Examine a Configuration and Discover Options for How to Modify It

The following steps demonstrate the use of the CLI to make configuration changes to a policy object. Sometimes the solution to a problem requires you to apply the same configuration change across multiple policy rules, objects, or systems. Sometimes you can use the CLI to replicate configuration adjustments more efficiently than you can by using the web interface.

In this scenario, your security team has discovered unwanted traffic that your management team wants the firewall to block. An analysis of the URL Filtering profiles that are used in current Security policy rules reveals that not all URL categories that your organization has established as a best practice to block are included.

You need to set all of your custom URL Filtering profiles to block the following categories: copyright-infringement, dynamic-dns, parked, proxy-avoidance-and-anonymizers, and phishing.

12. Go to the **CLI**. You can use an existing SSH session (via **Remmina**) if you have one open; otherwise launch **Remmina** and connect to Firewall-A.

13. Identify the current command mode indicated by the prompt symbol **#** or **>**.
    If the CLI is in operational mode (**>**), type **`configure`** and press **Enter**.
    The prompt sign should be "**#**".
14. Type:
    `find command keyword filtering` and press **Enter**.
    This command will help you search for likely useful commands related to "filtering."
15. Refine the search by modifying the command. Type:
    `find command keyword url-filtering` and press **Enter**:

```
admin@firewall-a# find command keyword url-filtering
show profiles url-filtering
show profiles url-filtering <name>
[. . .]
set profiles url-filtering
set profiles url-filtering <name>
set profiles url-filtering <name> description <value>
set profiles url-filtering <name> allow  [ <allow1> <allow2>... ]
set profiles url-filtering <name> alert  [ <alert1> <alert2>... ]
set profiles url-filtering <name> block  [ <block1> <block2>... ]
set profiles url-filtering <name> continue  [ <continue1> <continue2>... ]
set profiles url-filtering <name> override  [ <override1> <override2>... ]
[. . .]
```

    Notice that most commands displayed are **`show`** and **`set`** commands for URL Filtering profiles. Look for a command statement that is likely to allow you to change the action for a named URL category. Do the query results show any commands for *copying* or *deleting* an existing profile or other configuration element?

    Answer: No. In configuration mode, the output of **`find command keyword <string>`** does not display options for the **`copy`**, **`delete`**, **`edit`**, **`move`**, or **`rename`** commands. The CLI does provide autocomplete and suggestions for these commands. The syntax for **`copy`**, **`delete`**, and **`rename`** is similar to that of **`show`** and **`set`**, but examples of these additional commands are not included in the output of **`find command`** queries.

    (Optional) Use the command **`find command`** with no additional options, and then use the **`less`** command **/** (slash) and the keyword **copy** (or **delete**, **edit**, **move**, or **rename**) to search through all possible output that the **`find`** command can provide in configuration mode. Press the **spacebar** as needed, or type **q**, to display the rest of the output and return to the command prompt.
16. Type:
    `show profiles url-filtering`, press the **spacebar**, then press **Tab**.
    Or type **sh[Tab] pro[Tab]s[Tab]url[Tab] [Tab]**.
    This procedure displays a list of the names of any existing URL Filtering profiles, along with any other optional or required parameters that are applicable.

```
admin@firewall-a# show profiles url-filtering [Tab]
```

```
  corp-default-url-fltr        corp-default-url-fltr
  general-and-admin-url-fltr   general-and-admin-url-fltr
  guest-wifi-url-fltr          guest-wifi-url-fltr
  lab-url-filtering            lab-url-filtering
  t1-call-ctr-url-fltr         t1-call-ctr-url-fltr
  <name>                       <name>
  |                            Pipe through a command
  <Enter>                      Finish input
```

Note: With the **show** command, if the "<name>" option is listed along with the names of existing configuration elements, you must use the name of an existing element. If you use a name that does not exist, the execution of the command may not produce an error, but it will not produce any data, because the named element does not exist.

Note: With the **show** command, if you press **Enter** when the "<Enter>" option is listed along with the names of multiple existing configuration elements, the command typically will return the available configuration data for *all* elements in the list.

17. Type:
show profiles url-filtering corp[Tab] and press **Enter**:

```
admin@firewall-a# show profiles url-filtering corp-default-url-fltr
corp-default-url-fltr {
  credential-enforcement {
    mode {
      disabled;
    }
    log-severity medium;
    block [ abused-drugs adult command-and-control extremism gambling hacking malware
            questionable unknown weapons];
  }
  block [ abused-drugs adult command-and-control extremism gambling hacking malware
          questionable unknown weapons];
  local-inline-cat yes;
  cloud-inline-cat no;
}
[edit]
```

In configuration mode, the **show** command provides configuration about the candidate configuration, as if the candidate configuration were a saved configuration file (which it may also be). The configuration details displayed typically are limited to the *differences* between the system's *default* settings and the current settings, relative to the level in the configuration hierarchy from which the **show** command was executed and the options used.

The system's default output is a human-readable version of the JSON format, sometimes called Human JSON or Hjson. It is referred to in the configuration options of the CLI as "**default**." Configurable output options also include JSON proper, XML, and "set-command." Subsequent steps in this lab activity will demonstrate how to specify the output format.

Examine the hierarchy of the configuration as displayed by the default output format. Try to correlate what you see in the configuration with what you see in the web interface.

Formulate the best answers you can to the following questions:

- Why are the URL categories that are currently *allowed* not reflected in the CLI output? How do you think the web interface might work to produce this extra information?
- What does the output tell you about the relationship of the configuration file to the default behavior of the firewall? For example, how does the output help explain the fact that you cannot delete the "default" URL Filtering Security profile?
- What does the output tell you about what you need to know to be able to read a configuration file and develop a full mental picture of how the firewall will behave based on what is in the configuration?
- What is the difference between the configuration as the CLI displays it and what you would see in an exported configuration file, if you had one and opened it in a Notepadqq?

If you use the **set** configuration command to change the CLI output format, the CLI will give you a different perspective about how configurations relate to the behavior of the firewall.

18. Type:
run set cli config-output-format set and press **Enter**.

This command *runs* the operational-mode command **cli config-output-format <value>** from the configuration-mode prompt. The value **set** changes the output format to the series of **set** commands that would be required to create the current candidate configuration:

```
admin@firewall-a# run set cli config-output-format set [Enter]
[edit]
```

19. Type:
show profiles url-filtering corp[Tab] and press **Enter**.

```
admin@firewall-a# show profiles url-filtering corp-default-url-fltr [Enter]
set profiles url-filtering corp-default-url-fltr credential-enforcement mode disabled
set profiles url-filtering corp-default-url-fltr credential-enforcement log-severity
      medium
set profiles url-filtering corp-default-url-fltr credential-enforcement block
      [ abused-drugs adult command-and-control extremism gambling hacking malware
      questionable unknown weapons ]
set profiles url-filtering corp-default-url-fltr block [ abused-drugs adult command-
      and-control extremism gambling hacking malware questionable unknown weapons ]
set profiles url-filtering corp-default-url-fltr local-inline-cat yes
set profiles url-filtering corp-default-url-fltr cloud-inline-cat no
[edit]
```

Count the number of commands returned. Now, look at the JSON-formatted output from your prior execution of the command. How many lines in the output do *not* include a curly brace, that is, "**{**" or "**}**"? The number of commands in the set-formatted output should match the number of lines in the JSON-formatted output without curly braces. (A wrapped line counts as one line.)

Notice how each **set** command corresponds to a setting in the JSON-formatted output.

The last line of output after execution of a CLI command shows the location in the configuration hierarchy. Note that you still are at the "**[edit]**" level, which is the top level of the hierarchy.

Note: The syntax of the **set** commands that the CLI displays when the configuration-output mode is set to "set" *is not relative* to the working context of the configuration-mode prompt. This particular absence of relativity to the working context is different than for the other configuration output modes, including "default," "json," and "xml." The benefit of this difference is that you can copy such set commands for later use regardless of context.

20. At the configuration-mode command prompt, with no text entered, press **Tab**:

```
admin@firewall-a# [Tab]
  check      Check configuration status
  commit     Commit current set of changes
  copy       Copy a statement
  delete     Delete a data element
  edit       Edit a sub-element
  exit       Exit from this level
  find       Find CLI commands with keyword
  load       Load configuration from disk
  move       Move a node within an ordered collection
  override   Override a template element
  quit       Quit from this level
  rename     Rename a statement
  revert     Revert changes from configuration
  run        Run an operational-mode command
  save       Save configuration to disk
  set        Set a parameter
  show       Show a parameter
  top        Exit to top level of configuration
  up         Exit one level of configuration
  validate   Validate current set of changes
```

Notice the **copy**, **delete**, and **rename** commands. This lab activity explicitly does not demonstrate the use of these commands. However, after you develop a proficiency in the use of the **set** command, you should take the opportunity to explore and practice the use of these other commands (as appropriate relative to the context and time constraints of the current course). To duplicate or "clone" an existing profile, rule, or other configuration element, you use the **copy** command.

21. Type **edit profiles** and press **Enter**. Then, at the blank prompt, press **Enter**:

```
admin@firewall-a# edit profiles
[edit profiles]
admin@firewall-a# [Enter]
[edit profiles]
```

22. At the blank command prompt, press **Tab**:

```
admin@firewall-a# [Tab]
  check      Check configuration status
  commit     Commit current set of changes
```

```
  copy       Copy a statement
  delete     Delete a data element
  edit       Edit a sub-element
[. . .]
  top        Exit to top level of configuration
  up         Exit one level of configuration
  validate   Validate current set of changes
```

The basic command options listed are the same as for the prior **[edit]** level of the hierarchy. However, subsequent suggestions for each of these basic commands (**check**, **commit**, **copy**, **delete**, **edit**, and so on) will be relative to your location in the configuration hierarchy.

To know where you are in the configuration hierarchy, clear any existing command text from the configuration-mode prompt and press **Enter**.

23. Type **edit**, press the **spacebar**, then press **Tab**:

```
admin@firewall-a# edit [Tab]
  custom-url-category          custom-url-category
  data-filtering               data-filtering
  data-objects                 data-objects
  decryption                   decryption
  dos-protection               dos protection profile
  file-blocking                file-blocking
  gtp                          gtp
  hip-objects                  hip-objects
  hip-profiles                 hip profiles
  sctp                         sctp
  sdwan-error-correction       sdwan error correction profile
  sdwan-path-quality           sdwan path quality profile
  sdwan-saas-quality           sdwan saas quality profile
  sdwan-traffic-distribution   sdwan traffic distribution profile
  spyware                      spyware
  url-filtering                url-filtering
  virus                        virus
  vulnerability                vulnerability
  wildfire-analysis            wildfire-analysis
  <Enter>                      Finish input
```

You are presented with options for the **edit** command relative to the current level in the configuration hierarchy.

Note: To move up a level in the configuration hierarchy, type **up** and press **Enter**. To move down in the hierarchy, use the **edit** command followed by a valid next-level configuration parameter.

24. Add **url-filtering** to the **edit** command and press **Enter**:

```
admin@firewall-a# edit url-filtering [Enter]

Invalid syntax.
[edit profiles]
```

Note that the error “Invalid syntax” is returned.

25. Retype:
`edit url-filtering` (or press the **Up Arrow**), add a **space** after the command if you did not use autocomplete, and press **Tab**.

This procedure will display the available options and indicate why the syntax was invalid:

```
admin@firewall-a# edit url-filtering [Tab]
  corp-default-url-fltr        corp-default-url-fltr
  general-and-admin-url-fltr   general-and-admin-url-fltr
  guest-wifi-url-fltr          guest-wifi-url-fltr
  lab-url-filtering            lab-url-filtering
  t1-call-ctr-url-fltr         t1-call-ctr-url-fltr
  <name>                       <name>
```

Notice that **`<Enter>`** is not an option. Thus, you received the earlier "Invalid syntax" error message. You likely will recognize the names of the existing profiles listed from your prior review of the configuration in the web interface.

Note: In configuration mode, when **`<name>`** is listed as a valid option, typically you can create a new element by typing a new name for the desired element. With the **`edit`** command, the new element will not be created until you set at least one configurable parameter. For all unconfigured parameters, the firewall automatically will apply any associated default configuration options and parameters to the new element.

26. Add **`corp-default-url-fltr`** to the **`edit`** command and press **Tab**.

Notice that **`<Enter>`** is now presented as an option.

```
admin@firewall-a# edit url-filtering corp-default-url-fltr [Tab]
  credential-enforcement          credential enforcement settings
  http-header-insertion           http-header-insertion
  <Enter>                          Finish input
```

27. Press **Enter**:

```
admin@firewall-a# edit url-filtering corp-default-url-fltr [Enter]
[edit profiles url-filtering corp-default-url-fltr]
```

Note your new location in the configuration hierarchy. If you were to type commands such as **`copy`**, **`edit`**, **`set`**, and **`show`**, followed by a **space**, and then press **Tab**, the CLI would display valid options relative to this current location.

28. Type **`show`** and press **Enter**:

```
admin@firewall-a# show [Enter]
set profiles url-filtering corp-default-url-fltr credential-enforcement mode disabled
set profiles url-filtering corp-default-url-fltr credential-enforcement log-severity
     medium
set profiles url-filtering corp-default-url-fltr credential-enforcement block
     [ abused-drugs adult command-and-control extremism gambling hacking malware
     questionable unknown weapons ]
```

```
set profiles url-filtering corp-default-url-fltr block
     [ abused-drugs adult command-and-control extremism gambling hacking malware
     questionable unknown weapons ]
set profiles url-filtering corp-default-url-fltr local-inline-cat yes
set profiles url-filtering corp-default-url-fltr cloud-inline-cat no
[edit profiles url-filtering corp-default-url-fltr]
```

This is the same output you earlier saw after executing the command **show profiles url-filtering corp-default-url-fltr** from the top of the command hierarchy.

1.4.3 Modify Object Parameters

The following steps will demonstrate the use of autocomplete, hierarchical-navigation, and copy-and-paste to save time when you need to implement repetitive configuration changes.

Your task is to set each of the existing URL Filtering profiles to block these URL categories:

- copyright-infringement
- dynamic-dns
- parked
- proxy-avoidance-and-anonymizers
- phishing

In the output from the previous command, scan the categories in the "corp-default-url-fltr" profile and confirm that the categories to add to the configuration are not included in the list that is set to block.

29. Type:
`set`, press the **spacebar**, and then press **Tab**:

```
[edit profiles url-filtering corp-default-url-fltr]
admin@firewall-a# set [Tab]
+ cloud-inline-cat          Enable cloud inline categorization
+ description              description
[. . .]
> alert                    categories to alert on
> allow                    categories to allow
> block                    categories to block
> continue                 categories to block/continue
[. . .]
  <Enter>                  Finish input
```

Among the available options, confirm that **`block`** is listed.

30. Type:
`blo[Tab][Tab]`.

This procedure produces a list of values that are valid for the current command string. Notice the first option, the square open bracket:

```
admin@firewall-a# set block [Tab]
  [                                  Start a list of values.
```

```
  abortion                          pan-url-categories abortion
  alcohol-and-tobacco               pan-url-categories alcohol-and-tobacco
  artificial intelligence           pan-url-categories artificial-intelligence
  auctions                          pan-url-categories auctions
  business-and-economy              pan-url-categories business-and-economy
  computer-and-internet-info        pan-url-categories computer-and-internet-info
  content-delivery-networks         pan-url-categories content-delivery-networks
  copyright-infringement            pan-url-categories copyright-infringement
  cryptocurrency                    pan-url-categories cryptocurrence
  dating                            pan-url-categories dating
  dynamic-dns                       pan-url-categories dynamic-dns
  educational-institutions          pan-url-categories educational-institutions
  encrypted-dns                     pan-url-categories encrypted-dns
  entertainment-and-arts            pan-url-categories entertainment-and-arts
  financial-services                pan-url-categories financial-services
  games                             pan-url-categories games
--more--
```

Press the **spacebar** as needed to display the rest of the output and return to the command prompt. You may notice, for example, that "command-and-control" already is included in the current block statement and, therefore, is not included in the list of valid parameters displayed.

31. Type:
    [and press [**Tab**][**Tab**].
    Notice the first option, the square close bracket:

```
admin@firewall-a# set block [ [Tab][Tab]
  ]                                 End a list of values.
  abortion                          pan-url-categories abortion
  alcohol-and-tobacco               pan-url-categories alcohol-and-tobacco
  artificial intelligence           pan-url-categories artificial-intelligence
  auctions                          pan-url-categories auctions
  business-and-economy              pan-url-categories business-and-economy
[. . . ]
```

You do not want to end your list yet. You want to add to it. Press the **spacebar** as needed, or type **q**, to display the rest of the output and return to the command prompt.

The list of categories to add is: **copyright-infringement**, **dynamic-dns**, **parked**, **proxy-avoidance-and-anonymizers**, and **phishing**.

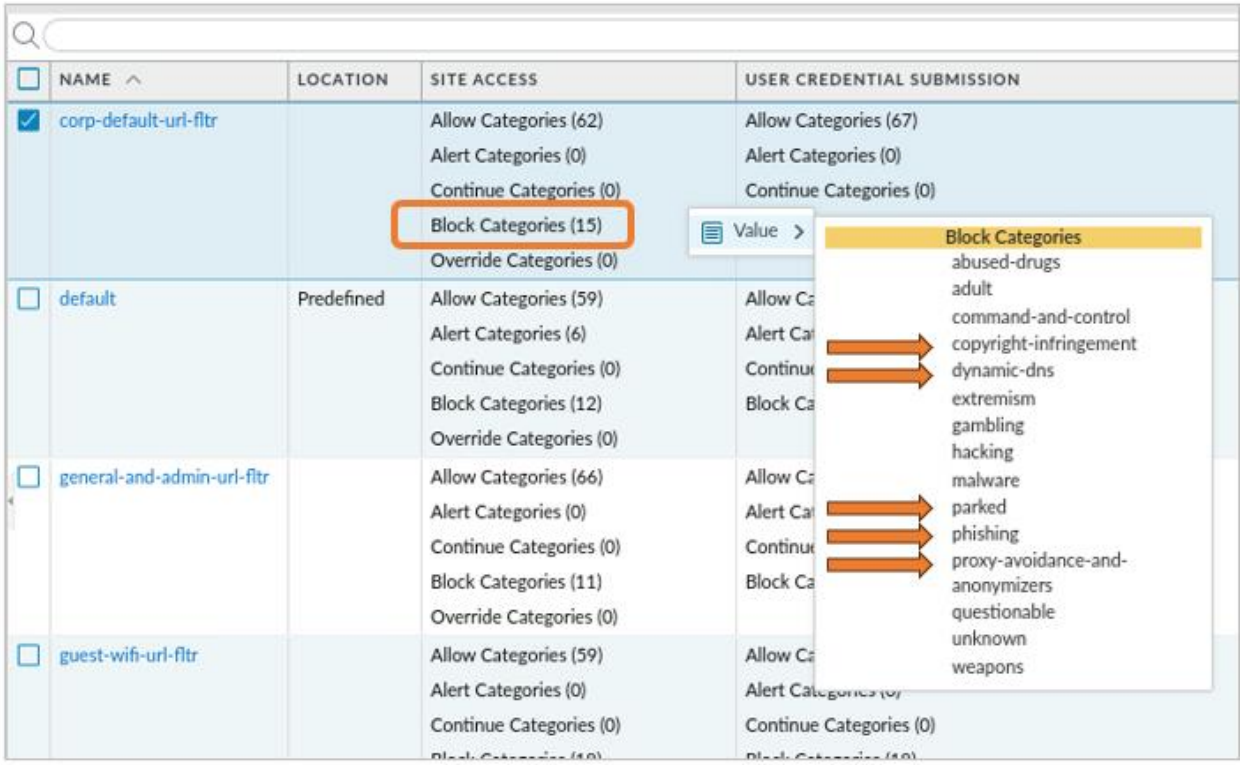
32. Type the *first few letters* of each category as shown below and press **Tab** until you have added all the categories in the list. Then, type a square close bracket ("**]**") and press **Enter**:
    cop[Tab] dyn[Tab] par[Tab] pro[Tab] phis[Tab]].

```
admin@firewall-a# set block [ copyright-infringement dynamic-dns parked proxy-
avoidance-and-anonymizers phishing ]

[edit profiles url-filtering corp-default-url-fltr]
```

Note: This command will not *replace* the summarized **`set`** command displayed by the **`show`** command demonstrated previously. The command will set only the specified categories to "block," *in addition* to any categories that already are set to "block."

33. Go to the web interface and click **Objects > Security Profiles > URL Filtering** to refresh the page.

    Confirm that the number of **Block Categories** for **Site Access** has changed from 10 to 15.



    (Optional) Click **Value** to review the list of categories to be blocked and confirm that the categories that you just added by use of the CLI are included.

34. Go back to the CLI.

    In the following steps, you will navigate the configuration hierarchy and use the copy-and-paste function of Remmina to add the same five categories to the other URL Filtering profiles (excluding the default profile, which cannot be edited).

35. Type:
    `up` and press **Enter**.

    This command will take you to the [edit profiles] level in the configuration hierarchy.

36. Type:
    `edit url[Tab][Tab]`.

    This procedure displays the names of the existing URL Filtering profiles.

37. Type:
    `gen[Tab]` and press **Enter**.

Use of the **Tab** key autocompletes the long name of the next profile that you need to edit, and the execution of the command changes the working context of the CLI to the level in the configuration hierarchy that will be required to edit the profile with the same **set** command that you used previously to edit the "corp-default-url-fltr" profile.

38. Confirm the working level is the following:
    [edit url-filtering general-and-admin-url-fltr]
39. Use your pointer to select the **set block** command that you used to modify the configuration of the "corp-default-url-fltr" profile.
    Scroll up through the Remmina terminal window as required.

```
admin@firewall-a# set block [ copyright-infringement dynamic-dns parked proxy-
avoidance-and-anonymizers phishing ]
```

    Select from the "**s**" in "set" to the square bracket "**]**" that ends the command. Whenever you select text inside the Remmina window, you must right-click to copy and paste it.
40. **Right-click** inside the Remmina window and select paste.
    This procedure inserts the content of the clipboard at the current location of the cursor.
41. Press **Enter**.
    If there is no error, the CLI returns to the prompt. No additional feedback is provided.

```
admin@firewall-a# set block [ copyright-infringement dynamic-dns parked proxy-
avoidance-and-anonymizers phishing ]

[edit profiles url-filtering general-and-admin-url-fltr]
admin@firewall-a#
```

    Next, repeat the procedure for the next profile that you need to edit.
42. Type:
    up and press **Enter**.
43. Type:
    edit url[Tab][Tab].
44. Type:
    gue[Tab] and press **Enter**.
45. **Right-click** and select paste.
    (Optional) With the required **set** command stored in the command line history, you could use the **Up Arrow** to recall the **set** command *instead* of pasting it from the clipboard:

```
admin@firewall-a# up [Enter]
[edit profiles]
admin@firewall-a# edit url-filtering [Tab]
  corp-default-url-fltr        corp-default-url-fltr
  general-and-admin-url-fltr   general-and-admin-url-fltr
  guest-wifi-url-fltr          guest-wifi-url-fltr
```

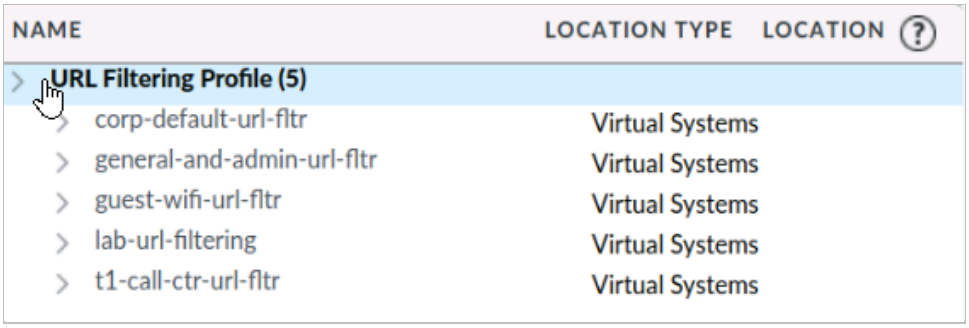
```
[. . .]
admin@firewall-a# edit url-filtering guest-wifi-url-fltr [Return]
[edit profiles url-filtering guest-wifi-url-fltr]
admin@firewall-a# set block [ copyright-infringement dynamic-dns parked proxy-
avoidance-and-anonymizers phishing ]

[edit profiles url-filtering guest-wifi-url-fltr]
admin@firewall-a#
```

46. (Optional) Use the CLI or web interface to verify the changes.
47. Repeat the *up-edit-paste-and-execute* procedure for the two profiles that remain: "lab-url-filtering" and "t1-call-ctr-url-fltr."

    Complete the repetition of these steps on your own. Refer to the prior steps in this manual and the Remmina terminal itself as needed for step-by-step guidance.

1.4.4 Review Changes and Commit the Configuration

48. After you add the required categories to all of the URL Filtering profiles, go to the web interface and click **Objects > Security Profiles > URL Filtering** to refresh the page.
49. Click the **magnifying glass** in the top right of the web interface to launch the Global Find feature, delete any pre-populated text in the text box, and then type: **`parked`** and press **Enter**.
50. Verify that the query returns five (5) URL Filtering profiles that include "parked" in their configurations. Then, click to expand the list and verify that the names in the list correspond to the names of the profiles that you just edited.



"Parked" is one of the shorter names of the URL categories that you added to each of the profiles and thus is easier to use to verify that you have modified all of the existing URL Filtering profiles as required.

Note: Depending on the situation, Global Find can be a powerful tool for producing useful information to help you troubleshoot configuration issues, especially issues related to a parameter that you know must exist in one or more objects and/or policies for an expected result to be produced.

51. Go to the CLI and press **Enter** to display the working context of configuration mode:

```
admin@firewall-a#[Enter]
```

```
[edit profiles url-filtering t1-call-ctr-url-fltr]
```

52. If the context is not **[edit profiles url-filtering t1-call-ctr-url-fltr]**, use your knowledge, and refer to prior steps as needed, to execute the commands required to match the CLI to this context.

    Next, you will display the candidate configuration in the other output formats to see what the configuration output looks like for modified candidate configurations.

53. Type:
    run set cli config-output-format json and press **Enter**.

    This command runs the operational-mode command to change the output format to json. The CLI does not display a confirmation message on a successful execution of the command.

54. Type:
    show and press **Enter**.

    The configuration-mode output for JSON-formatted output is relative to the working context. For the current working context, the output of the command will be more than 80 lines long.

55. Press the **spacebar** as needed to return to the command prompt.

    Focus your attention on the last 15 or so lines of output.

```
[. . .]
{
"@dirtyId":"10",
"dynamic-dns",
{
"@dirtyId":"10",
"parked",
{
"@dirtyId":"10",
"proxy-avoidance-and-anonymizers",
{
"@dirtyId":"10",
"phishing"]
}
,
"local-inline-cat":
"yes",
"cloud-inline-cat":
"no"}
[edit profiles url-filtering t1-call-ctr-url-fltr]
```

    What is different in this output compared to the default format?

    Note: Your “dirtyId” numbers likely will differ from the number (“18”) shown in the example.

56. Type:
    run set cli config-output-format xml and press **Enter**.

    This command changes the output format to XML.

57. Type:
    show and press **Enter**.

58. Press the **spacebar** as needed to return to the command prompt.

    Focus your attention on the last 10 or so lines of output.

```
[. . .]
        <member>web-hosting</member>
        <member admin="admin" dirtyId="10" time="2023/04/26 17:54:48">copyright-
           infringement</member>
        <member admin="admin" dirtyId="10" time="2023/04/26 17:54:48">dynamic-
           dns</member>
        <member admin="admin" dirtyId="10" time="2023/04/26 17:54:48">parked</member>
        <member admin="admin" dirtyId="10" time="2023/04/26 17:54:48">proxy-
           avoidance-and-anonymizers</member>
        <member admin="admin" dirtyId="10" time="2023/04/26
           17:54:48">phishing</member>
      </block>
      <local-inline-cat>yes</local-inline-cat>
      <cloud-inline-cat>no</cloud-inline-cat>
    </entry>
  </result>
</response>
[edit profiles url-filtering t1-call-ctr-url-fltr]
```

    Notice the **dirtyID** parameter, which also appears in the JSON-formatted output. This parameter is part of a tagging system for tracking changes to the candidate configuration. The default and set formats do not include this indicator of change.

59. Type:
    validate full and press **Enter**.

    This command initiates a validation of the configuration before you attempt to commit it:

```
admin@firewall-a# validate full

Validate job enqueued with jobid 3622
3622

[edit profiles url-filtering t1-call-ctr-url-fltr]
```

60. Type:
    run show jobs id <number> and press **Enter**.

    This command allows you to inspect the job results and confirm that the configuration is valid.

    Use the specific job ID number from your individual lab environment:

```
admin@firewall-a# run show jobs id 3622

Enqueued              Dequeued      ID          Type       Status Result Completed
------------------------------------------------------------------------------
2023/04/08 15:09:21   15:09:21    3622      Validate          FIN     OK 15:09:27
Warnings:
Details:Configuration is valid
```

```
[edit profiles url-filtering t1-call-ctr-url-fltr]
```

Confirm that the status is **FIN** and the **Result** is **OK**.

61. In the CLI, type:
    commit and press **Enter**:

```
admin@firewall-a# commit

Commit job 3645 is in progress. Use Ctrl+C to return to command prompt
...55% ...86% ...100%
Configuration committed successfully

[edit profiles url-filtering t1-call-ctr-url-fltr]
```

Note: If a configuration requires additional parameters to be set, the CLI will report a commit "validation error" similar to what you likely have seen before in the use of the web interface.

62. Verify that the current working level in the configuration hierarchy is **[edit profiles url-filtering t1-call-ctr-url-fltr]**.
63. Type:
    show and press **Enter**:

```
admin@firewall-a# show [Enter]
<response status="success" code="19">
  <result total-count="1" count="1">
    <entry name="t1-call-ctr-url-fltr">
[. . .]
        <member>web-hosting</member>
        <member>copyright-infringement</member>
        <member>dynamic-dns</member>
        <member>parked</member>
        <member>proxy-avoidance-and-anonymizers</member>
        <member>phishing</member>
      </block>
    </entry>
  </result>
</response>
[edit profiles url-filtering t1-call-ctr-url-fltr]
```

The output format still should be set to XML.

Notice that the configuration no longer contains any "dirtyID" parameters because all configuration elements displayed are the same as those in the current running configuration. That is, no part of the configuration displayed is pending a commit.

64. Type:
    run set cli config-output-format default and press **Enter**.

This command changes the output format to the default human-readable JSON format.

65. In the web interface, go to **Monitor > Logs > Configuration**.

    Look for the configuration activity that relates to the changes you have made in the CLI. Notice the correlation of the **Client**, **Command**, **Result**, **Configuration Path**, and **After Change** columns to the various actions you have taken.

    Some actions, such as uploading the configuration file, are not logged. However, logs are created for all operational activity that results in a change in policy or other system behavior:

ADMINISTRATOR	HOST	CLIENT	COMMAND	RESULT	CONFIGURATION PATH	FULL PATH	BEFORE CHANGE	AFTER CHANGE
admin	192.168.1.20	CLI	commit	Submitted				
admin	192.168.1.20	CLI	set	Succeeded	vsys vsys1 profiles url-filtering t1-call-ctr-url-fltr block	/config/devices/... filtering/entry[@... call-ctr-url-fltr']/block		block [copyright-infringement dynamic-dns parked proxy-avoidanc
admin	192.168.1.20	CLI	set	Succeeded	vsys vsys1 profiles url-filtering lab-url-filtering block	/config/devices/... filtering/entry[@... url-filtering']/block		block [copyright-infringement dynamic-dns parked proxy-avoidanc
admin	192.168.1.20	CLI	set	Succeeded	vsys vsys1 profiles url-filtering guest-wifi-url-fltr block	/config/devices/... filtering/entry[@... wifi-url-fltr']/block		block [copyright-infringement dynamic-dns parked proxy-avoidanc
admin	192.168.1.20	CLI	set	Succeeded	vsys vsys1 profiles url-filtering general-and-admin-url-fltr block	/config/devices/... filtering/entry[@... and-admin-url-fltr']/block		block [copyright-infringement dynamic-dns parked proxy-avoidanc
admin	192.168.1.20	CLI	set	Succeeded	vsys vsys1 profiles url-filtering corp-default-url-fltr block	/config/devices/... filtering/entry[@... default-url-fltr']/block		block [copyright-infringement dynamic-dns parked proxy-avoidanc

    Your specific log file entries may differ from the example shown.

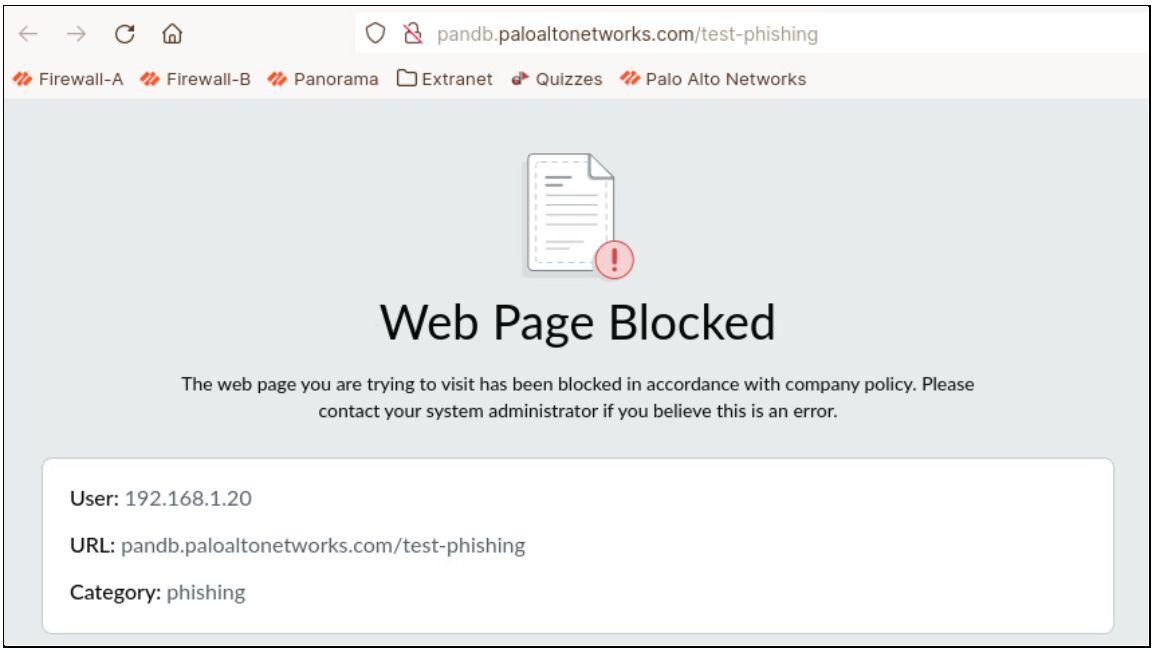
66. Leave the configuration browser open.

1.4.5 (Optional) Test URL Filtering Profile Changes

67. Open the testing browser and go to:
    `http://pandb.paloaltonetworks.com/test-phishing`

    Important: Make sure you use “http://” for the resource specification. If you use “https://”, the firewall will not block the connection because SSL decryption currently is not configured. The test site automatically will issue a redirect to HTTPS, which creates a race condition for the return of the initial URL lookup.

    If the redirection to HTTPS is successful, close the browser, wait several seconds, and then reissue the HTTP request. The connection attempt then should be blocked.



The URL categorization database (PAN-DB) catalogs this page as "phishing" so that you can test firewall policies related to this URL categorization.

Verify that a block page is displayed.

68. In the web interface of the firewall, go to **Monitor > Logs > URL Filtering** and look for the related log entry:

1.4.6 Reference Information

For an excellent example of various CLI commands that you can use to assess the current state of your firewall, generate a Tech Support File, download it, and open the text file (**techsupport_<hostname>_<YYYYMMDD>_<HHMM>.txt**) in the **/tmp/cli/logs** folder. Search for "**>[space]**" to advance from command to command and examine the output of each command.

1.4.7 Clean Up Your Lab Environment

69. Close all open testing browser tabs and windows. Clear any filters in logs.

    Leave open the configuration browser and the connection to the web interface of the firewall.

70. Close SSH windows for connections to the firewall.

Stop. This is the end of the lab.

2. Lab: Tracing Data-Plane Packet Flow

Lab Objectives

- Correlate packet-diagnostics log data with the high-level flow logic of the firewall
- Trace a *first packet* through the processing stages of the firewall
- Identify the difference between the process flow of the first packet and the process flows of subsequent packets that belong to the same session
- Demonstrate useful procedures for tracing sessions within packet-diagnostics logs

Lab Scenario

You have studied several flow charts that describe the flow logic of the firewall. You have participated in discussions about how the firewall processes traffic. You want to see some firewall-generated data that provides direct information about the flow logic of the firewall:

1. Find and open an existing packet-diagnostics log file.
2. Review individual log entries and interpret them based on what you have learned.

You will review a log file for a short HTTP session. After the TCP handshake is completed, the HTTP GET request is blocked. The URL requested is on the block list. No packets from other sessions are within this log file. The session consists of 10 packets from the initial SYN packet to the last ACK packet.

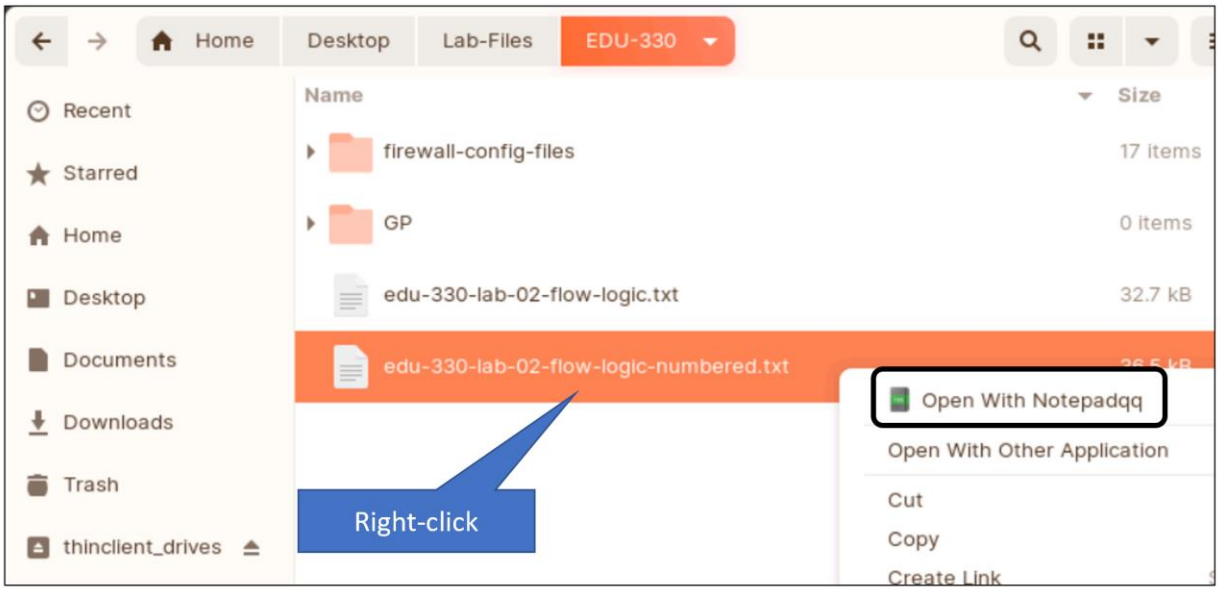
The following table describes the packet flow of the session traced. If you get lost in the log output, you can use the information provided in this table to help re-orient yourself.

No.	Packet type	Description
1.	SYN (c2s)	This is the “first packet” of the session. You can trace this packet through the following stages: Ingress > slowpath (session setup) > fastpath (security processing) > forwarding (egress).
2.	SYN-ACK (s2c)	This is the “second packet” of the session. You can trace this packet through the following stages: Ingress > fastpath (security processing) > forwarding (egress).
3.	ACK (c2s)	This packet completes the TCP handshake. You can trace this packet through the ingress, fastpath (security processing), and forwarding (egress) stages.
4.	ACK-PSH (c2s)	This packet contains an HTTP GET request. This log file includes content-based threat detection (CTD) tracing, in addition to *basic* packet-flow logging. After NAT is applied, you can find the detection of the blocked URL. The firewall *drops* this packet and triggers two

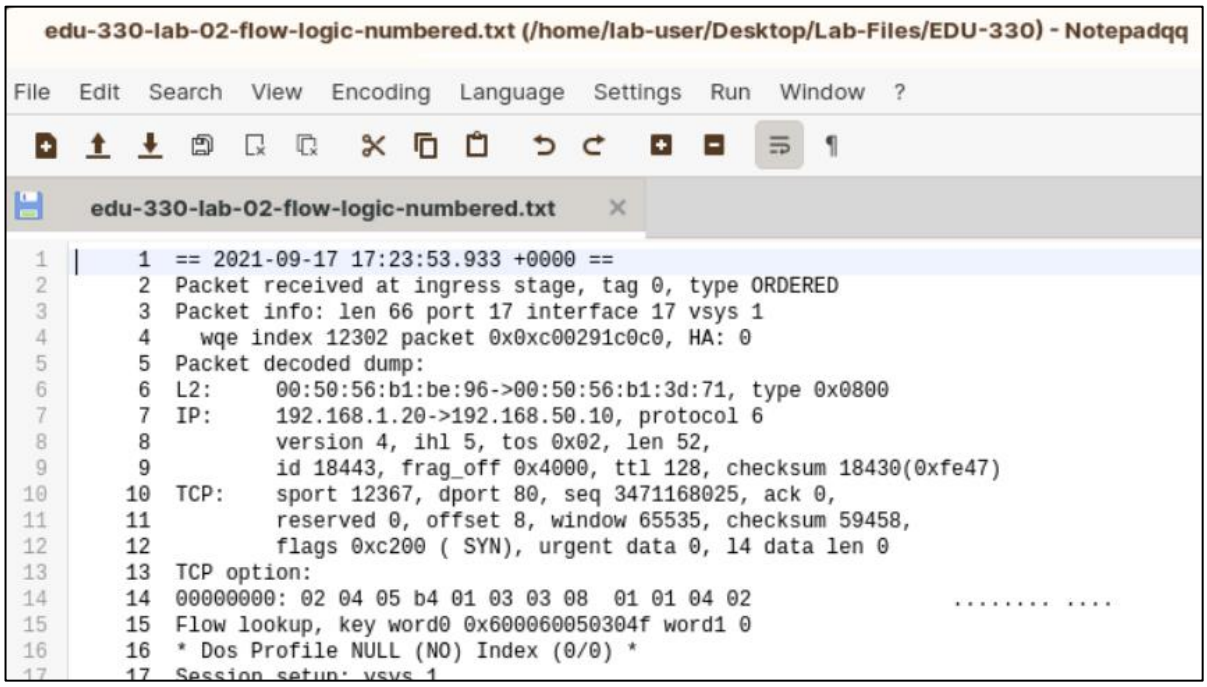
No.	Packet type	Description
		primary actions: 1) Close the session and send a block page to the initiator, and 2) Close the session with the responder.
5.	ACK-RST (f2s)	This TCP reset (RST) packet is sent to the server from the firewall itself. This packet is generated by the firewall and, from the point of view of the log file, is injected directly into the forwarding (egress) stage. This packet requests a termination of the server-side connection.
6.	ACK-PSH (f2c)	This packet is sent from the firewall to the client (f2c). This packet is a spoofed reply from the server. It contains an HTTP 503 Service Unavailable header, along with an HTML block page. (Note, however, that this Layer 7 information cannot be seen in the log file.) This packet is generated by the firewall and, from the point of view of the log file, is injected directly into the forwarding (egress) stage.
7.	ACK-FIN (f2c)	This TCP finish (FIN) packet is sent from the firewall to the client. This packet requests a termination of the client-side connection. This packet is generated by the firewall and, from the point of view of the log file, is injected directly into the forwarding (egress) stage.
8.	ACK (c2s)	The client acknowledges the receipt of the HTTP 503 error with this packet. You can trace this packet from ingress to fastpath. The firewall then *drops* this packet (does not forward it to the server) and brokers the close of the session with the server.
9.	ACK-FIN (c2s)	The client acknowledges the ACK-FIN packet from the firewall. You can trace this packet from ingress to fastpath, where the packet is *dropped*.
10.	ACK (f2c)	The firewall acknowledges the ACK-FIN from the client. This packet completes the session from the point of view of the client. The server does not reply after receiving the prior RST packet. This is the last packet of the session.

2.1 Open the Packet-Diagnostics File

1. Open the **Lab-Files/EDU-330 folder** on the student desktop.
2. Find the **edu-330-lab-02-flow-logic-numbered.txt** file, right-click it, and choose **Open with "Notepadqq"**:



3. Verify that the file opens properly in **Notepadqq**:



Tip: Resize Notepadqq window to maximize the number of vertical lines that you can see at one time. You can click the bottom frame of the window and drag it to the bottom of the desktop.

Note the line numbering on the left side of Notepadqq window. Subsequent steps in this activity will reference these line numbers to help guide your interpretation of the log data.

2.2 Trace the First Packet Through the Firewall

Note: You will see much more data in the log file than this lab activity will explain. You likely will be able to decode much of the meaning of various log data based on your general knowledge of networking, combined with your practical experience with Palo Alto Networks firewalls. Some items in the packet-diagnostics log output may be considered proprietary to engineering and will not be addressed at all.

4. On line **2**, find the text "**`Packet received at ingress stage`**":

```
1   == 2021-09-17 17:23:53.933 +0000 ==
2   Packet received at ingress stage, tag 0, type ORDERED
3   Packet info: len 66 port 17 interface 17 vsys 1
4     wqe index 12302 packet 0x0xc00291c0c0, HA: 0
5   Packet decoded dump:
[...]
```

Note the location of the hexadecimal packet identifier, **`0x0xc00291c0c0`** in this example.

5. In lines **6 through 15**, examine the results of the initial packet parsing of the Layer 2 (Ethernet), Layer 3 (IP), and Layer 4 (TCP) headers of the packet:

```
[...]
5   Packet decoded dump:
6   L2:     00:50:56:b1:be:96->00:50:56:b1:3d:71, type 0x0800
7   IP:     192.168.1.20->192.168.50.10, protocol 6
8           version 4, ihl 5, tos 0x02, len 52,
9           id 18443, frag_off 0x4000, ttl 128, checksum 18430(0xfe47)
10  TCP:    sport 12367, dport 80, seq 3471168025, ack 0,
11          reserved 0, offset 8, window 65535, checksum 59458,
12          flags 0xc200 ( SYN), urgent data 0, l4 data len 0
13  TCP option:
14  00000000: 02 04 05 b4 01 03 03 08  01 01 04 02                ........ ....
[...]
```

This information is similar to what you might see in the text-based display of a pcap file.

6. On line **15**, find the text "**`Flow lookup`**."
7. On line **18**, find the text "**`No active flow found`**":

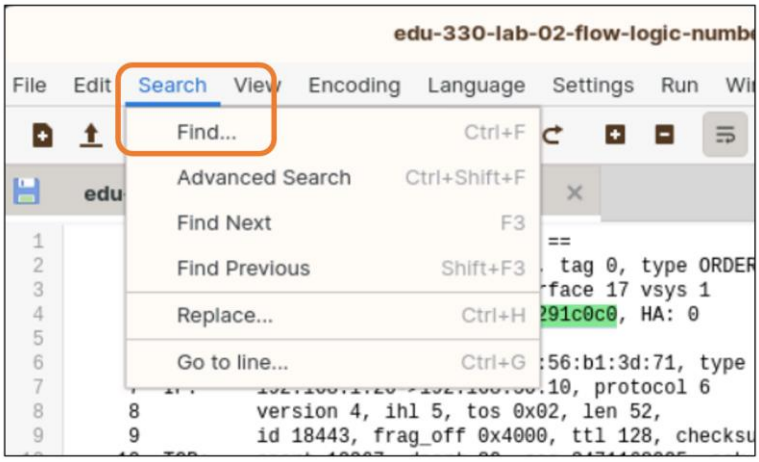
```
[...]
14  00000000: 02 04 05 b4 01 03 03 08  01 01 04 02                ........ ....
15  Flow lookup, key word0 0x600060050304f word1 0
16  * Dos Profile NULL (NO) Index (0/0) *
17  Session setup: vsys 1
18  No active flow found, enqueue to create session
[...]
```

(Optional) Cross-reference these events to the "**Match to an existing session?**" decision point in the **Ingress** flow diagram in your Student Guide. Because this TCP SYN packet is the *first packet* of a new session, no active flow is found. The next stage is session setup, or *slowpath*.

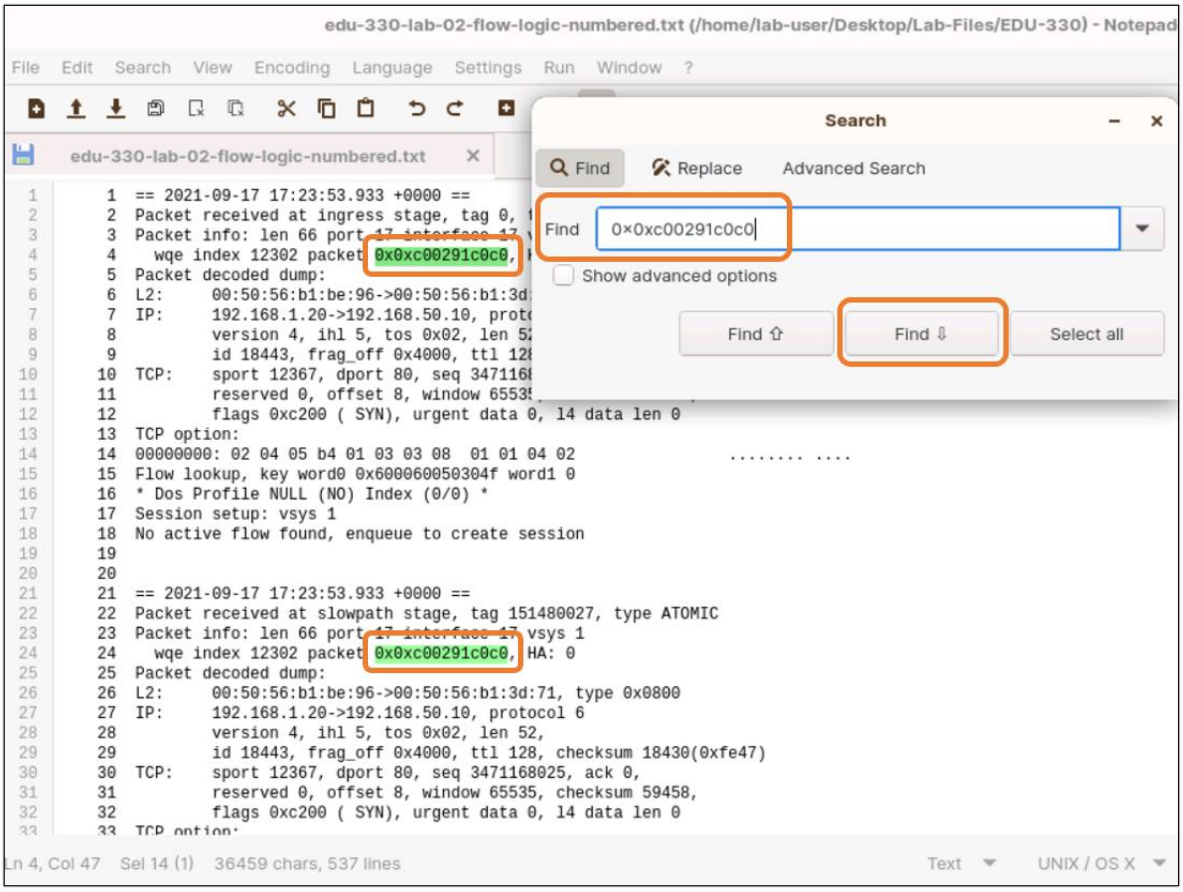
8. On line **4**, locate "**`packet 0x0xc00291c0c0`**" and double-click **`0x0xc00291c0c0`**.
   In this procedure we will utilize the search feature of Notepadqq

   Note: You can use IP ID numbers or TCP sequence numbers to cross-reference specific packets in flow-basic output with the contents of the pcap file, and vice versa.

9. Go to the **Search** menu at the top of the file and select **Find**.



10. The following search fields will appear:



11. Take the selected packet and paste it into the "**Search for**" field. Click **Find** ↓ to continue as outlined in the steps that follow.
12. Notice the heading "**`Packet received at slowpath stage`**" on line **22**, two lines above the second instance of the packet ID.
13. Use the scroll bar for **Notepadqq** (or your mouse wheel or touchpad function) to scroll down past line **49** until you see the *third* instance of this same packet ID that first appears on line **4**.
14. Notice the heading "**`Packet received at fastpath stage`**" on line **49**, two lines above the third instance of the packet ID:

```
[. . .]
48  == 2021-09-17 17:23:53.934 +0000 ==
49  Packet received at fastpath stage, tag 44187, type ATOMIC
50  Packet info: len 66 port 17 interface 17 vsys 1
51    wqe index 12302 packet 0x0xc00291c0c0, HA: 0
52  Packet decoded dump:
53  L2:     00:50:56:b1:be:96->00:50:56:b1:3d:71, type 0x0800
[. . .]
```

(Optional) Scroll down through the rest of the file to verify that that the packet ID is unique to the three records (ingress, slowpath, and fastpath) that you have just reviewed.

Tip: In **Notepadqq** *with your search word selected*, press **Ctrl+F** to automatically populate the **Find** dialog, and then click **Next**. After you search, close the **Find** dialog.

15. On line **49**, double-click the **tag** number of the fastpath record, hit CRTL+F, and then **scroll up** to display all of the slowpath record and as much of the fastpath record as possible:

```
21  == 2021-09-17 17:23:53.933 +0000 ==
22  Packet received at slowpath stage, tag 151480027, type ATOMIC
23  Packet info: len 66 port 17 interface 17 vsys 1
24    wqe index 12302 packet 0x0xc00291c0c0, HA: 0
25  Packet decoded dump:
[. . .]
35  Session setup: vsys 1
36  PBF lookup (vsys 1) with application web-browsing
37  Session setup: ingress interface ethernet1/2 egress interface ethernet1/3 (zone
3)
38  NAT policy lookup, matched rule index 1
39  Policy lookup, matched rule index 0,
40  Allocated new session 44187.
41  Rule: index=1 name=source-nat-to-dmz, cfg_pool_idx=3 cfg_fallback_pool_idx=0
42  NAT Rule: name=source-nat-to-dmz, cfg_pool_idx=3; Session: index=44187,
nat_pool_idx=3
43  Packet matched vsys 1 NAT rule 'source-nat-to-dmz' (index 2),
44  source translation 192.168.1.20/12367 => 172.16.16.16/14305
45  Created session, enqueue to install

== 2021-09-17 17:23:53.934 +0000 ==
49  Packet received at fastpath stage, tag 44187, type ATOMIC
50  Packet info: len 66 port 17 interface 17 vsys 1
51    wqe index 12302 packet 0x0xc00291c0c0, HA: 0
52  Packet decoded dump:
53  L2:     00:50:56:b1:be:96->00:50:56:b1:3d:71, type 0x0800
54  IP:     192.168.1.20->192.168.50.10, protocol 6
55          version 4, ihl 5, tos 0x02, len 52,
56          id 18443, frag_off 0x4000, ttl 128, checksum 18430(0xfe47)
57  TCP:    sport 12367, dport 80, seq 3471168025, ack 0,
58          reserved 0, offset 8, window 65535, checksum 59458,
59          flags 0xc200 ( SYN), urgent data 0, 14 data len 0
60  TCP option:
61  00000000: 02 04 05 b4 01 03 03 08  01 01 04 02                ........ ....
62  Flow fastpath, session 44187
```

After a session ID has been assigned (reference line **40**), you can use the session ID to trace the succession of packets that belong to an individual session. In contrast to the example log file, which is limited to a single session, most real-world packet traces will include interleaving packet entries from multiple sessions.

(Optional) With the session ID number highlighted, scroll down through the file to see the persistent nature of the session ID relative to subsequent packet-diagnostics log entries. Notice that none of *ingress-stage records* includes a reference to the session ID. Why?

16. On line **35**, find the text "**Session setup: vsys 1**."

17. Cross-reference the events recorded on lines **36** through **44** with the **Session Setup (“slowpath”)** diagram in your Student Guide:

```
35  Session setup: vsys 1
36  PBF lookup (vsys 1) with application web-browsing
37  Session setup: ingress interface ethernet1/2 egress interface ethernet1/3 (zone
3)
38  NAT policy lookup, matched rule index 1
39  Policy lookup, matched rule index 0,
40  Allocated new session 44187.
41  Rule: index=1 name=source-nat-to-dmz, cfg_pool_idx=3 cfg_fallback_pool_idx=0
42  NAT Rule: name=source-nat-to-dmz, cfg_pool_idx=3; Session: index=44187,
nat_pool_idx=3
43  Packet matched vsys 1 NAT rule 'source-nat-to-dmz' (index 2),
44  source translation 192.168.1.20/12367 => 172.16.16.16/14305
45  Created session, enqueue to install
```

Consider the following notes:

- The “**Zone Protection Profile | TCP state check**” step illustrated in the Student Guide is not logged unless there is a problem.
- Lines **36** and **37** consist of the “**Forwarding lookup: Get egress zone**” process. Specifically, on line **37** find the text “**`egress interface ethernet1/3 (zone 3)`**.”
- The “**D-NAT apply?**” decision point in the flow chart occurs on line **38**.
- A “**DoS Protection policy**” is not in effect. The log does not reference this step.
- “**Security Policy evaluation**” is referenced on line **39**.
- NAT information continues to be identified and is logged on lines **41** through **44**.
- The “**Set Up Session**” action illustrated in the Student Guide occurs on line **45**.

18. On line **67**, in the fastpath-stage record, locate the text “**`NAT session, run address/port translation`**.”

    This event correlates to the “**Apply NAT**” action illustrated in the **Security Processing (“fastpath”)** diagram in your Student Guide. Contrast this log event with the “**`NAT policy lookup`**” event that is logged earlier in the Session Setup (“slowpath”) stage.

19. On line **69**, in the fastpath-stage record, locate the text “**`Forwarding lookup, ingress interface <number>`**.”

    If a QoS policy or other egress-*queue* condition does not apply, a separate log section for the forwarding (egress) stage typically is not created. For SYN, ACK, FIN, RST, and other packets that do not have payload data, even if additional debug logging features are applied (as *is* the case in this example), logging for egress is appended directly to the fastpath stage. The “**`Forwarding lookup`**” action is the same as the first action of the **Egress** flow chart in the Student Guide.

20. On line **75**, find the text “**`Transmit packet on port <number>`**.”

    This entry marks the completion of the flow logic of the firewall for the first packet.

2.3 Trace the Second Packet

21. Locate line **81**, notice that this line is just below a new ingress-stage log record, and then double-click the packet ID.
22. Use **CTRL+F** to update the search field in Notepadqq to highlight it.
23. Use the window controls to display line **81** and line **100** at the same time:

```
78  == 2021-09-17 17:23:53.935 +0000 ==
79  Packet received at ingress stage, tag 0, type ORDERED
80  Packet info: len 66 port 18 interface 18 vsys 1
81    wqe index 11217 packet 0x0xc003027cc0, HA: 0
[. . .]
92  Flow lookup, key word0 0x6000337e10050 word1 0
93  Flow 88375 found, state 2, HA 0
94  Active flow, enqueue to fastpath process, type 0

97  == 2021-09-17 17:23:53.935 +0000 ==
98  Packet received at fastpath stage, tag 44187, type ATOMIC
99  Packet info: len 66 port 18 interface 18 vsys 1
100   wqe index 11217 packet 0x0xc003027cc0, HA: 0
101 Packet decoded dump:
102 L2:     00:50:56:b1:fe:8e->00:50:56:b1:a5:1e, type 0x0800
103 IP:     192.168.50.10->172.16.16.16, protocol 6
[. . .]
```

Notice on line **94** that the flow key for this second packet of the session is found without the firewall ever having seen a packet with the 6-tuple properties required to produce this key. The firewall's correct anticipation of the flow key of this packet results from the forwarding and NAT lookups that the firewall performed in the slowpath process for the *first packet*.

24. Examine the fastpath-stage log data for this second packet and compare it to the fastpath-stage processing of the first packet.

    This packet does not have any payload, and so the rest of its profile is similar to the fastpath processing of the SYN packet.

25. For the third packet of the session, apply the same procedures and conduct the same analysis of the ingress and fastpath records as you did for the first and second packets.

    The third packet is logged on lines **125** through **165**.

2.4 Trace the Content Inspection of a Packet

At line **165**, the end-to-end TCP handshake between client **192.168.1.20** and server **192.168.50.10** is complete. The next packet is an HTTP GET request for the URL **www.test.lab/orange**. The firewall is configured to block this URL.

26. On line **171**, double-click the packet ID **0x0xc001a9a840**.
27. Use **CTRL+F** to update the search field in Notepadqq to highlight it.

The ingress stage produces a flow key that matches to an existing session. Therefore, the packet is admitted to fastpath processing.

For this log file, the debug *feature* that traces Layer 7 content-based threat detection (CTD) has been enabled. This extra logging enables you to see that NAT translation is applied on line **201** *before* content analysis begins as logged on lines **202** through **204**. This is the first packet of the session that has a *payload*:

```
168 == 2021-09-17 17:23:53.935 +0000 ==
169 Packet received at ingress stage, tag 0, type ORDERED
170 Packet info: len 315 port 17 interface 17 vsys 1
171   wqe index 11217 packet 0x0xc001a9a840, HA: 0
172 Packet decoded dump:
[. . .]
174 IP:     192.168.1.20->192.168.50.10, protocol 6
[. . .]
181 Flow lookup, key word0 0x600060050304f word1 0
182 Flow 88374 found, state 2, HA 0
183 Active flow, enqueue to fastpath process, type 0

186 == 2021-09-17 17:23:53.935 +0000 ==
187 Packet received at fastpath stage, tag 44187, type ATOMIC
188 Packet info: len 315 port 17 interface 17 vsys 1
189   wqe index 11217 packet 0x0xc001a9a840, HA: 0
190 Packet decoded dump:
[. . .]
199 Flow fastpath, session 44187
200 IP checksum valid
201 NAT session, run address/port translation
202 session 44187 packet sequeunce old 0 new 1
203 192.168.1.20[12367]-->192.168.50.10[80]
204 2021-09-17 17:23:53.936 +0000  pan_ctd_session_init(pan_ctd.c:3087): ********
session_init app= 109, profileid = 1, decoder_appid = 109
[. . .]
```

28. Scroll down from line **205** to line **274** or **279**:

```
[. . .]
274 2021-09-17 17:23:53.937 +0000 debug: __pan_sml_vm_run_impl(pan_sml_vm.c:2818):
seq 0 doff 174
275 2021-09-17 17:23:53.937 +0000 debug: pan_sml_vm_end_field(pan_sml_vm.c:5566):
[field_end] offset 174 seq 0 is_cts 1
276 2021-09-17 17:23:53.937 +0000 debug: pan_sml_vm_end_field(pan_sml_vm.c:5595):
[end field] fid  2483 from 69 to ae flag:a000080a
277 2021-09-17 17:23:53.937 +0000 debug: pan_sml_vm_add_field(pan_sml_vm.c:5623):
add_field fid 2483 offset 174 seq 0 is_cts 1
278 2021-09-17 17:23:53.937 +0000 debug: __pan_sml_vm_run_impl(pan_sml_vm.c:2818):
seq 0 doff 191
279 2021-09-17 17:23:53.937 +0000 debug: __pan_sml_vm_run_impl(pan_sml_vm.c:2818):
seq 0 doff 196
280 2021-09-17 17:23:53.937 +0000 debug: __pan_sml_vm_run(pan_sml_vm.c:4717):
281 [skip] PC 53761 seq 0 skipoff 197 dlen 261
[. . .]
```

Generally you see traces of content scans as they progress deeper and deeper into the packet data, based on the threat types and pattern profiles that are available to the system.

The term **doff** means "data offset." As you progress down the log entries, notice that the **doff** number increases as various *field* scans of various sizes are made in a similarly progressive way.

29. At line **357** notice that the firewall now is focused explicitly on URL information.

    Scan for the word "**block**" in the CTD events:

```
[. . .]
357 2021-09-17 17:23:53.938 +0000 debug:pan_urlcache_lookup(pan/src/pan_urlcache.c:930):
    In pan_urlcache_lookup, e 0, c 252
358 2021-09-17 17:23:53.938 +0000 debug:
pan_urlcache_lookup(pan/src/pan_urlcache.c:966): TRIE LOOKUP: url www.test.lab/orange/
359 2021-09-17 17:23:53.938 +0000 debug:
pan_urlcache_lookup(pan/src/pan_urlcache.c:1044):
    res 1, PAN_URL_TRIE_NOT_IN_DB 4, cat.num 1,cat.cat[0] 251, PAN_URL_CTGR_NOT_RESOLVED
    252, pan_url_trie_is_rfs_expired 0, ucache->cloud_up 1
360 2021-09-17 17:23:53.938 +0000 debug:
    pan_urlcache_lookup_ext(pan/src/pan_urlcache.c:1131): Add to the vector
361 2021-09-17 17:23:53.938 +0000 debug: pan_ctd_handle_url(pan_ctd.c:11403): appid
109(from 109), num. of categories 1
362 2021-09-17 17:23:53.938 +0000 debug: pan_ctd_app_policy_lookup_i(pan_ctd.c:6822):
    Session id(44187): rule changed to internal-inside-dmz-http from
    internal-inside-dmz-http action(0); logging(2); profile id(1) category
    private-ip-addresses(10077)
363 192.168.1.20[12367]-->192.168.50.10[80]
364 2021-09-17 17:23:53.938 +0000  pan_ctd_url_log_action(pan_ctd.c:6400):
365        url 'www.test.lab:0/orange' category block-list, action 0 sess 44187 idx 1
366 192.168.1.20[12367]-->192.168.50.10[80]
367 2021-09-17 17:23:53.938 +0000  pan_ctd_handle_url_denied_i(pan_ctd.c:6493):
    url action block, credential not matched
[. . .]
```

30. On lines **368 through 370**, find the text "**Flow action**":

```
[. . ]
368 Flow action close for session 44187, option 2
369 Flow action send data for session 44187 direction S2C
370 Flow action close for session 44187, option 1
[. . .]
```

Notice that neither a "Forwarding lookup" entry nor a "Transmit packet" entry is made at the end of the fastpath record for this packet. In fact, no forwarding-stage log entries are made at all, because the firewall drops the packet and triggers the following two actions:

- Close the session with the client-initiator after sending a block page
- Close the session with the server-responder

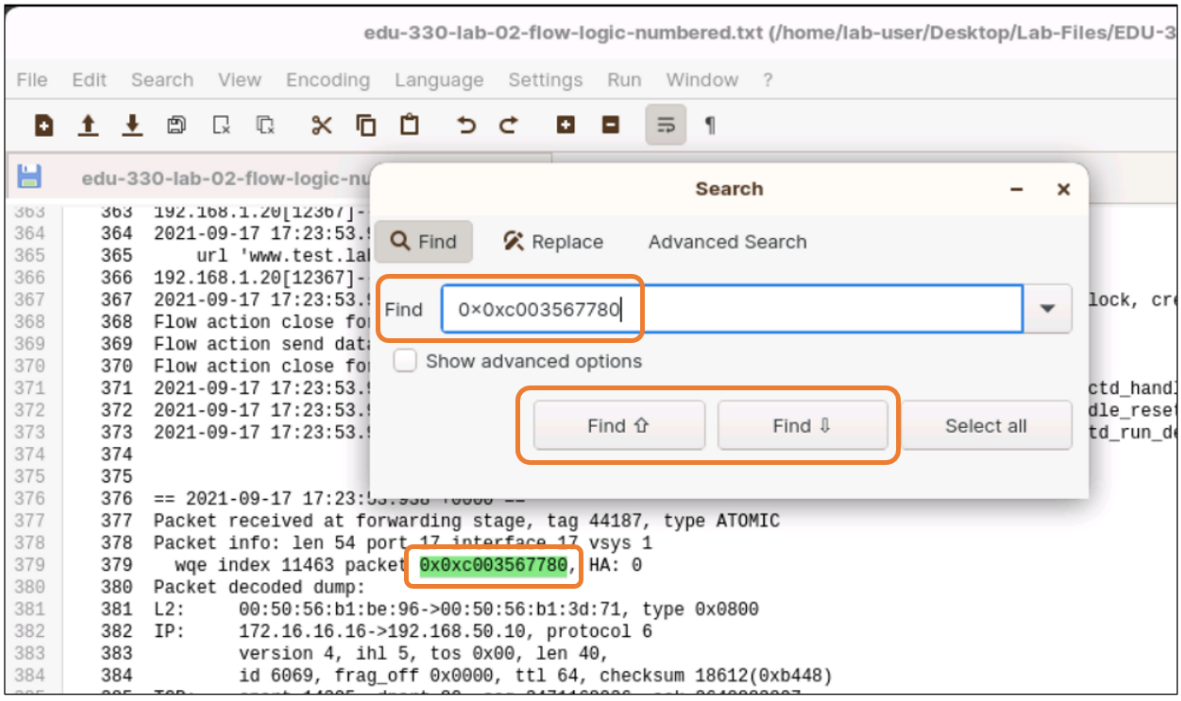
2.5 Identify Firewall-Generated Packets

31. Scroll down so that lines **375** to **395** are displayed.

32. On line **377**, double-click the tag number **44187** to verify that this packet is linked to the same session ID (**44187**) that we have been following.
33. Next, double-click the packet ID on line **379**:

```
376 == 2021-09-17 17:23:53.938 +0000 ==
377 Packet received at forwarding stage, tag 44187, type ATOMIC
378 Packet info: len 54 port 17 interface 17 vsys 1
379   wqe index 11463 packet 0x0xc003567780, HA: 0
380 Packet decoded dump:
381 L2:     00:50:56:b1:be:96->00:50:56:b1:3d:71, type 0x0800
382 IP:     172.16.16.16->192.168.50.10, protocol 6
383         version 4, ihl 5, tos 0x00, len 40,
384         id 6069, frag_off 0x0000, ttl 64, checksum 18612(0xb448)
385 TCP:    sport 14305, dport 80, seq 3471168026, ack 2648883807,
386         reserved 0, offset 5, window 1024, checksum 50057,
387         flags 0x1400 ( ACK RST), urgent data 0, l4 data len 0
388 TCP option:
389 Forwarding lookup, ingress interface 17
[. . .]
```

34. Press **Ctrl+F** and use the **Find** dialog to search next and previous for another instance of the packet ID **0x0xc003567780**:



This packet ID does not appear anywhere else in the log file.

35. On line **382**, identify the source and destination IP addresses.

172.16.16.16 is the source NAT address for the session. **192.68.50.10** is the server's address.

36. On line **387**, identify the type of TCP packet.

What is the function of this RST packet? Why is the firewall sending it?

37. Scroll down so that lines **398** to **417** are displayed.
38. Perform the same analysis that you just performed on the prior packet:
    - Check the tag number; verify that the packet belongs to the same session.
    - Check the packet for its uniqueness.
    - Identify the source and destination addresses.
    - Identify the type of packet.

```
398 == 2021-09-17 17:23:53.938 +0000 ==
399 Packet received at forwarding stage, tag 44187, type ATOMIC
400 Packet info: len 1289 port 17 interface 18 vsys 1
401   wqe index 10959 packet 0x0xc001f30c80, HA: 0
402 Packet decoded dump:
403 L2:     00:50:56:b1:3d:71->00:50:56:b1:be:96, type 0x0800
404 IP:     192.168.50.10->192.168.1.20, protocol 6
405         version 4, ihl 5, tos 0x00, len 1275,
406         id 16535, frag_off 0x0000, ttl 64, checksum 63360(0x80f7)
407 TCP:    sport 80, dport 12367, seq 2648883807, ack 3471168026,
408         reserved 0, offset 5, window 1024, checksum 46624,
409         flags 0x1800 ( ACK PSH), urgent data 0, l4 data len 1235
410 TCP option:
411 Forwarding lookup, ingress interface 18
412 L3 mode, virtual-router 1
413 Route lookup in virtual-router 1, IP 192.168.1.20
414 Route found, interface ethernet1/2, zone 6
415 Resolve ARP for IP 192.168.1.20 on interface ethernet1/2
416 ARP entry found on interface 17
417 Transmit packet on port 17
```

The payload of this packet contains an HTTP 503 Service Unavailable response, along with an HTML-formatted block page that the end user's browser will render. This type of Layer 7 information is not provided in this type of log. Only if you run a simultaneous packet capture would you be able to know the content of the packet payload precisely.

However, the TCP segment length information (**len 1235**) on line **409** does tell you that this packet contains a payload. Compare the length of this packet with the length value on line **387** for the prior RST packet, which has no payload.

39. Scroll down so that lines **420** to **439** are displayed and examine the elements that interest you.

    This ACK-FIN packet also is generated by the firewall. The firewall sends the ACK-FIN packet to the client to close the session after the firewall has sent the block page.

2.6 Identify Other Dropped Packets and the Session End

40. On line **421**, double-click the tag number **44187**.
41. Use **CTRL+F** to update the search field in Notepadqq to highlight it.
42. Scroll down until you find the next highlighted tag number on line **461** and the corresponding session number log entry on line **473**.

43. On line **463**, double-click the packet ID number, and scan up to find the ingress-stage log data that corresponds to this packet:

```
460 == 2021-09-17 17:23:53.938 +0000 ==
461 Packet received at fastpath stage, tag 44187, type ATOMIC
462 Packet info: len 60 port 17 interface 17 vsys 1
463   wqe index 11217 packet 0x0xc0035c4b40, HA: 0
464 Packet decoded dump:
465 L2:     00:50:56:b1:be:96->00:50:56:b1:3d:71, type 0x0800
466 IP:     192.168.1.20->192.168.50.10, protocol 6
467         version 4, ihl 5, tos 0x00, len 40,
468         id 18449, frag_off 0x4000, ttl 128, checksum 20478(0xfe4f)
469 TCP:    sport 12367, dport 80, seq 3471168287, ack 2648885043,
470         reserved 0, offset 5, window 1019, checksum 49071,
471         flags 0x1000 ( ACK), urgent data 0, l4 data len 0
472 TCP option:
473 Flow fastpath, session 44187
474 IP checksum valid
475 FIN proxy proc ACK 2648885043, first seq 2648883807
476 Packet dropped: FIN proxy enabled. tcb state 5
```

Where did this packet come from? What happened to it? Why?

Answers: This packet comes from the client in response to the ACK-PSH packet that the firewall sent and that contained the block page. The firewall drops the ACK packet from the client because the firewall now is blocking the connection to the addressed destination server (192.168.50.10), to which the firewall already has sent a reset (RST) packet.

44. Use the techniques that you have learned so far to find and analyze the last two packets in the log.

Formulate answers to the following questions about the second-to-last packet:

- Which major processing stages does the packet pass through?
- What happens to the packet?
- Which node sent the packet, and why?
- Does the packet have a payload?

Formulate answers to the following questions about the last packet:

- Which major processing stages does the packet pass through?
- Which node sent this packet, and why?

2.7 Clean Up Your Lab Environment

45. Close Notepadqq program after you are finished.
46. Close the File Manager and Terminal windows.

2.8 Reference Information

Key terms that you may find in packet-diagnostics logs are as follows:

- **AHO**: Abbreviated term for Aho-Corasick trie, which is a pattern-matching technology for the detection of viruses and other threats.
- **DFA**: Deterministic Finite Automata is a jargon term for a pattern-matching and RegEx engine for App-ID.
- **dlen**: Length of a data segment to parse and/or scan.
- **doff**: A data offset value, often in bytes, that identifies how far from the starting point to begin to define the next segment or field in a stream of bits.
- **ORDERED** and **ATOMIC**: Names for two types of processor work. *Ordered* work can be processed by the next available core. *Atomic* work consists of multiple tasks that must be handled by the same core.
- **packet [hexadecimal number]**: This number provides a unique identifier for each packet.
- **tag**: Number used to track processing tasks (or *work*). The tag for the ingress stage always is 0. The tag for slowpath (session-setup) is a globally sequential number for the work queue across all available cores. After a session ID is assigned, the tag assigned by the ingress stage will be equal to the session ID, which will be indexed back to the core that set up the session.
- **trie**: A jargon term in computer science that references a tree-based method for searching a data set.
- **wqe index**: A work queue number for tracking work. These numbers link various elements of the session.

Stop. This is the end of the lab.

3. Lab: Packet Capture

Lab Objectives

- Configure packet filters and examine the effects of using various options
- Monitor marked sessions using the CLI
- Take a packet capture for a session that includes dropped packets
- Review the difference in packets captured at each stage

Lab Scenario

Packet captures can help you to troubleshoot many network and firewall-configuration issues. If you know what is typical to see in pcaps taken from the different capture stages of the firewall, you more accurately can interpret the data that pcaps provide.

You will use the lab environment to take one or more simple packet captures that you can use to discover the differences among packets captured at all four capture stages. To accomplish this goal, you will do the following:

- Use the (DMZ) server at 192.168.50.10 as a remote web server.
- Load a configuration that includes a dynamic IP and port (DIPP) NAT policy:
  - The external IP address used for this policy is 172.16.16.16.
  - The NAT policy will enable you to see the effects of NAT on filtering and packet captures and to identify the state of packets as pre- or post-Layer-2-to-4 processing.
- Configure packet-capture filters and capture stages, perform packet captures, and analyze the results.

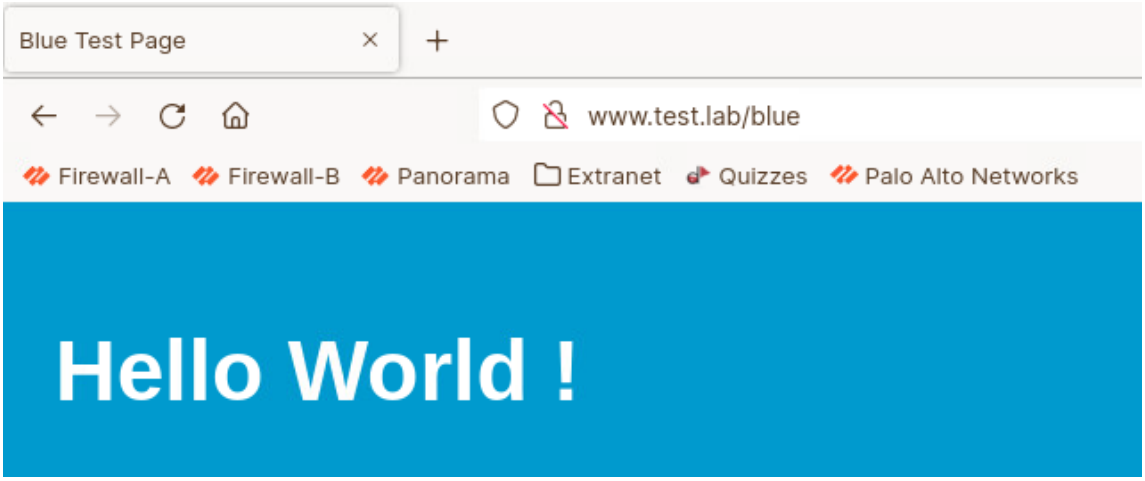
3.1 Load a Configuration and Test Baseline Functionality

1. Use the configuration browser to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-03.xml**

   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.
2. After the load task is complete, use the web interface to **Commit** the configuration.
3. Go to **Policies > NAT**.
4. Note (write down) the current **Hit Count** number for the **source-nat-to-dmz** policy.
5. Open a Terminal and type:
   `ping www.test.lab` and press **Enter**.

   The ping should be successful. **Press Ctrl+C to stop the Ping after several seconds**.

6. Close the **Terminal** window.
7. Return to **Policies** > **NAT**, refresh the page, and verify that the hit count number has increased.
8. Open the testing browser and go to **`http://www.test.lab/blue`** and verify that a blue page with text that reads "Hello World" is displayed:



Alternate colors are available, including **`black`**, **`red`**, **`orange`**, **`yellow`**, **`green`**, **`sky`**, **`purple`**, **`gray`**, and **`white`**. You may substitute any of these colors for blue throughout the activity.

Note: If you use the Google Chrome, Firefox or Chromium browsers, you *must* type "**`http://`**" each time before the hostname; otherwise, they will turn the address-bar entry into a search, and your attempt to load the page will fail.

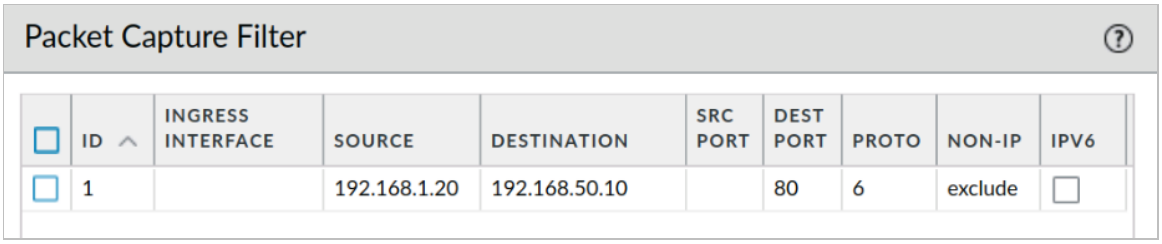
9. Close the testing browser.

3.2 Configure a Packet Filter

10. In the firewall web interface, go to **Monitor** > **Packet Capture**.
11. Click **Manage Filters** and **Add** a packet-capture filter using the following specifications:

Parameter	Value
Id	**`1`**
Ingress Interface	none [blank]
Source	**`192.168.1.20`**
Destination	**`192.168.50.10`**
Src Port	[blank]
Dest Port	**`80`**
Proto	**`6`**

Parameter	Value
Non-IP	**Exclude**
IPv6	not selected



12. Click **OK**.

3.3 Test Session Marking

13. Click to turn the **Filtering** function **ON** (if it is not already on):

14. Go to the **CLI of the firewall**. You can use an existing SSH session (via **Remmina**) if you have one open; otherwise launch **Remmina** and connect to **Firewall-A**.
15. Identify the current command mode, indicated by the prompt symbol **#** or **>**. If the CLI is in configuration mode (**#**), type **exit** and press **Enter**. The prompt sign should be "**>**".
16. Type:
    find command keyword mark and press **Enter**:

```
admin@firewall-a> find command keyword mark
debug dataplane show ctd ctd-queue-water-mark
debug dataplane reset ctd ctdf-water-mark
debug dataplane reset ctd ctd-queue-water-mark
debug dataplane packet-diag set filter-marked-session id <1-4294967295>
debug dataplane packet-diag clear filter index <1-4>|<all> clear-marked-session <yes|no>
debug dataplane packet-diag clear filter-marked-session id <1-4294967295>
debug dataplane packet-diag clear filter-marked-session all
debug dataplane packet-diag show filter-marked-session
debug dataplane pow status high-watermark worker <value> cos <value> reset <yes|no>
```

17. Use your mouse to select "**debug dataplane packet-diag show filter-marked-session**" and then right-click and copy and paste to add this command to the active command prompt.
18. Press **Enter**:

```
admin@firewall-a> debug dataplane packet-diag show filter-marked-session [Enter]

No Active Marked Sessions
admin@firewall-a>
```

19. Verify that there are "**No Active Marked Sessions**".
20. Open the testing browser and go to **http://www.test.lab/blue**.
    Leave the window open.
21. Return to the CLI, press the **Up Arrow** key to reload the previous command (**debug dataplane packet-diag show filter-marked-session**), and then press **Enter**:

```
admin@firewall-a> debug dataplane packet-diag show filter-marked-session [Enter]

--------------------------------------------------------------------------------
ID          Application    State   Type Flag  Src[Sport]/Zone/Proto (translated IP[Port])
Vsys                                          Dst[Dport]/Zone (translated IP[Port])
--------------------------------------------------------------------------------
24934       web-browsing   ACTIVE  FLOW  NS   192.168.1.20[32636]/inside/6   (172.16.16.16[20865])
vsys1                                         192.168.50.10[80]/dmz  (192.168.50.10[80])
```

    Note: If you do not see a marked session, make sure that you have closed and opened the testing browser. If you do not close the browser, it will reload the page from cache. Even if you change the target page, for example, from **www.test.lab/blue** to **www.test.lab/orange**, the browser may reuse the existing session. The firewall typically will mark only *new* sessions after turning on filtering.
22. Close the testing browser.
    This action terminates the TCP connection to the website. In the CLI, you can rerun the **debug dataplane packet-diag show filter-marked-session** command and see that the firewall no longer displays the session as marked.
23. Leave the SSH connection open.

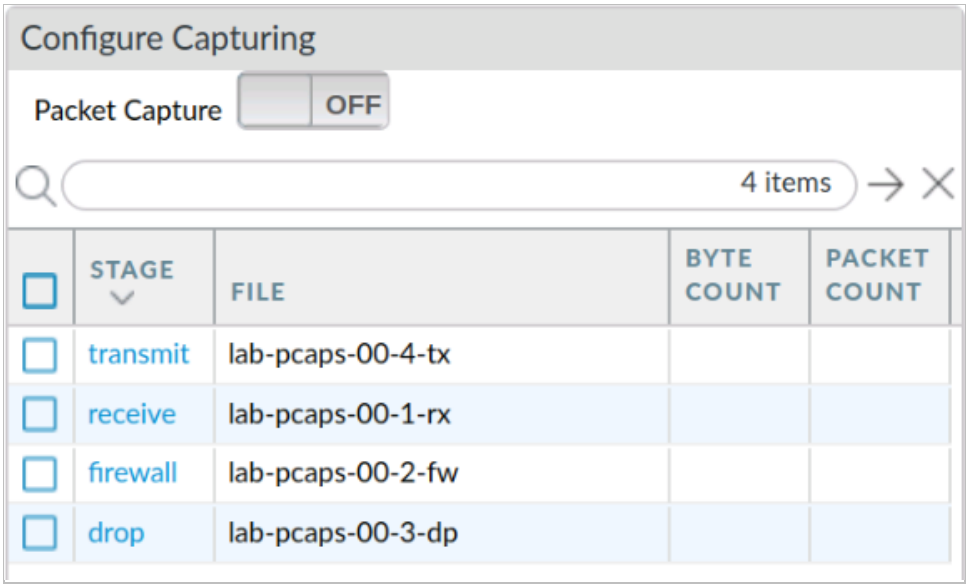
3.4 Configure Capture Stages

24. In the firewall's web interface, under Configure Capturing, click **Add** to create the following capture filters:

Stage	File	Packet Count	Byte Count
receive	**lab-pcaps-00-1-rx**	[blank]	[blank]
firewall	**lab-pcaps-00-2-fw**	[blank]	[blank]

Stage	File	Packet Count	Byte Count
drop	**lab-pcaps-00-3-dp**	[blank]	[blank]
transmit	**lab-pcaps-00-4-tx**	[blank]	[blank]

Tip: Use keyboard shortcuts (Ctrl+C and Ctrl+V) to copy and paste "**lab-pcaps-00-**" from your configuration of the first stage to the configuration of subsequent stages:



25. Go to the firewall CLI and type:
debug dataplane packet-diag and press the **spacebar**, and then press **Tab**.
This procedure displays the basic command line options for packet diagnostics:

```
admin@firewall-a> debug dataplane packet-diag [Tab]
> aggregate-logs   Aggregate packetdiag log
> clear            Clear packet-related diagnosis parameters
> set              Set packet-related debugging parameters
> show             Show packet-related diagnosis information
```

Packet diagnostics includes several packet-trace logging options in addition to packet captures. These other diagnostic features generate a separate log file for every security-processor core on the data plane that processes traffic that belongs to a marked session. These logs are aggregated automatically when you generate a Tech Support File. You can aggregate them manually by using the **aggregate-logs** option. After you use the **aggregate-logs** option, then you can export the packet-diagnostics log manually. You do not need to use the **aggregate-logs** option for packet captures.

You can use the **clear** option to remove configurations created using the **set** option. You can use the **show** option to display current settings.

26. Type:
`debug dataplane packet-diag show setting` and press **Enter**.
This command displays the current packet-diagnostics settings:

```
admin@firewall-a> debug dataplane packet-diag show setting

--------------------------------------------------------------------------------
Packet diagnosis setting:
--------------------------------------------------------------------------------
Packet filter
  Enabled:                   yes
  Match pre-parsed packet:   no
  Filter offload:            yes
  Index 1: 192.168.1.20/32[0]->192.168.50.10/32[80], proto 6
           ingress-interface any, egress-interface any, exclude non-IP
--------------------------------------------------------------------------------
Logging
  Enabled:                   no
  Log-throttle:              no
  Sync-log-by-ticks:         yes
  Features:
  Counters:
  Timeout duration:          60 seconds
  Buffer threshold:          80%
  CPU threshold:             80%
--------------------------------------------------------------------------------
Packet capture
  Enabled:                   no
  Snaplen:                   0
  Username:
  Stage receive           :  file lab-pcaps-00-1-rx
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage firewall          :  file lab-pcaps-00-2-fw
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage transmit          :  file lab-pcaps-00-4-tx
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage drop              :  file lab-pcaps-00-3-dp
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
--------------------------------------------------------------------------------
```

In the **`Packet filter`** section, notice that the blank setting that you configured for the source port using the web interface is **`0`** in the CLI output. Also notice that the “none” or blank setting that you configured for the ingress interface using the web interface is presented in the CLI output as **any**. Blank or unspecified parameters in the capture filters typically translate to **any** for the actual match filters.

Note: When you are in a production environment, before you turn on packet capture, Palo Alto Networks recommends that you check for the existence of any filter-marked sessions that you

do not want to capture. The firewall will start to capture all marked sessions immediately after you turn on packet capture.

3.5 Clear Marked Sessions

In the following steps, you will use the CLI to discover and execute the command to clear existing marked sessions.

27. Identify the options for the command **`debug dataplane packet-diag clear [Tab]`**.

    To get the information you need, use the autocomplete feature and the suggestions that the CLI provides in response to pressing the **Tab** key at the end of complete options. You can use the following screenshot as a guide:

```
admin@firewall-a> debug dataplane packet-diag clear [Tab]
> all                     Clear all settings and turn off log/capture
> capture                 capture setting
> filter                  Packet filter
> filter-marked-session   Unmark session for debug
> log                     log setting

admin@firewall-a> debug dataplane packet-diag clear filter-marked-session [Tab]
> all   Unmark all sessions in debug
> id    Unmark a specific session in debug
```

28. Type:
    `debug dataplane packet-diag clear filter-marked-session all` and press **Enter**:

```
admin@firewall-a> debug dataplane packet-diag clear filter-marked-session all

Unmark All sessions in packet debug
```

    In a production environment, you should check for and clear unwanted marked sessions as a best practice, especially in environments with multiple administrators. Even when the likelihood of an active collision of activity is low, the possibility of unexpected impacts from prior troubleshooting activity should be eliminated.

    The option to specify an **`id`** to clear the filter mark on a single session typically is useful only when you are focused on long-running sessions, such as VPN connections. In most cases, you will use the **`all`** option.

3.6 Turn On Packet Capture and Capture Packets

29. Turn on packet capture by typing:
    `debug dataplane packet-diag set capture on` and press **Enter**:

```
admin@firewall-a> debug dataplane packet-diag set capture on

Packet capture is enabled
```

30. Go to the web interface and select **Monitor** > **Packet Capture** to refresh the page.
31. Verify that the graphical toggle switch for **Packet Capture** displays as **ON**.
32. Launch the testing browser and go to **`http://www.test.lab/blue`**.
33. Go to the firewall CLI and type:
    `debug dataplane packet-diag show setting` and press **Enter**.
    This command can help you check the status of the packet capture.
    In the output, review the **`Packet capture`** section and confirm that the firewall shows a number greater than zero for packets and bytes captured:

```
[...]
--------------------------------------------------------------------------------
Packet capture
  Enabled:                   yes
  Snaplen:                   0
  Username:
  Stage receive           :  file lab-pcaps-00-1-rx
    Captured:     packets - 5          bytes - 541
    Maximum:      packets - 0          bytes - 0
  Stage firewall          :  file lab-pcaps-00-2-fw
    Captured:     packets - 5          bytes - 541
    Maximum:      packets - 0          bytes - 0
  Stage transmit          :  file lab-pcaps-00-4-tx
    Captured:     packets - 4          bytes - 1138
    Maximum:      packets - 0          bytes - 0
  Stage drop              :  file lab-pcaps-00-3-dp
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
--------------------------------------------------------------------------------
```

34. Type:
    `debug dataplane packet-diag show filter-marked-session`.
    Look for **`No Active Marked Sessions`**.
35. Repeat the command every 10 to 15 seconds until the session times out:

```
admin@firewall-a> debug dataplane packet-diag show filter-marked-session

No Active Marked Sessions
```

    The default session timeout for this type of connection is 120 seconds.

36. Execute the command:
    `debug dataplane packet-diag set capture off`.
    This command turns off packet capture.

37. Use the **`debug dataplane packet-diag show setting`** command to verify that packet capture is no longer enabled.

In the following steps, you will view the pcap in the CLI.

38. In the firewall CLI type:
    `view-pcap` (add a space at the end if you did not use autocomplete) and press **Tab**.
    This procedure displays the options available for viewing pcaps:

```
admin@firewall-a> view-pcap [Tab]
+ absolute-seq            Print absolute TCP sequence numbers
+ delta                   Print a delta (in ms) between current and previous line
+ follow                  Monitor pcap file in real time
+ hex                     Print each packet (minus link header) in hex
+ hex-ascii               Print each packet (minus link header) in hex and ASCII
+ hex-ascii-link          Print each packet (including link header) in hex and ASCII
+ hex-link                Print each packet (including link header) in hex
+ link-header             Print the link-level header on each dump line
+ no-dns-lookup           Don't convert host addresses to names
+ no-port-lookup          Don't convert protocol and port numbers to names
+ no-qualification        Don't print domain name qualification of host names
+ no-timestamp            Don't print a timestamp
+ timestamp               Print a timestamp proceeded by date
+ undecoded-NFS           Print undecoded NFS handles
+ unformatted-timestamp   Print an unformatted timestamp
+ verbose                 Verbose output
+ verbose+                Even more verbose output
+ verbose++               Lots of verbose output
> application-pcap        YYYYMMDD/filename
> debug-pcap              packet capture generated for purpose of debugging daemons
> filter-pcap             YYYYMMDD/filename
> mgmt-pcap               packet capture generated from management interface
> threat                  pcap id
```

39. Type:
    `view-pcap filter-pcap` (add a space at the end if needed) and press **Tab**.
    This procedure lists the filter-pcap files available for display:

```
admin@firewall-a> view-pcap filter-pcap [Tab]
lab-pcaps-00-1-rx  lab-pcaps-00-4-tx  lab-pcaps-00-2-fw
```

40. Type:
    `view-pcap filter-pcap lab-pcaps-00-1-rx` and press **Enter**.
    This command displays the receive-stage pcap:

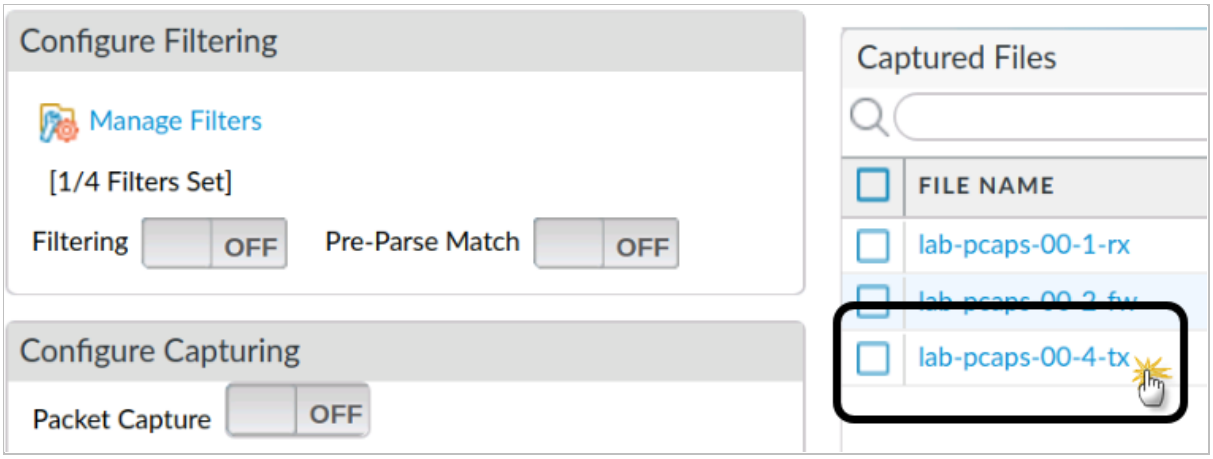
```
admin@firewall-a> view-pcap filter-pcap lab-pcaps-00-1-rx
19:18:42.293284 IP 192.168.1.20.17706 > 192.168.50.10.http: [SEW], seq 2756844710, win
65535, options [mss 1460,nop,wscale 8,nop,nop,sackOK], length 0
19:18:42.293984 IP 192.168.1.20.17706 > 192.168.50.10.http: [.], ack 2577168401 win 1024
19:18:42.293996 IP 192.168.1.20.17706 > 192.168.50.10.http: [P.], 0:259(259) ack 1 w[...]
```

```
19:18:42.294045 IP 192.168.1.20.17706 > 192.168.50.10.http: [.], ack 911 win 1020
19:18:47.281260 IP 192.168.1.20.17706 > 192.168.50.10.http: [.], ack 912 win 1020
19:20:42.272247 IP 192.168.1.20.17706 > 192.168.50.10.http: [F. ], 259:259(0) ack 91[...]
```

41. Type:
    `view-pcap filter-pcap lab-pcaps-00-4-tx` and press **Enter**.
    This command displays the transmit-stage pcap:

```
admin@firewall-a> view-pcap filter-pcap lab-pcaps-00-4-tx
19:18:42.293966 IP 192.168.50.10.http > 192.168.1.20.17706: [S.], seq 2577168400, ack
2756844711, win 29200, options [mss 1460,nop,nop,sackOK,nop,wscale 6], length 0
19:18:42.294026 IP 192.168.50.10.http > 192.168.1.20.17706: [.] ack 260 win 473
19:18:42.294036 IP 192.168.50.10.http > 192.168.1.20.17706: [P.] 1:911(910) ack 260 [...]
19:18:47.281209 IP 192.168.50.10.http > 192.168.1.20.17706: [F.] 911:911(0) ack 260 [...]
19:20:42.272319 IP 192.168.50.10.http > 192.168.1.20.17706: [R], seq 2577169312, win[...]
```

42. Close the testing browser.
43. Go to the firewall web interface and click **Monitor > Packet Capture** to refresh the page.
44. Turn filtering **OFF**.
45. Click to download all three pcaps (save them to the **`/home/lab-user/Downloads/`** folder):



Note: If the captured files do not appear, click the **refresh** button until they do appear.

3.7 Analyze the Pcaps

46. Open all three pcaps in **Wireshark**.
    If needed, browse to **/home/lab-users/Downloads/** and double-click each file to open it in Wireshark.
47. Formulate answers to the following questions:

    How many packets were captured at each stage?

    Do you see the same number of packets in the receive and firewall stages?

Are there any differences in the IP addressing that you see in each stage, compared to the IP addresses that you set in the filters?

Answer: No and yes. Source and destination IP addresses should correlate precisely to what you specified in the filters. However, you may notice that, even though the transmit-stage pcap source and destination IP addresses are reversed, the filters still match to this traffic.

Which packets are missing?

Answer: Server-to-client (responder) traffic is missing from the receive-stage pcap and the firewall-stage pcap. Client-to-server (initiator) traffic is missing from the transmit-stage pcap. Notice, for example, that you cannot find the SYN-ACK packet for the connection setup in the receive-stage pcap or firewall-stage pcap. In the transmit-stage pcap, there will be no SYN packet.

48. Before you start the next section, close all the open **Wireshark** windows.

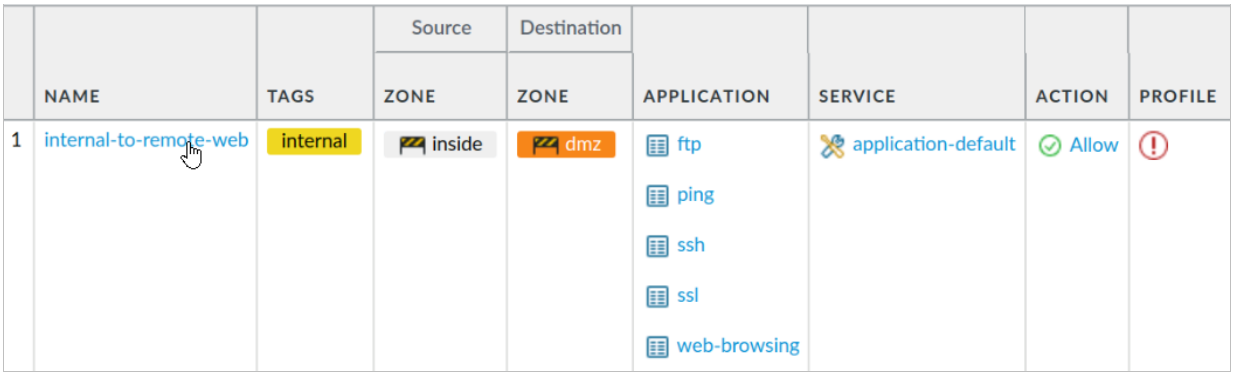
3.8 Add a Security Policy Configuration to Drop Traffic

You want to run a packet capture to see what happens when the firewall drops traffic based on Layer 7 inspection activity.

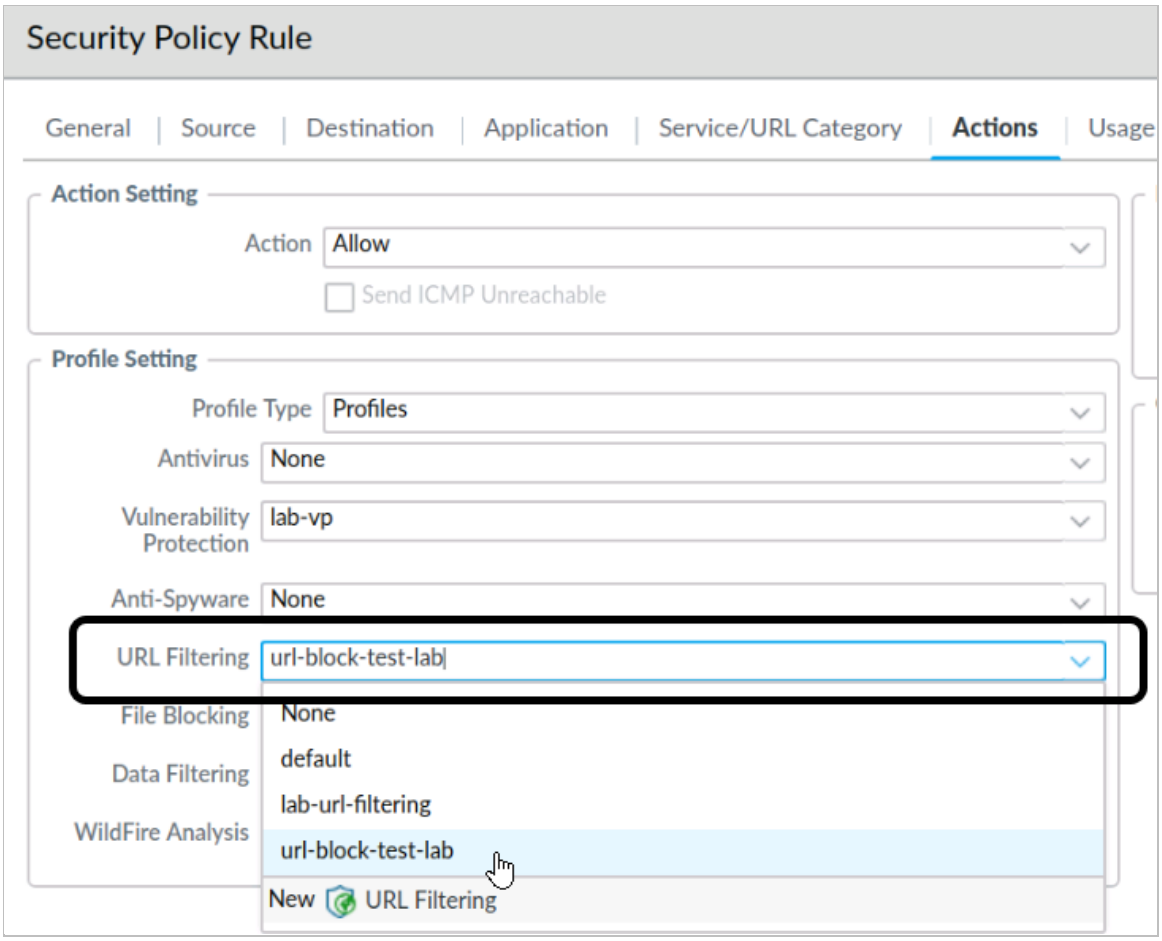
If you can cause the Layer 7 inspection capability of the firewall to drop a packet while you are running a packet capture, you subsequently can analyze the drop-stage pcap to discover the state of the packet after the capture point of the firewall stage and before the packet reaches the transmit stage. You will discover that the address headers for packets that are dropped by Layer 7 inspection are in a post-NAT state.

In the following steps, you will configure the Security policy to drop the connection to the www.test.lab web server by adding a preconfigured URL Filtering profile to the Security policy rule that matches traffic to the target server.

49. In the firewall web interface, go to **Policies** > **Security**.
50. Click the **internal-to-remote-web** rule:



51. In the **Security Policy Rule** window, click the **Actions** tab.
52. Click the **URL Filtering** drop-down menu, select **url-block-test-lab**, and then click **OK**:



53. Click **OK** and then **commit** the configuration.

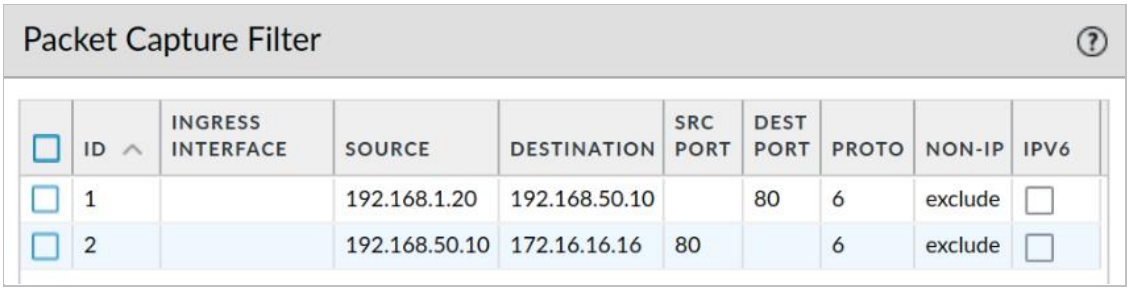
3.9 Reconfigure the Filter

In the following steps, you will add a second packet capture filter that will match to the return traffic. The destination address for return traffic must be defined because the firewall is applying source NAT to the session. In this case, the destination address that the firewall will receive on the ingress port of the server-responder will be the NAT address 172.16.16.16.

54. Go to **Monitor > Packet Capture** and click **Manage Filters**.

55. **Add** a filter for the return traffic and then click **OK**:

Parameter	Value
Id	**2**
Ingress Interface	none [blank]
Source	**192.168.50.10**
Destination	**172.16.16.16**
Src Port	**80**
Dest Port	[blank]
Proto	**6**
Non-IP	**exclude**
IPv6	not selected



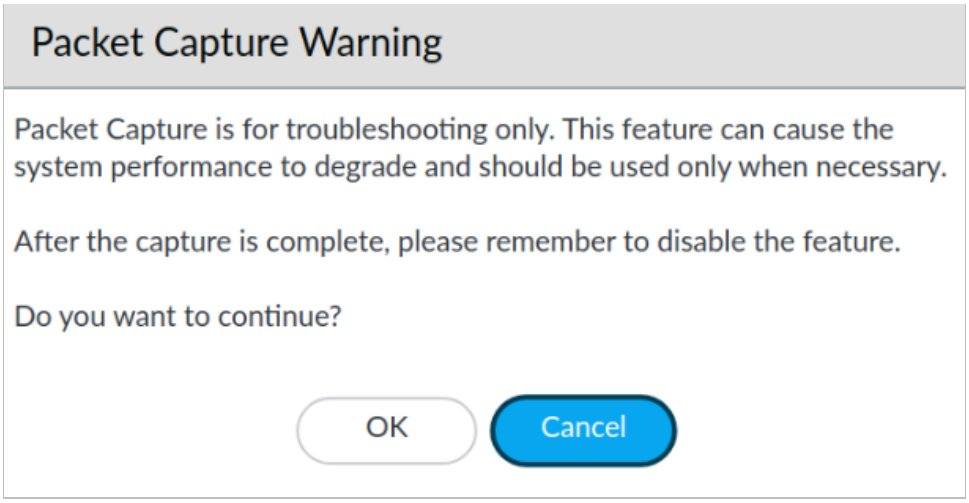
56. Use the web interface to update the name of each packet filter stage, according to the following table. Click the **Stage** name of each existing configuration to edit it:

Stage	File	Packet Count	Byte Count
receive	**lab-pcaps-01-1-rxtx**	[blank]	[blank]
transmit	**lab-pcaps-01-1-rxtx**	[blank]	[blank]
firewall	**lab-pcaps-01-2-fw**	[blank]	[blank]
drop	**lab-pcaps-01-3-dp**	[blank]	[blank]

Important: For this packet capture, you will specify *the same filename* for the receive and the transmit stages. This naming will cause the firewall to merge the receive and transmit stages automatically so that you can follow the c2s and s2c flows within the same pcap:

	STAGE	FILE	BYTE COUNT	PACKET COUNT
☐	transmit	lab-pcaps-01-1-rxtx		
☐	receive	lab-pcaps-01-1-rxtx		
☐	firewall	lab-pcaps-01-2-fw		
☐	drop	lab-pcaps-01-3-dp		

57. Use the web interface to turn **Filtering ON**.
58. *The next steps assume that you just used the CLI to clear all the currently marked sessions.*
59. Use the web interface to turn **Packet Capture ON**.
60. After the **Packet Capture Warning** appears, click **OK**:

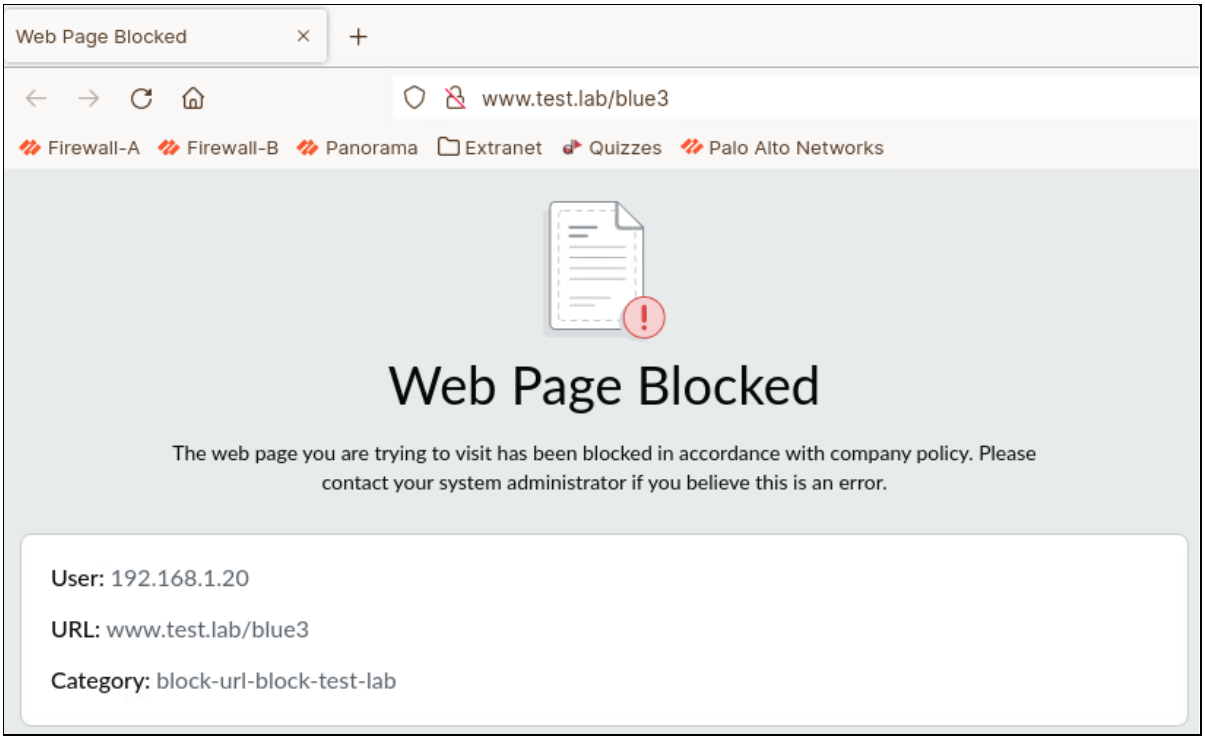


3.10 Capture a New Session and Download the Pcaps

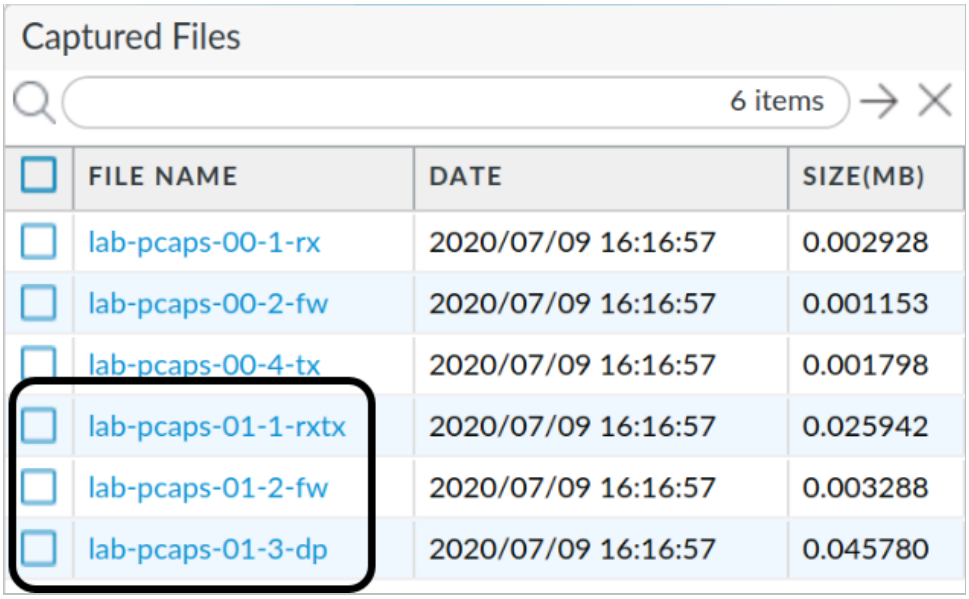
61. Open a new instance of the testing browser.
62. Go to **`http://www.test.lab/blue3`**.

    The **`blue3`** file is about 4,200 bytes and, therefore, requires three packets to transfer. The **`blue3`** file gives you the chance to block a session that requires more than just one packet to transfer the requested HTML file, as does the **`blue`** file.

You should receive a block page similar to the following:



63. Close the testing browser.
64. Go to the web interface of the firewall and click **Monitor** > **Packet Capture** to refresh the page.
65. In the **Captured Files** section, verify that new pcap files have been captured:



Note: If you don't' see the same list, refresh the page.

66. Use the web interface to turn **Packet Capture OFF** and then **Filtering OFF**.

    Important: Always turn off packet capture *before* you turn off filtering. If filtering is off and packet capture is on, packet capture will capture *all packets*. Capture of all packets, even briefly, in production increases capture-file size and also could degrade overall performance.

3.11 Analyze the Pcaps

67. Click to download and open all three pcaps in **Wireshark**.

Drop-stage pcap: Formulate answers to the following questions:

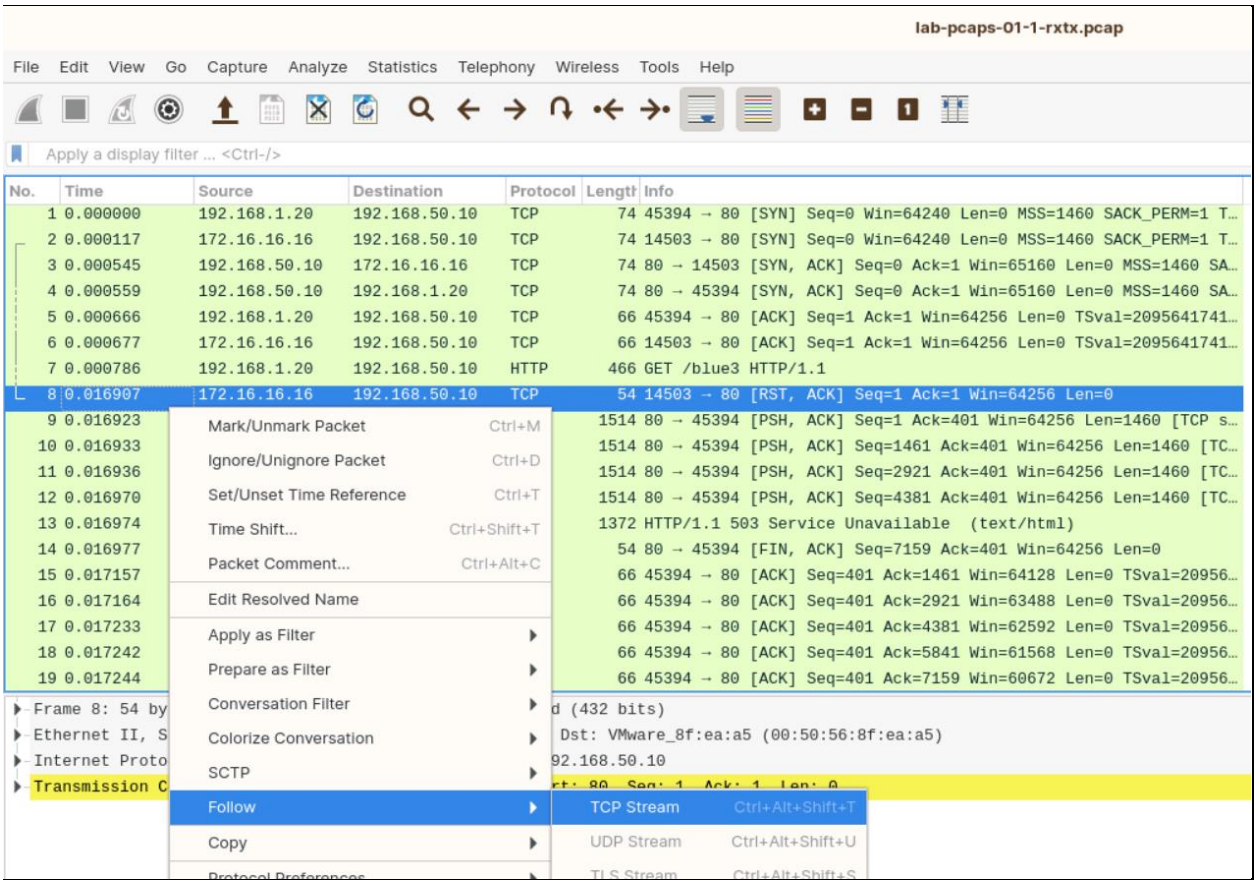
68. Why is the source IP address for the HTTP GET packet **`172.16.16.16`**?

69. From which processing stage are the ACK and FIN-ACK packets from the *client* most likely being dropped?

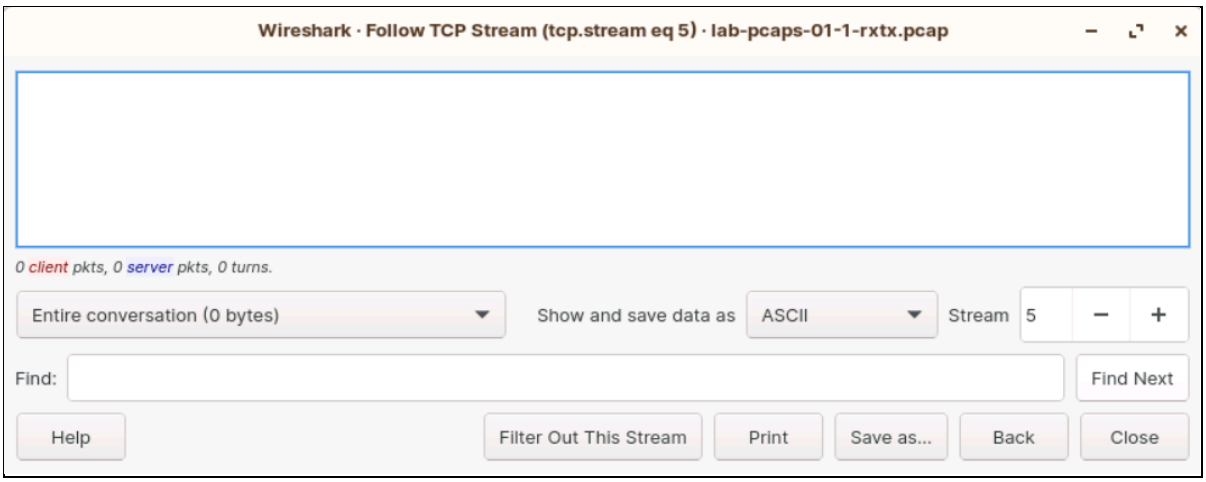
70. Why are packets from the client being dropped?

Receive-transmit pcap: Follow the instructions and answer the subsequent questions posed:

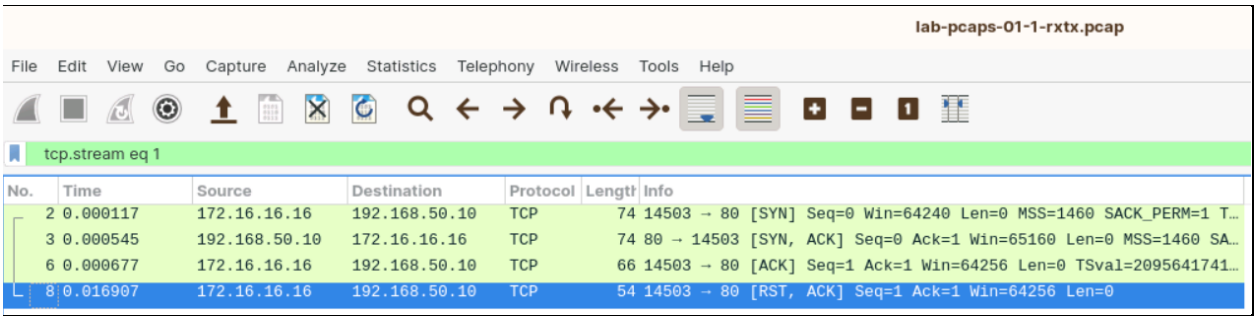
71. Find the **RST-ACK** packet sent to **`192.168.50.10`**.
72. Right-click **RST-ACK** packet and choose **Follow > TCP Stream**:



The session-content window will be empty:



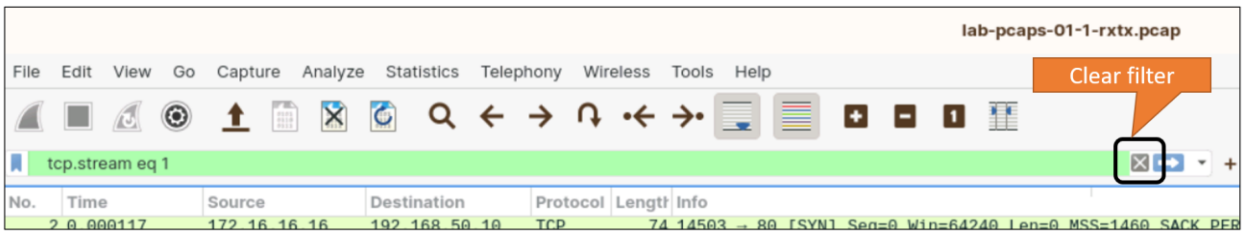
73. Click to **close** the **Wireshark – Follow TCP Stream** window.
74. Examine the packets displayed that are filtered by the **`tcp.stream.eq.<#>`** string:



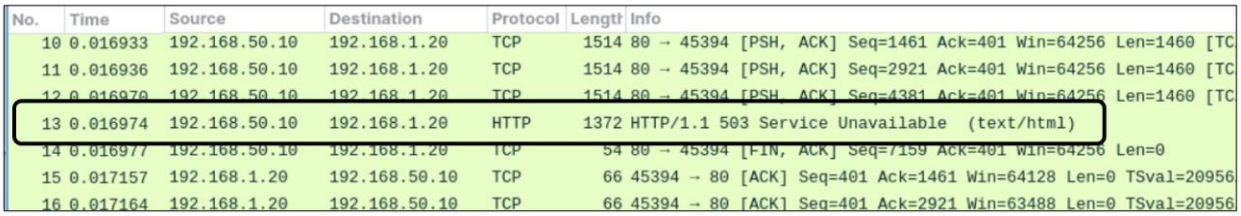
75. Why was the **Follow TCP Stream** window empty?

    Answer: Because none of the packets that belong to this stream contains any payload.

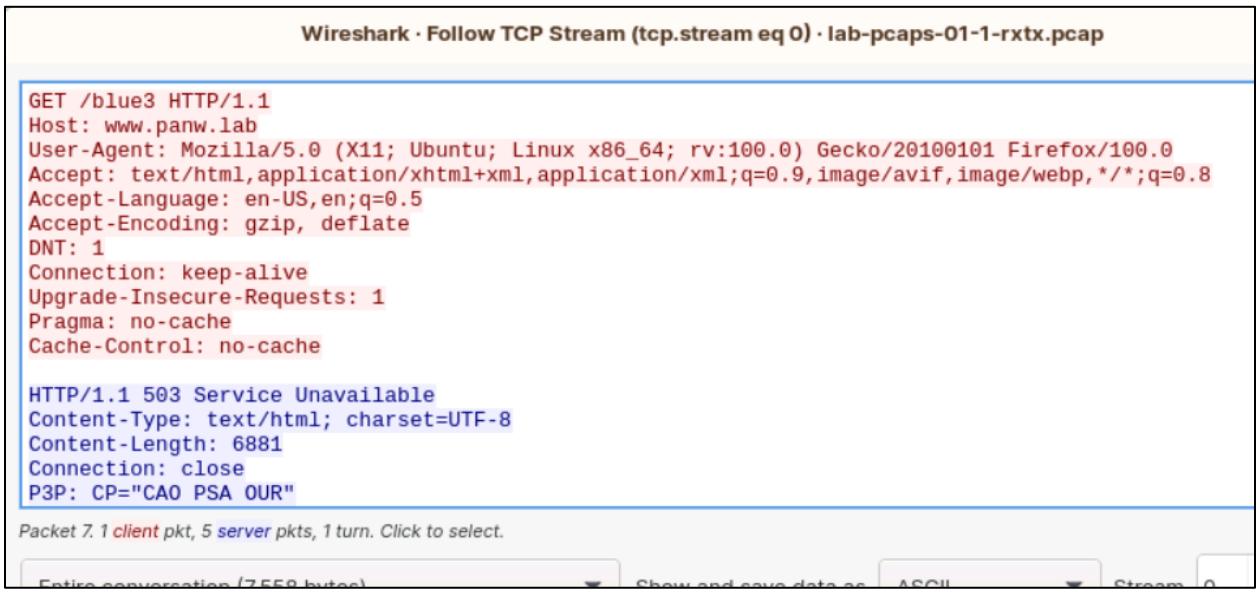
76. Click **Clear** to the right of the filter text box to clear the filter:



77. Find the **HTTP Service Unavailable** packet sent to **`192.168.1.20`**:

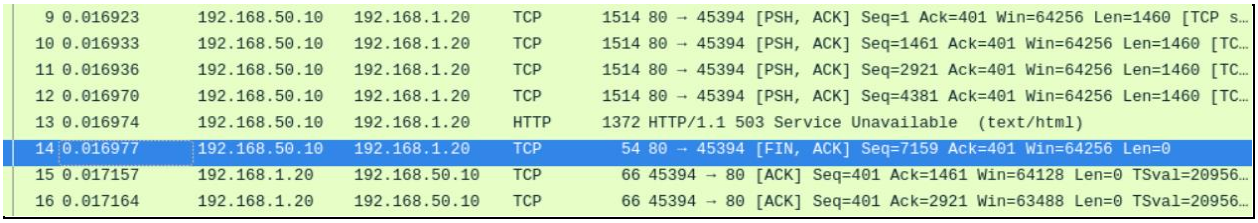


78. Right-click the **HTTP Service Unavailable** packet and choose **Follow > TCP Stream**:



79. Which HTML content does the **503 Service Unavailable** packet contain?
    What, if any, of the content of the **503 Service Unavailable** packet would the end user see?

80. Click to **close** the **Wireshark – Follow TCP Stream** window.
81. Examine the packets displayed that are filtered by the **`tcp.stream.eq.<#>`** string:



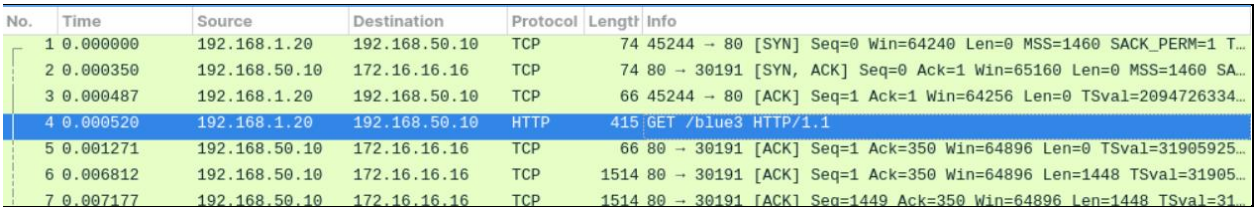
9	0.016923	192.168.50.10	192.168.1.20	TCP	1514 80 → 45394 [PSH, ACK] Seq=1 Ack=401 Win=64256 Len=1460 [TCP s…
10	0.016933	192.168.50.10	192.168.1.20	TCP	1514 80 → 45394 [PSH, ACK] Seq=1461 Ack=401 Win=64256 Len=1460 [TC…
11	0.016936	192.168.50.10	192.168.1.20	TCP	1514 80 → 45394 [PSH, ACK] Seq=2921 Ack=401 Win=64256 Len=1460 [TC…
12	0.016970	192.168.50.10	192.168.1.20	TCP	1514 80 → 45394 [PSH, ACK] Seq=4381 Ack=401 Win=64256 Len=1460 [TC…
13	0.016974	192.168.50.10	192.168.1.20	HTTP	1372 HTTP/1.1 503 Service Unavailable (text/html)
14	0.016977	192.168.50.10	192.168.1.20	TCP	54 80 → 45394 [FIN, ACK] Seq=7159 Ack=401 Win=64256 Len=0
15	0.017157	192.168.1.20	192.168.50.10	TCP	66 45394 → 80 [ACK] Seq=401 Ack=1461 Win=64128 Len=0 TSval=20956…
16	0.017164	192.168.1.20	192.168.50.10	TCP	66 45394 → 80 [ACK] Seq=401 Ack=2921 Win=63488 Len=0 TSval=20956…

82. In the stream shown, which device sends the first **FIN-ACK** packet?
    You may assume that the device that sends the first FIN-ACK packet is the firewall.
    Is there any way to know for sure?

83. What happens to the **ACK** and **FIN-ACK** responses sent by the client?
    Are these the same packets that you see in the drop-stage pcap?
    If you think that they are, how would you know?

Firewall-stage pcap: Follow the instructions and answer the subsequent questions posed:

84. Look for the **503 Service Unavailable** packet in the **firewall-stage** pcap:



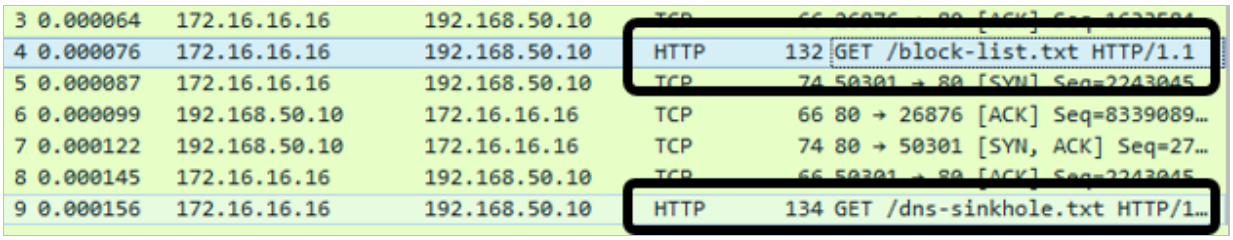
No.	Time	Source	Destination	Protocol	Lengt	Info
1	0.000000	192.168.1.20	192.168.50.10	TCP	74	45244 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T…
2	0.000350	192.168.50.10	172.16.16.16	TCP	74	80 → 30191 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA…
3	0.000487	192.168.1.20	192.168.50.10	TCP	66	45244 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=2094726334…
4	0.000520	192.168.1.20	192.168.50.10	HTTP	415	GET /blue3 HTTP/1.1
5	0.001271	192.168.50.10	172.16.16.16	TCP	66	80 → 30191 [ACK] Seq=1 Ack=350 Win=64896 Len=0 TSval=31905925…
6	0.006812	192.168.50.10	172.16.16.16	TCP	1514	80 → 30191 [ACK] Seq=1 Ack=350 Win=64896 Len=1448 TSval=31905…
7	0.007177	192.168.50.10	172.16.16.16	TCP	1514	80 → 30191 [ACK] Seq=1449 Ack=350 Win=64896 Len=1448 TSval=31…

85. The **503 Service Unavailable** packet does not appear in the firewall-stage pcap.
    Why?

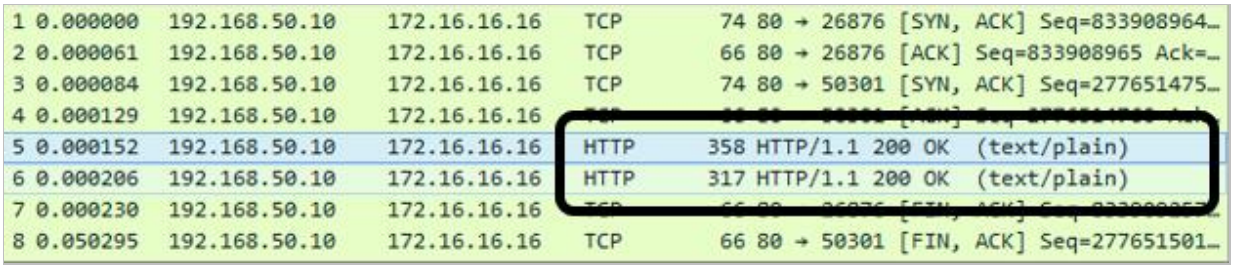
86. For this session, how many packets did the firewall stage see from the server?
    Why only one packet?

Optional analysis:

Depending on when you run your packet capture, and for how long, you may see additional traffic generated by the firewall itself that includes HTTP GET requests for **block-list**, **DNS-sinkhole**, or other use cases:



87. Compare the preceding **receive-transmit** pcap with the following **firewall-stage pcap** for these sessions:



88. Notice that the **firewall-stage** pcap includes HTTP GET requests.

    Why is this traffic being seen on the firewall's data plane?

    Does your analysis provide any insights for troubleshooting other firewall host-generated and host-inbound traffic?

3.12 Clean Up Your Lab Environment

89. Clean up your lab environment by closing all open windows such as Remmina and Wireshark and deleting pcap files from the firewall.

3.13 Reference Information

- If NAT is involved and you want to be able to see bidirectional "streams" in Wireshark, you must merge the transmit and receive pcaps. If you have separate receive-stage and transmit-stage pcap files, you can use the merge function built into Wireshark. As demonstrated in the lab, you also can use the firewall to merge receive-stage and transmit-stage packet captures by configuring each stage with the same filename.
- You can use the IP identification number (TCP ID) of a packet to make packet-level correlations among multiple pcap files, which could be collected from multiple sources in addition to the firewall, and other packet-trace log files, such as flow-basic logs.
- You can configure your packet analyzer to display *absolute* TCP sequence numbers, instead of *relative* TCP sequence numbers. When you work with larger pcap files and other packet-trace logs, if you use the absolute TCP sequence number to make packet-

level correlations, you may be more likely to produce unique matches because the TCP sequence number (32 bytes) is larger than the IP ID (16 bytes).

Stop. This is the end of the lab.

4. Lab: Flow Basic

Lab Objectives

- Identify a connectivity problem and use flow basic to help fix it
- Configure packet filters to capture traffic for debug-level packet-trace logs
- Enable the flow basic logging feature, capture packets, and log the packet flow
- Use your knowledge and skills to interpret the packet flow and solve the problem

Lab Scenario

Your organization hosts an FTP service for file transfers from partners. External users can get to the FTP server without problem. However, your internal users cannot connect to the server. To complete this lab, you must do the following:

- Verify the problem.
- Look for evidence of the problem in the Traffic logs.
- Run packet-diagnostics logging the flow basic feature to diagnose the issue.
- Implement a solution to the problem and validate it.

4.1 Load the Lab Config File and Start the FTP Server

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-04.xml**

   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. **Commit** the configuration after the load task is complete.

4.2 Verify External Connectivity to the FTP Server

Your support ticket says that external users can connect to the FTP server without problem. In the following steps, you will use the server in the DMZ zone to verify that inbound connections can be made.

3. On the student desktop, double-click the **Remmina** shortcut.
4. Double-click the Server-Extranet configuration in the **Remmina Remote Desktop Client** window.

   This configuration will connect you to a server in the DMZ.

5. Type: **`curl -u “paloalto42:Pal0Alt0!” ftp://ftp.test.lab/ftptest.txt`** and press **Enter**.

This command uses the **curl** utility to get a test file from the FTP server.

6. Verify that the content of the test file is displayed as in the following example:

```
paloalto42@extranet1:~$ curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
  ===============
  You have connected to and retrieved a file
  from the internal ftp.test.lab server.
  Real IP: 192.168.1.20
  ===============
  This is a test file for
  validating basic ftp connectivity.
  ===============
  END OF FILE.
  ===============
paloalto42@extranet1:~$
```

Tip: If you receive a **550** error, you probably typed the filename incorrectly. If the request times out or results in an "Unknown error," check the spelling of the hostname.

7. Leave open the **Remmina** window that contains the SSH connection to the DMZ server. You will use the SSH connection to the server again in a later step.

4.3 Verify the Problem with the Internal Client

8. Open a terminal window on the client desktop.
9. Enter the following command to test access to the ftp.test.lab host:

```
curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

```
lab-user@client-a:~$ curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

10. The process will eventually fail, so you can use **CTRL+C** after a few moments to stop the connection attempt.

    Note: One or two minutes may elapse before the connection times out.

11. Try to connect without specifying a filename with this URL:

```
lab-user@client-a:~$ curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/
```

This URL eliminates the filename as the source of the problem, but the connection attempt still does not result in a connection.

12. This connection attempt will also fail, so you can use **CTRL+C** after a few moments to stop the process.

13. Use the **`Real IP`** address of the client host in the connection – **192.168.1.20**.

```
lab-user@client-a:~$ curl -u "paloalto42:Pal0Alt0!" ftp://192.168.1.20/ftptest.txt
  ===============
  You have connected to and retrieved a file
  from the internal ftp.test.lab server.
  Real IP: 192.168.1.20
  ===============
  This is a test file for
  validating basic ftp connectivity.
  ===============
  END OF FILE.
  ===============
lab-user@client-a:~$
```

Success! You should see the contents of the file. Perhaps there is something wrong in the name resolution. Which IP address is the Extranet-Server (ftp server) resolving to?

14. On the student desktop, open a Terminal and type: **`ping ftp.test.lab`**. After a few seconds, press **Ctrl+C**:

```
C:\home\lab-user\Desktop\Lab-Files> ping ftp.test.lab
PING ftp.test.lab (172.22.22.2) 56(84) bytes of data.
^C
--- ftp.test.lab ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 5999ms

C:\home\lab-user\Desktop\Lab-Files>
```

Note: The ping will *not* succeed.

Note the IP address **172.22.22.2**.

Check to see if the address resolution to **172.22.22.2** is the same on the Server-Extranet.

15. On the Server-Extranet, open the SSH connection and type: **`ping ftp.test.lab`**. After a few seconds, press **Ctrl+C**:

```
paloalto42@extranet1:~$ ping ftp.test.lab
PING ftp.test.lab (172.22.22.2) 56(84) bytes of data.
^C
---ftp.test.lab ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7055ms

paloalto42@extranet1:~$
```

Note: The ping will *not* succeed.

Both clients are trying to connect to the same IP address (**172.22.22.2**). The ping messages sent to IP address 172.22.22.2 from the student desktop and the Extranet hosts both result in 100%

packet loss. However, we are using ping only as a simple way to see the result of DNS resolutions.

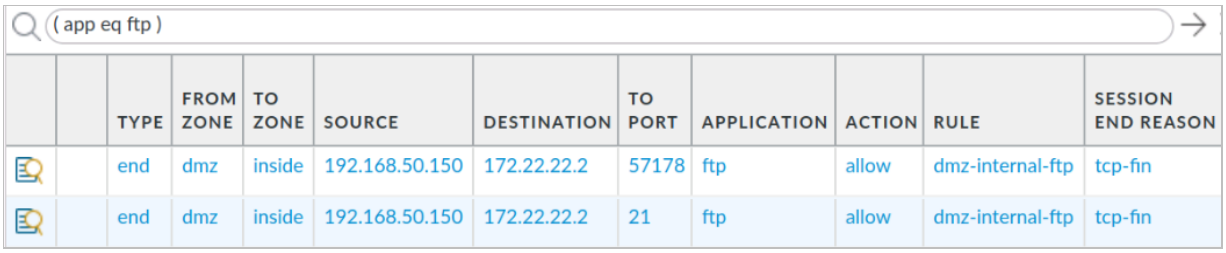
You already know that the firewall must be using destination NAT to provide external access to the server, because the "real" or internal IP address of the server (192.168.1.20) is different than the "public" or external address (172.22.22.2). Destination NAT translations typically are limited to specific services. You cannot assume that ping attempts would be routed to the same internal server. Thus, the ping results of 100% packet loss are not necessarily an issue.

Internally, you can make direct FTP connections to the address **192.168.1.20** but your users need to be able to use all their existing links and shortcuts that are based on DNS name.

You still need to find the root cause of the problem.

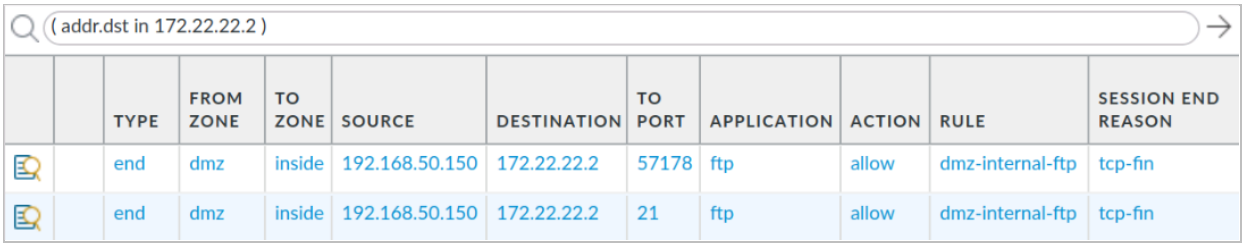
4.4 Examine Firewall Traffic Logs and Threat Logs

16. In the web interface of the firewall, go to **Monitor** > **Logs** > **Traffic**.
17. In the filter box, type:
    `(app eq ftp)` and press **Enter** or click → (**Apply Filter**):



		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		end	dmz	inside	192.168.50.150	172.22.22.2	57178	ftp	allow	dmz-internal-ftp	tcp-fin
		end	dmz	inside	192.168.50.150	172.22.22.2	21	ftp	allow	dmz-internal-ftp	tcp-fin

18. Scan the **Source** column for the IP address **`192.168.1.20`**.
    Unfortunately, there are no Traffic log entries that match this source address. What does this absence of log entries mean? Perhaps the application was not detected, and the search filter is eliminating the connection attempt from view.
19. To the right of the filter box, click **X** (**Clear Filter**).
    Or you can manually select and delete the existing filter text.
20. In the **Destination** column, click a destination address (**172.22.22.2**) to auto-populate the filter, then press **Enter** or click → (**Apply Filter**):

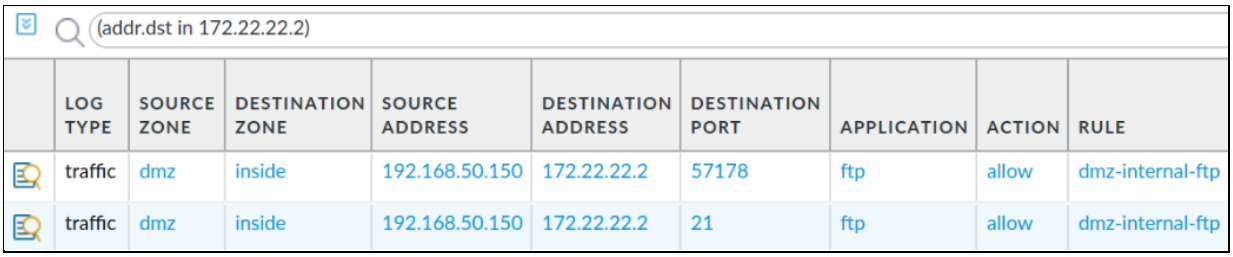


		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		end	dmz	inside	192.168.50.150	172.22.22.2	57178	ftp	allow	dmz-internal-ftp	tcp-fin
		end	dmz	inside	192.168.50.150	172.22.22.2	21	ftp	allow	dmz-internal-ftp	tcp-fin

Still there are no Traffic log entries for any connections from **192.168.1.20**.

21. Click **Monitor** > **Logs** > **Unified** and enter the same filter:
    `(addr.dst in 172.22.22.2)`

    This action will check for any useful log information across multiple log types:


(addr.dst in 172.22.22.2)

	LOG TYPE	SOURCE ZONE	DESTINATION ZONE	SOURCE ADDRESS	DESTINATION ADDRESS	DESTINATION PORT	APPLICATION	ACTION	RULE
	traffic	dmz	inside	192.168.50.150	172.22.22.2	57178	ftp	allow	dmz-internal-ftp
	traffic	dmz	inside	192.168.50.150	172.22.22.2	21	ftp	allow	dmz-internal-ftp

    Still there are no logs with a source address of **192.168.1.20**. If you search for destination IP address **172.22.22.2**, you also will not be able to find any logs.

22. (Optional) Open a **Remmina** connection to the firewall and use the following commands in the CLI to follow this same workflow from the command line:

    `> show log traffic app equal ftp direction equal backward`

    `> show log traffic dst in 172.22.22.2 direction equal backward`

    `> show log threat dst in 172.22.22.2 direction equal backward`

    `> show log threat dst in 192.168.1.20 direction equal backward`

    Note: There is no CLI equivalent to the Unified log. To create the Unified log, the web interface runs multiple individual queries and aggregates the output. You can display and select these queries by clicking the **chevron** button to the left of the **magnifying glass**.

4.5 Configure the Capture Filter

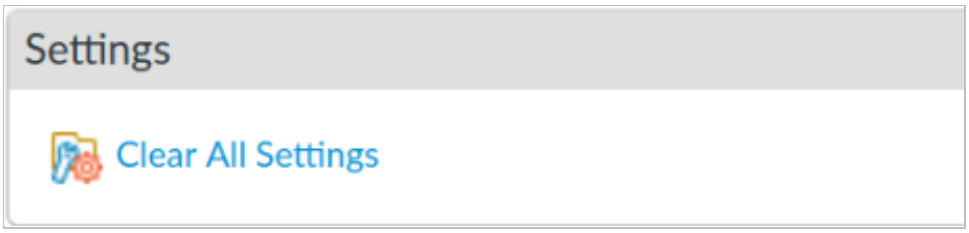
Because there is no log information to help identify the root cause of the problem, your next step is to confirm that traffic from the internal client is arriving at the firewall. The problem may not be with the firewall.

The firewall's system counters track much data that is not logged. There are more than 2,350 individual event counters for the firewall's packet-processing flow, log generation, and other functions. Counter data generally is not useful without extensive filtering and analysis.

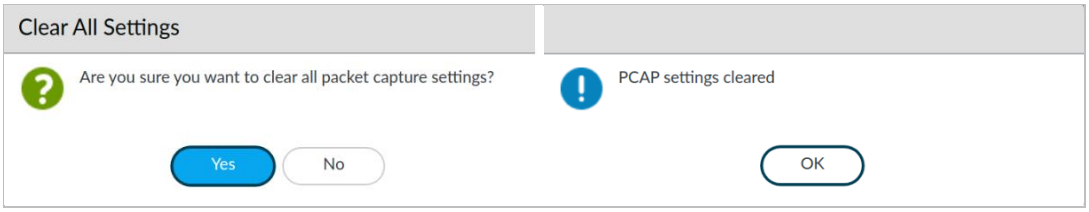
Important: The specific meaning of any individual counter may depend on certain conditions that are proprietary to engineering. You also should assume that a *common term* used in the name of a counter could have a meaning that is unique to a specific set of processes and events that is not intuitive and that is *not* common. Terms such as "error" and "fail," for example, do not mean that anything necessarily is wrong, but simply may mean that a conditional branch of logic could not be applied or did not produce a positive verdict, which in the packet-processing flow is normal.

Fortunately, the counter-data display functions of the CLI provide several ways to filter counter data, which in many cases can help you solve problems. One way to filter counter data is to apply the current *capture filter*.

23. In the firewall web interface, go to **Monitor > Packet Capture**.
24. In the **Settings** section in the bottom center, click **Clear All Settings**:



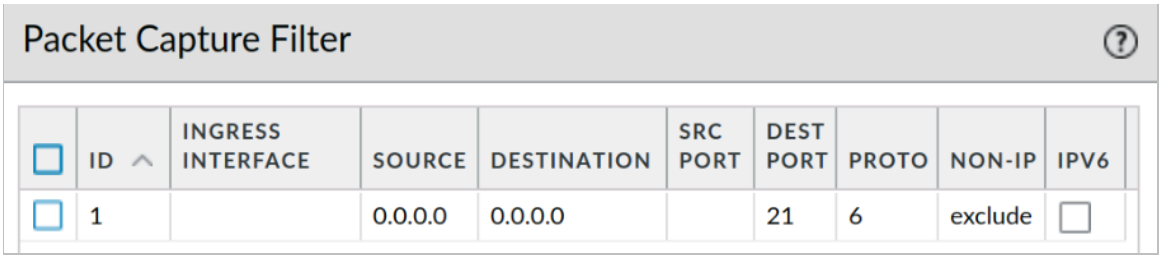
Click **Yes**, and then click **OK**:



Note: Delete any existing pcap files before you proceed.

25. Click **Configure Filtering** > **Manage Filters**.
26. Configure the filter using the following specifications and click **OK**:

Parameter	Value
Id	**1**
Ingress Interface	none [blank]
Source	[blank]
Destination	[blank]
Src Port	[blank]
Dest Port	**21**
Proto	**6**
Non-IP	**exclude**
IPv6	not selected

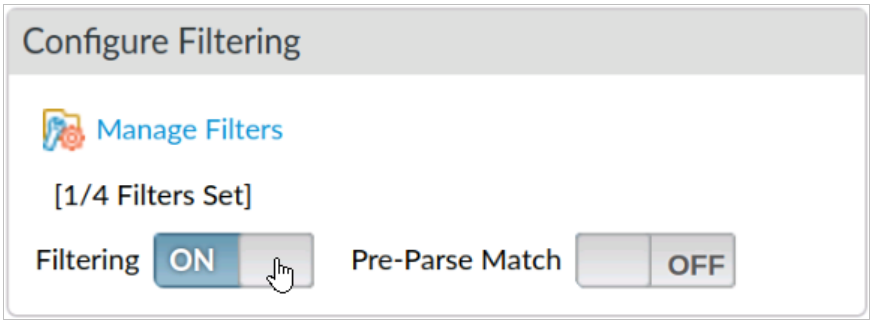


Note: Source and Destination addresses will automatically be populated with 0.0.0.0 when you leave them blank.

You should configure packet capture filters as narrowly as possible and yet as broadly as is required to capture all the data that you need, without the risk that placing such a load on the firewall likely will impede transit data flow.

In this case, you know that TCP traffic on port 21 (FTP) is light. You also have some indication that the root cause of the problem may be related to IP addressing. This filter configuration will capture all FTP traffic, regardless of source or destination.

27. Click to turn filtering **ON**:



4.6 Check Counters

28. Go to the **CLI** of the firewall.

    You can use an existing SSH session (via **Remmina**) if you have one open; otherwise launch **Remmina**.

29. Type:
    `show counter global filter packet-filter yes` and press **Enter**.

    This command will show counters for traffic that matches the current packet filter and various other filter states since the global counters were reset or the firewall was restarted. You typically will want to use the "delta yes" option with this command, as specified in the next step.

```
admin@firewall-a> show counter global filter packet-filter yes [Enter]

Global counters:
Elapsed time since last sampling: 341.510 seconds

name                    value rate severity category aspect    description
----------------------------------------- ---------------
pkt_recv               293050    0 info     packet   pktproc   Packets received
pkt_recv_err                3    0 drop     packet   pktproc   Packet receive error
pkt_sent               365061    0 info     packet   pktproc   Packets transmitted
pkt_sent_err_drop        5584    0 error    packet   pktproc   Packet send error drop
pkt_sent_host            1861    0 info     packet   pktproc   Packets successfully t[...]
session_allocated       12401    0 info     session  resource  Sessions allocated
[. . .]
--------------------------------------------------------------------------------
```

```
Total counters shown: 86
--------------------------------------------------------------------------------
admin@firewall-a>
```

In your lab environment, the counters and values shown will differ from what is shown here.

Tip: Remember to press the **spacebar** to show additional content when the total content to be displayed exceeds the current window size.

30. Type:
    `show counter global filter packet-filter yes delta yes` and press **Enter**.
    If you add the **`delta yes`** option, the firewall will display only counter data that is new since the **`show counter global <options>`** command was last run.

```
admin@firewall-a> show counter global filter packet-filter yes delta yes [Enter]

Global counters:
Elapsed time since last sampling: 129.600 seconds

--------------------------------------------------------------------------------
Total counters shown: 0
--------------------------------------------------------------------------------
```

If the command produces other than zero counters, run the command again.

31. On the client desktop, use the terminal window to repeat your connection attempt to ftp.test.lab:

```
lab-user@client-a:~$ curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

32. The connection will time out eventually, but leave the window open while the ftp connection attempt continues.
33. In the CLI connection to the firewall, press the **Up Arrow** and then press **Enter** to rerun the command:
    `show counter global filter packet-filter yes delta yes`

```
admin@firewall-a> show counter global filter packet-filter yes delta yes [Enter]

Global counters:
://Elapsed time since last sampling: 77.998 seconds

name                value rate severity category  aspect    description
-------------------------------------------------------------
session_allocated      6     0 info     session   resource  Sessions allocated
session_freed          6     0 info     session   resource  Sessions freed
flow_policy_nat_land   6     0 drop     flow      session   Session setup: source
                                                           NAT IP allocation result
                                                           in LAND attack
--------------------------------------------------------------------------------
```

```
Total counters shown: 5
--------------------------------------------------------------------------------
```

Tip: You can also append "**| match flow_policy_nat_land**" to the end of the previous command to make it easier to locate the entry:

show counter global filter packet-filter yes delta yes | match flow_policy_nat_land [Enter]

Tip: If the command does not display any counters, wait 30 seconds, and then rerun the command. If you still do not receive output similar to the example, go to the terminal window and use the up arrow key to attempt the connection again. Then rerun the command in the firewall CLI.

Analysis: The filtered counter results provide you with some interesting information. There appear to have been multiple attempts to connect to the FTP server. The firewall allocates a row in the session tables for each connection, but then frees all of them.

Because no session records were in the Traffic logs, the session allocations likely were freed before the session-setup (slowpath) process was completed. The **flow_policy_nat_land** counter, which has a severity of "drop" explicitly indicates that these packets were dropped in the session-setup stage.

If you have encountered this issue before, you might have enough information at this point to diagnose the problem. If you have not, you will need more information.

If you take a packet capture, you will be able to see the packets exactly as the firewall receives them. If you run a packet-diagnostics trace (flow basic), you will be able to see how the firewall processed those packets.

Packet-capture data and flow-basic data can help you to confirm that the problem is not *downstream* from the firewall (or in between the client and the firewall).

Before you configure the firewall for packet capture and enable packet-diagnostics logging, you should see what a normal connection looks like. Your ability to reference a normal or healthy connection can help prove that the output is relevant to your case.

34. Press the **Up Arrow** and then press **Enter** to rerun the following command:
    `show counter global filter packet-filter yes delta yes`

    You want to rerun the command to zero out the current delta numbers:

    ```
    --------------------------------------------------------------------------------
    Total counters shown: 0
    --------------------------------------------------------------------------------
    ```

    If you do not see "**`0`**" total counters shown after running the command, run it again.

35. Go to the **Remmina CLI** window connected to the Server-Extranet.

    If the session has ended, close the Remmina window, double-click the **Remmina** icon on the desktop, and then double-click the **Server-Extranet** configuration.

36. Type:

`curl -u “paloalto42:Pal0Alt0!” ftp://ftp.test.lab/ftptest.txt”`

37. Press **Enter**.

    Verify that the test text file is printed to the screen.

38. Go back to the **CLI** of the **firewall**.

39. Press the **Up Arrow** and then press **Enter** to rerun the following command:
    `show counter global filter packet-filter yes delta yes`

```
admin@firewall-a> show counter global filter packet-filter yes delta yes

Global counters:
Elapsed time since last sampling: 62.796 seconds

name                         value rate severity category  aspect    description
------------------------------------------------------------------
pkt_recv                      2681   12 info   packet    pktproc   Packets received
pkt_recv_retry                2528   12 info   packet    pktproc   Full Burst Packets R[...]
pkt_sent                        26    0 info   packet    pktproc   Packets transmitted
session_allocated                2    0 info   session   resource  Sessions allocated
session_installed                1    0 info   session   resource  Sessions installed
session_predict_dst              1    0 info   session   resource  Active dst predict s[...]
flow_action_predict              1    0 info   flow      pktproc   Predict sessions created
flow_ip_cksm_sw_validation      29    0 info   flow      pktproc   Packets for which IP[...]
appid_proc                       1    0 info   appid     pktproc   The number of packet[...]
dfa_sw                          19    0 info   dfa       pktproc   The total number of [...]
ctd_sml_exit_detector_i          1    0 info   ctd       pktproc   The number of sessio[...]
ctd_err_bypass                   1    0 info   ctd       pktproc   ctd error bypass
ctd_run_detector_i               1    0 info   ctd       pktproc   run detector_i
ctd_sml_vm_run_impl_opcodeexit 1    0 info   ctd       pktproc   SML VM opcode exit
ctd_fwd_err_tcp_state            1    0 info   ctd       pktproc   Content forward erro[...]
ctd_pscan_sw                    19    0 info   ctd       pktproc   The total usage of s[...]
ctd_appid_reassign               1    0 info   ctd       pktproc   appid was changed
ctd_process                      1    0 info   ctd       pktproc   session processed by ctd
ctd_pkt_slowpath                19    0 info   ctd       pktproc   Packets processed by[...]
tcp_modi_q_pkt_alloc             1    0 info   tcp       pktproc   packets allocated by[...]
tcp_modi_q_pkt_free              1    0 info   tcp       pktproc   packets freed by tcp
modification queue
------------------------------------------------------------------------------
Total counters shown: 21
------------------------------------------------------------------------------
```

40. Compare the results that you receive from the connection from the server to the results from the attempt to connect from the client.

    Analysis: First, simply notice that the two outputs are not similar. In the **`severity`** column, notice that no instance of a **`drop`** appears anywhere in the output for the successful connection. For the failed connection, the count for the **`flow_policy_nat_land`** drop (**6**) exactly equals the individual counts of only two other counters triggered. On the failed connection, *everything* seems to have been dropped. In the successful connection you can see evidence of post-session-setup activity, including App-ID being triggered (**`appid_proc`**) and several CTD events triggered.

Conclusion: The counters that are triggered for the failed connection confirm that the connectivity problem is visible on the firewall. The problem likely can be diagnosed using additional data collected directly from the firewall. The problem may be solved by modifying the firewall configuration, but the firewall solution cannot be determined until you know the root cause. The root cause may not involve the firewall, even if the firewall may be able to compensate for the problem.

Caution: Counter details often are misinterpreted. Your initial goal should be to produce high-level and summary information. Is the severity level "**drop**"? Are *no* counters, some counters, or many counters triggering on the traffic in question? Are sessions being installed and allocated? Which categories of counters are represented?

As noted earlier, the specific meaning of various counters may be proprietary to engineering and certain terms may not mean what they seem to mean. For example, the term "slowpath" in the counter **ctd_pkt_slowpath** is entirely unrelated to the *session-setup (slowpath)* stage for processing the first packet of a session. In the larger, industry-wide scope of network processing, "fastpath" and "slowpath" are generic terms that can be applied to any type of process (from the low-level TCP/IP stack to full-context, deep-packet content inspection) where a branch in the logic results in asymmetric paths as measured in speed, logical rigor, and/or compute cycles.

In the successful connection, 19 packets *did not* go through the session-setup process. The counter data actually indicates that 19 packets went through the "slowpath" logical branch of the CTD process. Other terms that appear in the output include the following:

- **dfa**: Deterministic Finite Automata. Functionally, this is a RegEx-based pattern-matching engine and is used for App-ID.
- **pscan**: A threat scanning function of the firewall. It has no relation to publicly available port scanners or C source-code tools.
- **sml**: State machine language. This term relates to application-layer decoding, especially HTTP stateful inspection.

4.7 Configure Packet Capture and Enable Flow Basic

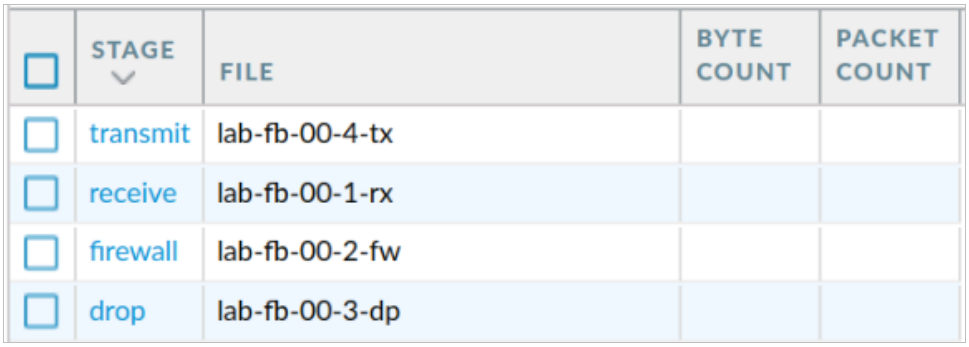
41. Go to **Monitor** > **Packet Capture** in the web interface of the firewall.
42. **Add** the following capture stages and name them as specified.

    Note: If you assign the receive and transmit stage captures separate names for this test, you will more easily be able to determine which packets, if any, are received and which are transmitted.

Stage	File	Packet Count	Byte Count
receive	**lab-fb-00-1-rx**	[blank]	[blank]
firewall	**lab-fb-00-2-fw**	[blank]	[blank]
drop	**lab-fb-00-3-dp**	[blank]	[blank]

Stage	File	Packet Count	Byte Count
transmit	**`lab-fb-00-4-tx`**	[blank]	[blank]

Tip: Use keyboard shortcuts (Ctrl+C and Ctrl+V) to copy and paste "**`lab-fb-00-`**" from one configuration stage to the next. (The letters "**`fb`**" stand for "flow basic.")



	STAGE	FILE	BYTE COUNT	PACKET COUNT
☐	transmit	lab-fb-00-4-tx		
☐	receive	lab-fb-00-1-rx		
☐	firewall	lab-fb-00-2-fw		
☐	drop	lab-fb-00-3-dp		

43. In the Remmina CLI of the **firewall**, type:
`debug dataplane packet-diag show setting` and press **Enter**.

Verify that the packet filter is enabled (**`yes`**) and that you can see the four packet capture stages enabled with filenames. Note that the **`Logging > Features`** line is blank:

```
admin@firewall-a> debug dataplane packet-diag show setting

--------------------------------------------------------------------------------
Packet diagnosis setting:
--------------------------------------------------------------------------------
Packet filter
  Enabled:                    yes
  Match pre-parsed packet:    no
  Filter offload:             yes
  Index 1: 0.0.0.0/0[0]->0.0.0.0/0[21], proto 6
           ingress-interface any, egress-interface any, exclude non-IP
--------------------------------------------------------------------------------
Logging
  Enabled:                    no
  Log-throttle:               no
  Sync-log-by-ticks:          yes
  Features:
  Counters:
  Timeout duration:           60 seconds
  Buffer threshold:           80%
  CPU threshold:              80%
--------------------------------------------------------------------------------
Packet capture
  Enabled:                    no
  Snaplen:                    0
  Username:
  Stage receive           :  file lab-fb-00-1-rx
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage firewall          :  file lab-fb-00-2-fw
```

```
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage transmit          :  file lab-fb-00-4-tx
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage drop              :  file lab-fb-00-3-dp
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
------------------------------------------------------------------------------
```

44. In the command prompt, press:
Up Arrow, then **Alt+Backspace+Backspace** (press **Backspace** twice).

This procedure loads the last command used and deletes the last two options (words).

```
admin@firewall-a> debug dataplane packet-diag
```

45. Type:
`set log feature flow basic` and press **Enter**:

```
admin@firewall-a> debug dataplane packet-diag set log feature flow basic [Enter]

admin@firewall-a>
```

Execution of the command does not produce any output to screen.

46. Press the **Up Arrow** *twice* and then press **Enter** to rerun the following command:
`debug dataplane packet-diag show setting`

```
[. . .]
------------------------------------------------------------------------------
Logging
[. . .]
  Features:
    flow    : basic
  Counters:
[. . .]
```

In the **`Logging`** section, verify that the feature **`flow: basic`** is listed.

47. Press the **Up Arrow**, then **Alt+Backspace** (once), then type:
`filter-marked-session` and press **Enter**.

Tip: Use autocomplete. Type **`filter`**, then press **Tab**:

```
admin@firewall-a> debug dataplane packet-diag show filter-marked-session [Enter]

No Active Marked Sessions
admin@firewall-a>
```

Verify that there are no active marked sessions. Marked sessions are sessions that the firewall has matched to the current packet filter. Because packet-diagnostics is resource-intensive, Palo

Alto Networks technical support recommends that you always verify that filter-marked sessions match your expectations and needs prior to turning on packet-diagnostics logs. In this case, you want to log a single session that has not yet been initiated.

4.8 Run Packet Capture and Flow Basic Diagnostic Logging

48. Execute the following commands:

    debug dataplane packet-diag set capture on

    This command turns on packet capture (visible in the web interface).

    debug dataplane packet-diag set log on

    This command turns on packet-diagnostics logging (not visible in the web interface):

```
admin@firewall-a> debug dataplane packet-diag set capture on [Enter]

Packet capture is enabled
admin@firewall-a> debug dataplane packet-diag set log on [Enter]

Packet log is enabled. WARNING: Enabling of debug commands could result in network
outage. Not recommended if dataplane CPU is above 60%.
```

    Verify that packet capture and packet logging are **enabled**.

    WARNING – enabling debug commands could result in a network outage. Do not enable debug commands if the firewall dataplane CPU usage is above 60%.

49. Rerun the following command:
    debug dataplane packet-diag show setting

    Verify that "Packet filter," "Logging," and "Packet capture" are enabled with "**yes**" tags:

```
[. . .]
Packet filter
  Enabled:                    yes
[. . .]
Logging
  Enabled:                    yes
[. . .]
Packet capture
  Enabled:                    yes
[. . .]
```

    You now are ready to re-create the problem so that the firewall can capture and log the connection attempt.

50. From the client desktop, open a terminal window (or use an existing one if you have not closed it).

curl -u “paloalto42:Pal0Alt0!” ftp://ftp.test.lab/ftptest.txt

```
lab-user@client-a:~$ curl -u “paloalto42:Pal0Alt0!” ftp://ftp.test.lab/ftptest.txt
```

51. Allow the connection attempt to continue (it will eventually time out) and move to the next step.
52. In the CLI of the firewall, rerun the following command:
    debug dataplane packet-diag show setting
    Examine the “Packet capture” section:

```
-------------------------------------------------------------------------------
Packet capture
  Enabled:                   yes
  Snaplen:                   0
  Username:
  Stage receive           :  file lab-fb-00-1-rx
    Captured:     packets - 6          bytes - 388
    Maximum:      packets - 0          bytes - 0
  Stage firewall          :  file lab-fb-00-2-fw
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage transmit          :  file lab-fb-00-4-tx
    Captured:     packets - 0          bytes - 0
    Maximum:      packets - 0          bytes - 0
  Stage drop              :  file lab-fb-00-3-dp
    Captured:     packets - 6          bytes - 388
    Maximum:      packets - 0          bytes - 0
-------------------------------------------------------------------------------
```

    Look for non-zero values across all the capture stages. Does the number of packets captured and dropped match what you saw in the counters?

    You will not always be able to isolate and reproduce on demand a failure condition as well as you have been able to do in this lab. Nevertheless, highly effective troubleshooters always are looking for opportunities to make correlations across endpoint logs, firewall logs, counters, packet captures, protocol details, payload characteristics, and debug logs.

    The finding and tracking of strong correlations (and equivalencies) becomes more important the more complicated the troubleshooting scenario becomes. Confidence in knowing exactly what you are looking for is invaluable. Also invaluable is the degree to which you *know* that “*this* event is directly related to *that* event” or that “this packet *here* is the exact same packet *there.*”
53. Execute the following commands:

    debug dataplane packet-diag set log off

    This command turns off packet-diagnostics logging.

debug dataplane packet-diag set capture off

This command turns off packet capture:

```
[. . .]
Packet log is disabled
[. . .]
Packet capture is disabled
```

Verify that the CLI returns confirmation that packet capture and packet logging are **disabled**.

4.9 Interpret the Flow Basic Log and Pcaps

54. Type:
debug dataplane packet-diag aggregate-logs and press **Enter**.

```
admin@firewall-a> debug dataplane packet-diag aggregate-logs [Enter]

pan_packet_diag.log is aggregated
```

For each processor core that processes a packet from a session that matches to the packet filter, that firewall generates an individual packet-diagnostics log file. Even if you can narrow your packet filter and test case to a single session, the command to aggregate packet-diagnostic logs should be run to produce a predictable file for analysis.

55. Type:
less mp-log pan_packet_diag.log and press **Enter**.

This command displays the aggregated packet-diagnostics log using the **less** program.

You may find data in the **pan_packet_diag.log** file that is outside the scope of the packet-diagnostic features that you have specifically enabled. Such extra data is variable based on the specific version of PAN-OS software that you are using and the configuration of the firewall. To use this file effectively, you must know what you are looking for and how to search for it.

Note: On a firewall that has a separate physical data plane, the pan_packet_diag.log file is in the **dp-log** folder. Thus, you would use the command **less dp-log pan_packet_diag.log**.

56. Type:
/ingress and press **Enter**.
57. Press the **Up Arrow** (or the **k** key) twice to display the timestamp header.
58. Analyze the **ingress stage** record and confirm that the packet profile fits the scenario that you are investigating:

```
== 2022-04-01 00:00:51.061 +0000 ==
Packet received at ingress stage, tag 0, type ORDERED
Packet info: len 66 port 17 interface 17 vsys 1
  wqe index 73538 packet 0x0xc0044e9b80, HA: 0
Packet decoded dump:
L2:     00:50:56:b1:b2:e3->00:50:56:b1:03:cb, type 0x0800
```

```
IP:     192.168.1.20->172.22.22.2, protocol 6
        version 4, ihl 5, tos 0x02, len 52,
        id 31618, frag_off 0x4000, ttl 128, checksum 27387(0xfb6a)
TCP:    sport 41920, dport 21, seq 156440602, ack 0,
        reserved 0, offset 8, window 65535, checksum 9529,
        flags 0xc2 ( SYN), urgent data 0, l4 data len 0
TCP option:
00000000: 02 04 05 b4 01 03 03 08  01 01 04 02                ........ ....
Flow lookup, key word0 0x600020015a3c0 word1 0 key word2 0x1401a8c0ffff0000
* Dos Profile NULL (NO) Index (0/0) *
Session setup: vsys 1
No active flow found, enqueue to create session

[. . . ]
```

The source IP address (**192.168.1.20**) and the destination IP address (**172.22.22.2**) do match the problem scenario. The destination port is **21** (FTP), which also matches the scenario.

The counter data indicated that multiple connection attempts arrived at the firewall, but the counter data does not identify source or destination addresses.

What if the source and destination addresses of the packets received by the firewall were different from what you just saw in the packet-diagnostics log? What if the address pairs were **192.168.1.20** and **192.168.1.20** or **172.22.22.2** and **172.22.22.2**?

Answer: If the firewall had received packets with source IP addresses equal to the destination IP addresses, the firewall likely would not be either the root cause nor capable of remediating the problem. In these cases, all additional efforts to troubleshoot the problem should be directed to devices downstream from the firewall.

59. Analyze the session-setup (**slowpath stage**) record.

    You can use the **Up Arrow** and **Down Arrow** keys or the **j** and **k** keys to scroll one line at a time:

```
[. . . ]
== 2022-04-27 15:15:44.558 +0000 ==
Packet received at slowpath stage, tag 58604962, type ATOMIC
Packet info: len 74 port 17 interface 17 vsys 1
  wqe index 92153 packet 0x0x1041aaac0, HA: 0, IC: 0
Packet decoded dump:
L2:     00:50:56:8f:24:d4->00:50:56:8f:7d:f5, type 0x0800
IP:     192.168.1.20->172.22.22.2, protocol 6
        version 4, ihl 5, tos 0x00, len 60,
        id 12589, frag_off 0x4000, ttl 64, checksum 47749(0x85ba)
TCP:    sport 57286, dport 21, seq 3053787596, ack 0,
        reserved 0, offset 10, window 64240, checksum 12117,
        flags 0x02 ( SYN), urgent data 0, l4 data len 0
TCP option:
00000000: 02 04 05 b4 04 02 08 0a  3b d4 ae 63 00 00 00 00    ........ ;..c....
00000010: 01 03 03 07                                         ....
Session setup: vsys 1
PBF lookup (vsys 1) with application none
Session setup: ingress interface ethernet1/2 egress interface ethernet1/3 (zone 3)
2022-04-27 15:15:44.558 +0000 debug: pan_policy_lookup(pan_policy.c:2402): [ACE] Tr[. . .]
```

```
2022-04-27 15:15:44.558 +0000 debug: pan_policy_match_service(pan_policy.c:1613): m[. . .]
2022-04-27 15:15:44.558 +0000 debug: pan_policy_lookup(pan_policy.c:2402): [ACE] Tr[. . .]
2022-04-27 15:15:44.558 +0000 debug: pan_policy_match_service(pan_policy.c:1613): m[. . .]
NAT policy lookup, matched rule index 1
Destination NAT, translated IP 192.168.1.20
PBF lookup (vsys 1) with application none
Session setup: egress zone 2 for natted IP
Translated IP in zone 2, egress id 17
2022-04-27 15:15:44.558 +0000 debug: pan_policy_lookup(pan_policy.c:2402): [ACE] Trigger
slow match for appid(0) uappid(0)
2022-04-27 15:15:44.558 +0000 debug: pan_policy_match_service(pan_policy.c:1613): match
4,0 for app 0 uapp 0 proto 6 sport 57286 dport 21
Policy lookup, matched rule index 4,
TCI_INSPECT: Do TCI lookup policy - appid 0
Allocated new session 67004.
set exclude_video in session 67004 0xe0564b1180 0 from work 0xe039a38c00 0
no swg configured
Packet dropped, vsys 1 NAT rule index 32770 result in LAND attack, same SA/DA 192.168.1.20
Packet dropped, Session setup failed
[. . .]
```

60. Focus on the following lines from the preceding output:

```
Destination NAT, translated IP 192.168.1.20 [. . .]
Allocated new session 67004. [. . .]
Packet dropped, vsys 1 NAT rule index 32770 result in LAND attack, same SA/DA 192.168.1.20
Packet dropped, Session setup failed
```

61. Formulate answers to the following questions:

    Is destination NAT *applied* at this point in the flow logic? If not, where is it applied?

    Answer: No. The effects of destination NAT, including IP addressing and final egress zone, are *analyzed* in the session-setup stage, but address transformation is not applied to the packet itself until the security-processing (fastpath) stage.

    Does the **`Session setup failed`** event explain why there are no **Traffic** logs for this connectivity problem?

    Answer: Yes. A **Traffic** log is in part a product of a session log written to disk. If a connection attempt fails during the session setup stage, the firewall will not generate a **Traffic** log.

62. (Optional) Download and open the receive-stage and drop-stage pcaps:

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.20	172.22.22.2	TCP	66	41920 → 21 [SYN, ECN, CWR] Seq=156440602 Win=655…
2	0.987228	192.168.1.20	172.22.22.2	TCP	66	[TCP Retransmission] 41920 → 21 [SYN, ECN, CWR] …
3	2.997211	192.168.1.20	172.22.22.2	TCP	62	[TCP Retransmission] 41920 → 21 [SYN] Seq=156440…
4	7.016517	192.168.1.20	172.22.22.2	TCP	66	41923 → 21 [SYN, ECN, CWR] Seq=3493276997 Win=65…
5	8.006518	192.168.1.20	172.22.22.2	TCP	66	[TCP Retransmission] 41923 → 21 [SYN, ECN, CWR] …
6	10.026536	192.168.1.20	172.22.22.2	TCP	62	[TCP Retransmission] 41923 → 21 [SYN] Seq=349327…

Formulate answers to the following questions:

- Do the packets in the drop-stage pcap show pre-NAT or post-NAT addressing?
- Is there any difference between the packets that appear in the receive stage and the drop stage?
- How do you account for the number of packets?

63. Explain the root cause of the problem and viable solutions.

Problem definition: Typically, internal clients will resolve the FQDNs of internally hosted resources, such as the FTP server, directly to an internal IP address. This internal name resolution occurs because internal clients typically should be getting DNS resolutions from internal DNS servers.

In this case, the client is resolving to the *external* IP address of the FTP server (172.22.22.2). This external address is configured by NAT policy to translate the destination to the same IP address as the client (192.168.1.20). This source-equals-destination translation happens only because the client is the host of the FTP service to which it is attempting to connect.

At first, this scenario may seem unique and too narrow to ever encounter in the real world. But, consider a situation in which a proxy server, or some server that effectively applies NAT for multiple clients, simultaneously is the host of one or more externally DNS-resolved applications. The server would not have to be the actual host, but perhaps just a front-end server that applies another layer of destination NAT to facilitate access to the actual host(s).

Solutions: Regardless of the scale of the problem, the range of viable solutions is about the same. To address the root cause, you can fix the problem of internal vs. external name-resolution by enforcing via endpoint and network policy the use of internal DNS servers for all internal clients (and properly configuring those DNS servers). To mitigate the issue and to provide a backup solution even if you also do fix the root cause, you can configure the NAT policy of the firewall to implement what is commonly called a “U-turn” or “hairpin” NAT rule.

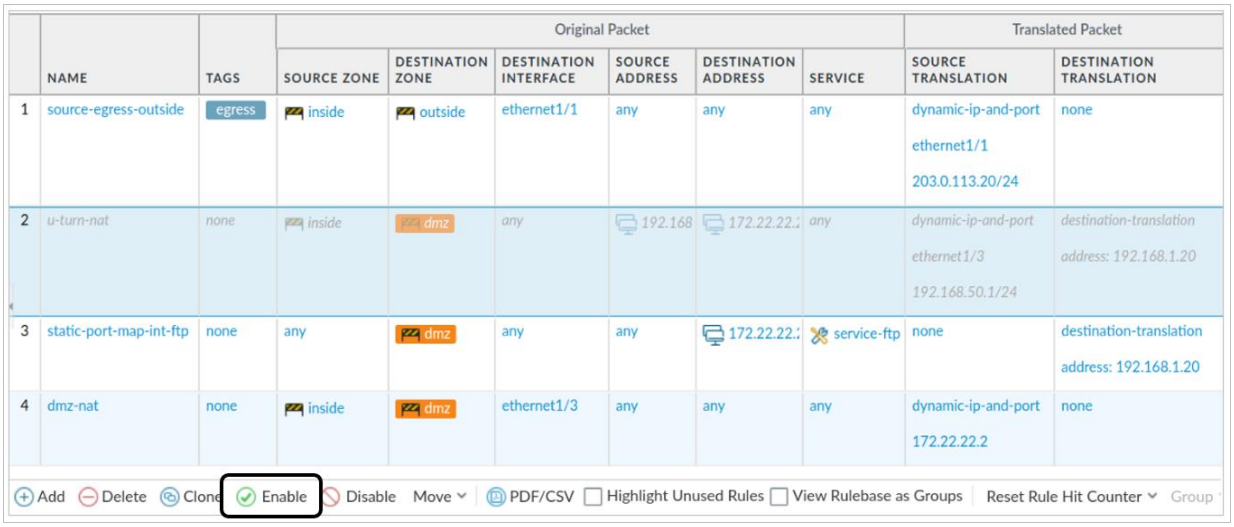
A U-turn NAT rule would apply only to packets from the internal client(s) (including 192.168.1.20) addressed to the external NAT address of the resource (172.22.22.2). The U-turn NAT rule then would apply *source NAT* translation, so that the source address of the internal client no longer is internal (in this case, the source IP address would become 192.168.50.1). Then, almost simultaneously the rule would apply *destination NAT* translation that converts the destination address back to the address of the internal source (in this case, 192.168.1.20).

On ingress, the firewall’s session-setup process will see internally initiated connection attempts as originating from 192.168.50.1 and destined for 192.168.1.20, thus passing the LAND attack analysis that is currently blocking the problem connection(s).

4.10 Implement a Solution and Verify

64. In the web interface of the firewall, go to **Policies** > **NAT**.
65. Select the **u-turn-nat** rule and click **Enable**:

    Note: This rule has been pre-configured to help illustrate the solution.



66. (Optional) Open the **u-turn-nat** rule and review the configuration.

    How does this rule work to solve the problem?

    Answer: It translates the source address of internal packets destined for 172.22.22.2 to an address (192.168.50.1) that is not 192.168.1.20, the translated destination address.

67. **Commit** the configuration.
68. Note the hit count (column) for the **u-turn-nat** rule.
69. Use the open terminal window on the client to rerun the curl command:

```
curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

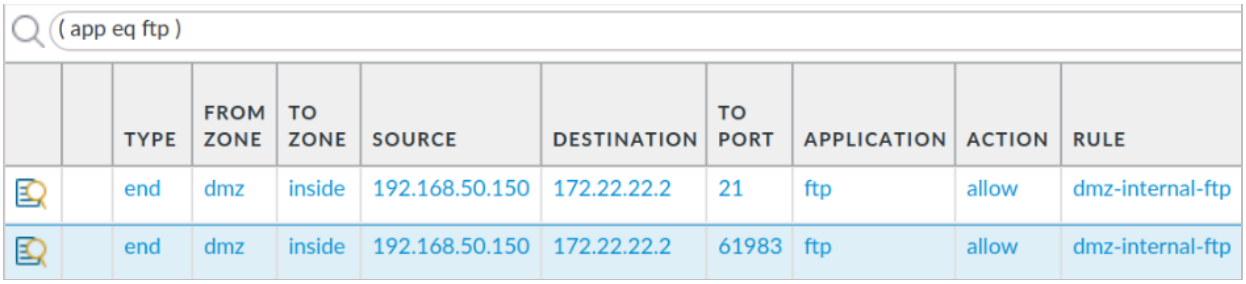
```
lab-user@client-a:~$ curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
=================
You have connected to and retrieved a file
From the internal ftp.test.lab server.
=================
This is a test file for
Validating basic ftp connectivity.
=================
END OF FILE
=================
lab-user@client-a:~$
```

70. In the web interface of the firewall, **refresh** the **Policies** > **NAT** page.

71. Verify that that **Hit Counter** has incremented by one (1).
72. (Optional) Check connectivity from the DMZ server to confirm that the new policy does not break external connectivity.

4.11 Check Logs and Enable Logging for Increased Visibility

73. Go to **Monitor** > **Logs** > **Traffic**.
74. To the right of the filter box, click **X** (**Clear Filter**) and in the filter box type: **`(app eq ftp)`** and press **Enter** or click → (**Apply Filter**):



		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE
		end	dmz	inside	192.168.50.150	172.22.22.2	21	ftp	allow	dmz-internal-ftp
		end	dmz	inside	192.168.50.150	172.22.22.2	61983	ftp	allow	dmz-internal-ftp

Notice that the only **Source** addresses listed in the logs still are **192.168.50.150**.

Is it reasonable to expect the firewall to log connections from the internal client?

You might be able to answer this question with information found in the Session Browser.

75. Use the open terminal window on the client to rerun the curl command:

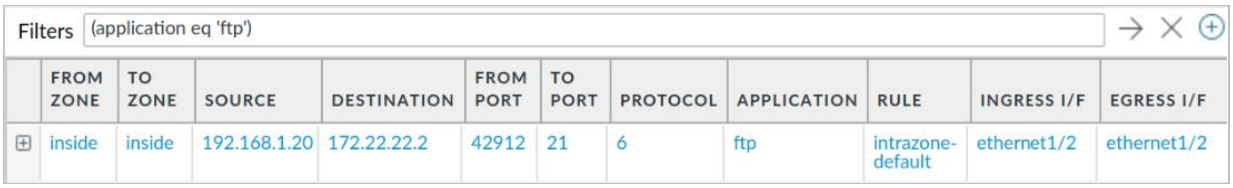
```
curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

Leave the terminal window open.

76. In the web interface of the **firewall**, go to **Monitor** > **Session Browser**.
77. In the filter box, type: **`(application eq 'ftp')`** and press **Enter** or click → (**Apply Filter**).

The summary line for open FTP session should be visible:

Note: This step must be done rather quickly before the session ends and moves to Traffic log.

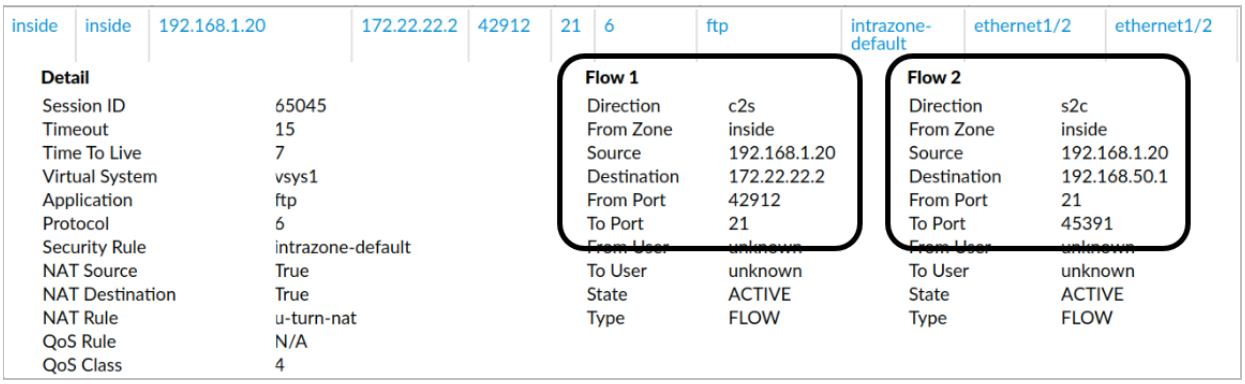


	FROM ZONE	TO ZONE	SOURCE	DESTINATION	FROM PORT	TO PORT	PROTOCOL	APPLICATION	RULE	INGRESS I/F	EGRESS I/F
⊞	inside	inside	192.168.1.20	172.22.22.2	42912	21	6	ftp	intrazone-default	ethernet1/2	ethernet1/2

Analyze the summary session information. Why are these sessions not logged?

- Note the **From Zone** and **To Zone** values. Both zones are "**inside**."
- Note the **Source** and **Destination** addresses.
- Is the address **172.22.22.2** part of the **inside** zone? (Cross-reference the **Destination** zone of the applicable NAT policies.)
- Note the **Rule** that is matched: **intrazone-default**.

78. Click ⊞ to open the session **Detail** view:



Analyze the session details. Does this additional information help provide a more complete answer for why these sessions are not logged?

- Note that the source translation to **192.168.50.1** appears as the **Destination** of the **Flow 2, s2c** return traffic. But, the first packet that will be transmitted to the server will have **192.168.50.1** as the *source* and **192.168.1.20** as the destination. The categorization of the addressing should tell you something about the way that the firewall defines flows.

  The firewall defines flows relative to the *ingress* of the traffic, that is, as the firewall *receives* the flow from the client and as the firewall *receives* the flow from the server.
- Note that the values for **From Zone** in the details for **Flow 1** and **Flow 2** both are "**inside**." How does the firewall define the **To Zone** for a *session*?

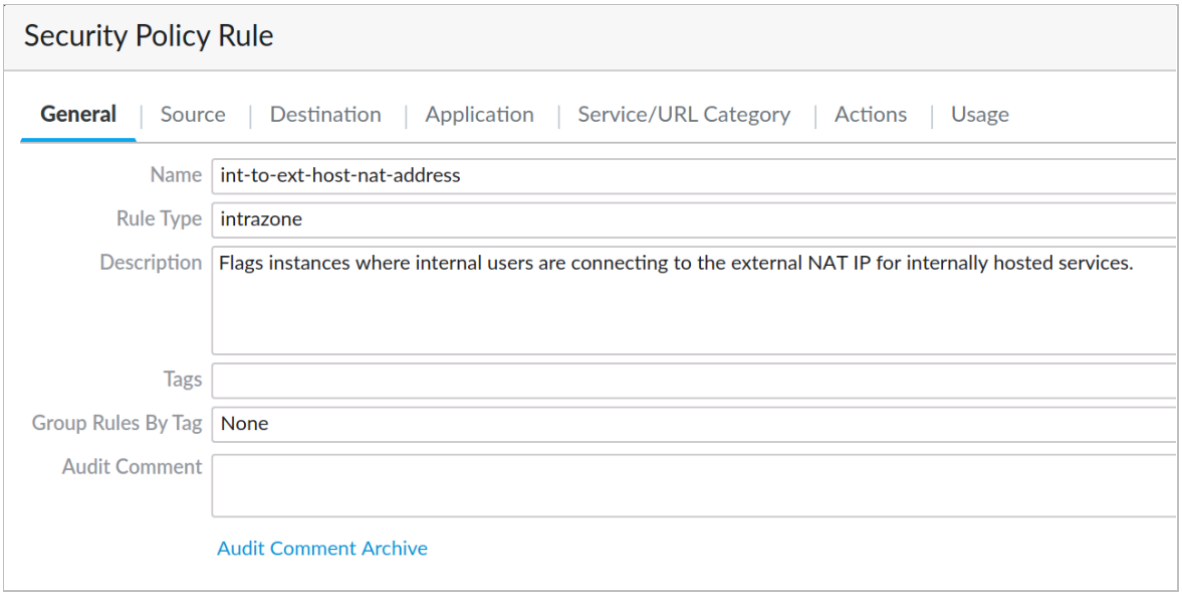
  Answer: By the value of the **From Zone** for the s2c flow.
- Is a destination zone listed for each flow? Is this different than for **NAT Policy** rules?

The firewall does not log FTP sessions from 192.168.1.20 to 172.22.22.2 because the firewall sees these sessions as *intrazone* traffic and matches them to the **intrazone-default** rule, which by default is *not* configured to log session data.

In the following steps, you will examine and enable a pre-configured Security policy rule that will log FTP sessions that originate from the internal client. Suppose that your organization also addresses this connectivity problem by tightening the DNS policies for endpoints, servers, and network devices. You can use Security policy rules not only to control traffic, but also to log it, associate it with a meaningful rule name, and monitor it.

79. In the web interface for the firewall, go to **Policies > Security**.

80. Click the rule name **int-to-ext-host-nat-address**:

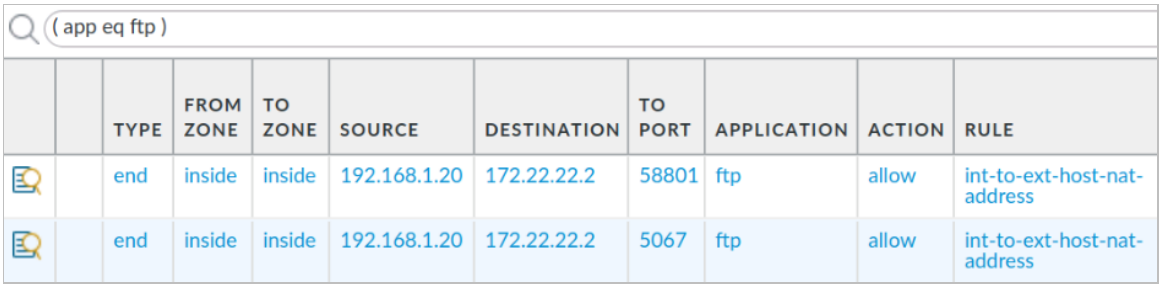


81. Click through the **Source**, **Destination**, **Application**, and **Actions** tabs. If you have questions about the configuration of this rule, ask your instructor or work out answers with a lab partner.
82. Click **OK**.
83. With the rule selected, click **Enable** at the bottom of the display area.
84. **Commit** the configuration.
85. Use the open terminal window on the client to rerun the curl command:

```
curl -u "paloalto42:Pal0Alt0!" ftp://ftp.test.lab/ftptest.txt
```

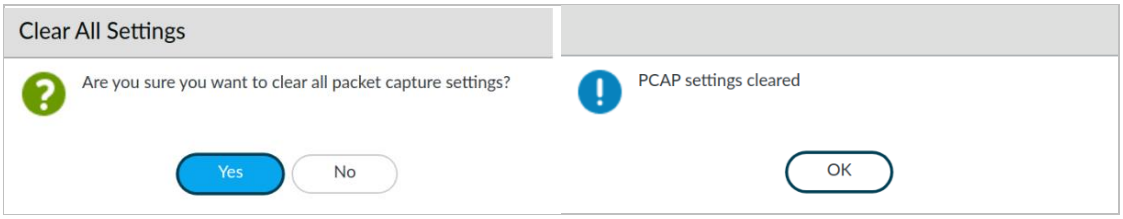
Verify that your text displayed.

86. Wait 30 to 60 seconds, then go to **Monitor** > **Logs** > **Traffic** and verify that the firewall is logging connections from **192.168.1.20**:



4.12 Clean Up Your Lab Environment

87. Go to the **Monitor** > **Packet Capture** page.
88. In the **Settings** section in the bottom center, click **Clear All Settings**.
89. When you are prompted, click **Yes**:



90. After you are notified that the pcap settings have been cleared, click **OK**.

    Note: Delete any existing pcap files.

91. In the CLI of the firewall, press **q** to exit the **less** program.
92. Execute the following command:
    debug dataplane packet-diag show setting

```
admin@firewall-a> debug dataplane packet-diag show setting

--------------------------------------------------------------------------------
Packet diagnosis setting:
--------------------------------------------------------------------------------
Packet filter
  Enabled:                  yes
  Match pre-parsed packet:  no
  Filter offload:           yes
--------------------------------------------------------------------------------
Logging
  Enabled:                  no
  Log-throttle:             no
  Sync-log-by-ticks:        yes
  Features:
  Counters:
  Timeout duration:         60 seconds
  Buffer threshold:         80%
  CPU threshold:            80%
--------------------------------------------------------------------------------
Packet capture
  Enabled:                  no
  Snaplen:                  0
  Username:
--------------------------------------------------------------------------------
```

    Verify that all settings have been cleared:

    - Confirm that **Logging > Enabled** reads **no**.
    - Confirm that no **Features** are listed.

93. Press **Ctrl+L** to clear the screen of the CLI of the firewall.

Leave the connection open for the next lab.

94. Close all open windows related to the following programs:
    - Remmina connection to the Server-Extranet
    - Terminal
    - Wireshark

Stop. This is the end of the lab.

5.1 Lab: Host-Inbound VPN Traffic—Case A

Lab Objectives

- Use the web interface data to check VPN health
- Use log data to identify and diagnose phase-2 tunnel-establishment issues
- Use the information collected to fix the problem

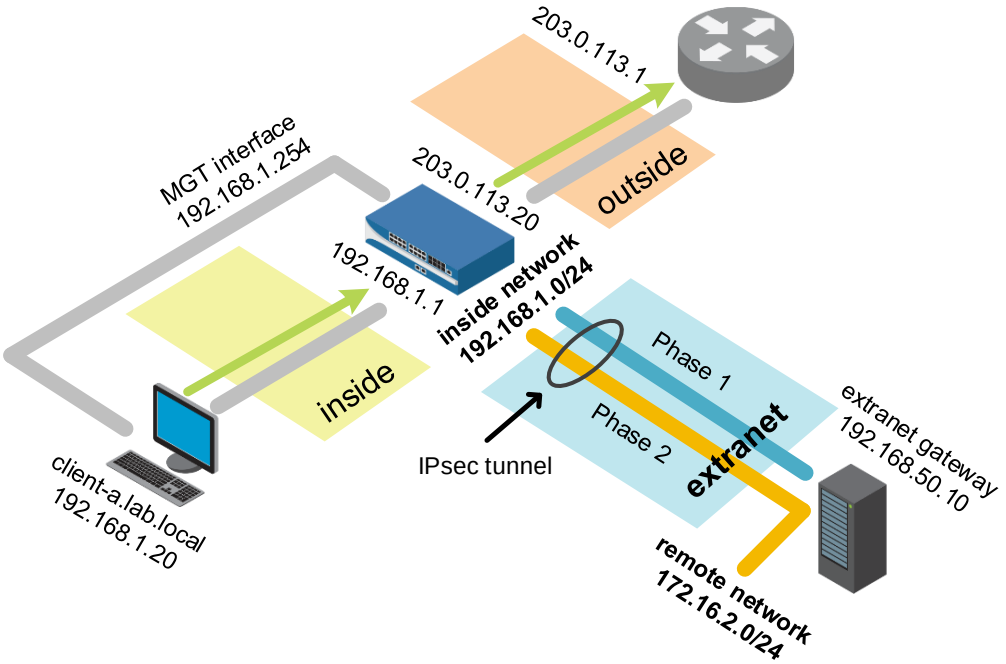
Lab Scenario

Users have created helpdesk tickets because they cannot access a mission-critical web-based application. Under normal conditions, traffic from internal users to the website is routed through a VPN tunnel to your main data center. Last night, because of a maintenance issue, other members of your organization executed a change order to tunnel this same traffic to a secondary data center.

You need to:

- Find the problem: Is it in the routing configuration, the VPN connection, the network connection at the remote site, the host server, or perhaps the target web-application?
- Diagnose the problem and identify a solution.
- Fix the problem and validate the results of your actions.

The lab configuration that you will use is based on the following network topology:



5.1.1 Apply a Baseline Configuration to the Firewall

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-05a.xml**

   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. After the load task is complete, use the web interface to **Commit** the configuration.

   Verify the **Result** reported is "Successful" and the **Details** include "Configuration committed successfully."

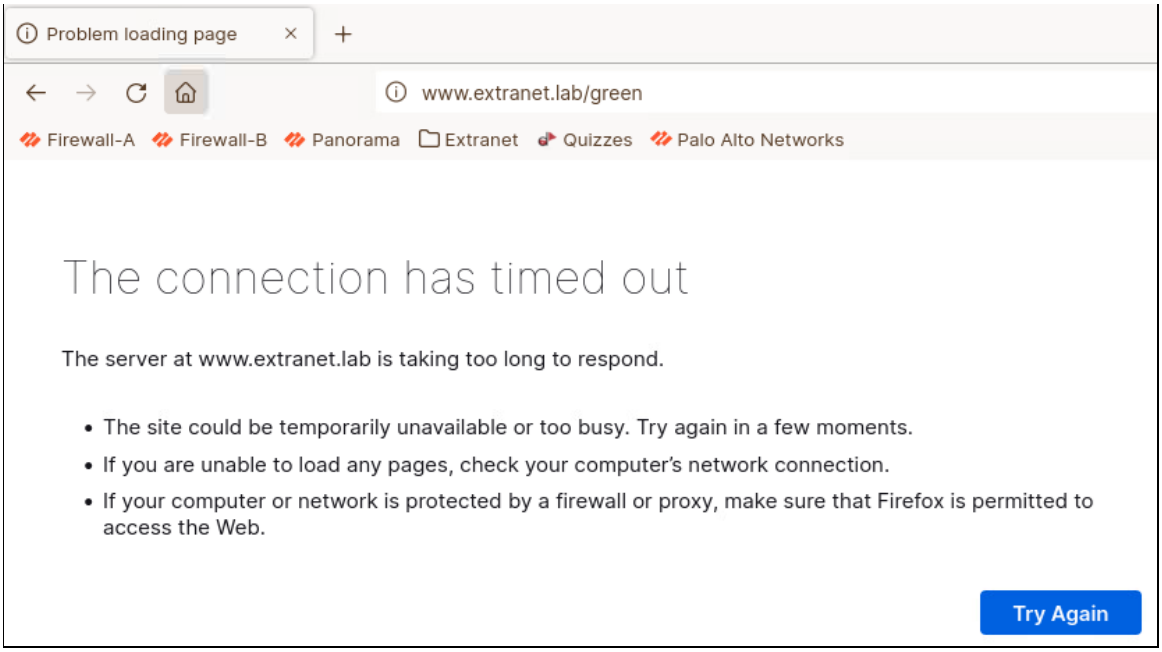
3. Click **Close**.

5.1.2 Verify the Problem

4. Open the testing browser and go to: **`http://www.extranet.lab/green`**

   Important: Make sure you use "http://" for the resource specification.

   Verify the connection attempt fails.



   Leave the testing browser window open.

5. Open a Terminal on the student desktop and type: **`ping www.extranet.lab`** and press **Enter**.

   If the destination website is down, perhaps the server is up. A ping to the hostname also will check DNS resolution.

```
lab-user@client-a:~/Downloads$ ping www.extranet.lab
PING www.extranet.lab (172.16.2.10) 56(84) bytes of data.
^C
--- www.extranet.lab ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7148ms

lab-user@client-a:~/Downloads$
```

Verify that the hostname resolves to the IP address 172.16.2.10.

Press Ctrl+C to stop the Ping after several seconds.

Note: The host will not respond to ping at this point.

5.1.3 Check Routing and Security Policy Rules

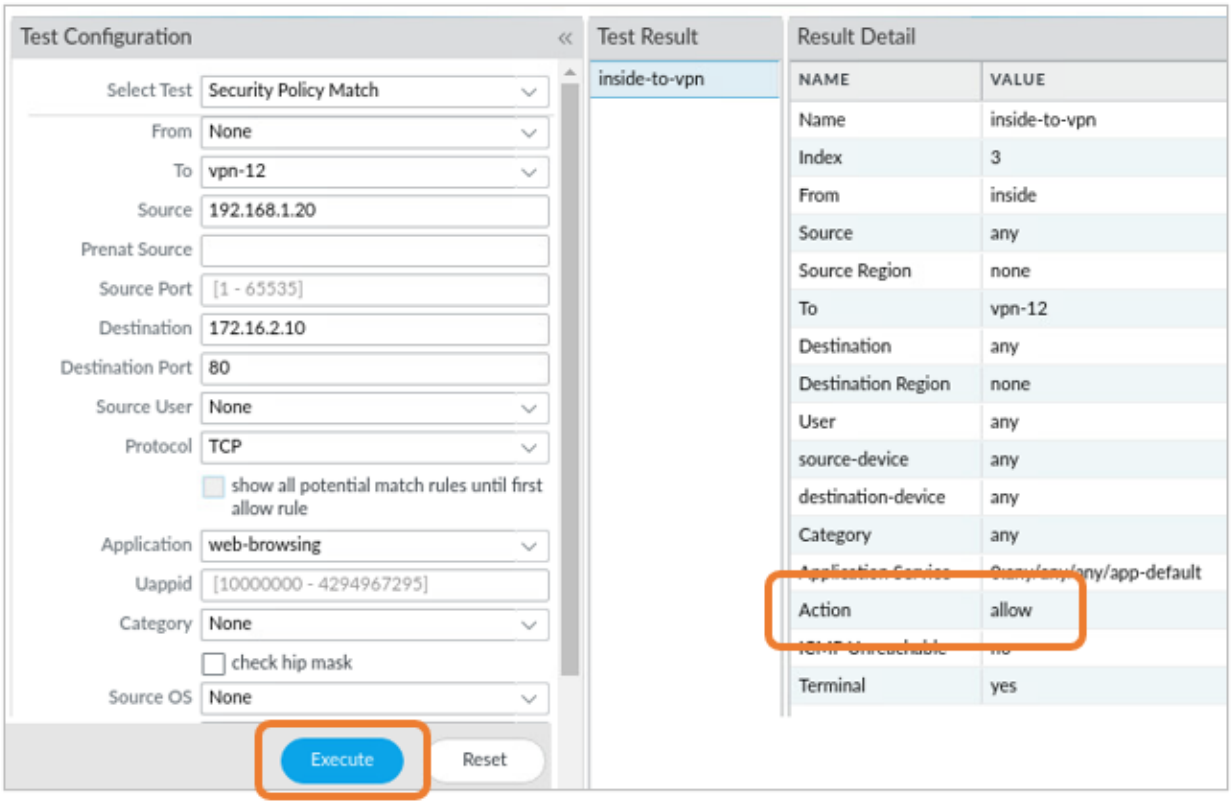
6. At the Terminal, type:
   `ping 192.168.1.1` and press **Enter**.

   Press Ctrl+C to stop the Ping after several seconds.

   A ping to the internal gateway will test the next hop in the route path. Verify the ping succeeds.
7. In the web interface of the firewall, go to **Device** > **Troubleshooting**.
8. On the **Device** > **Troubleshooting** page, configure a **Security Policy Match** lookup using the following settings:

Parameter	Value
Select Test	**Security Policy Match**
From	**None**
To	**vpn-12**
Source	**`192.168.1.20`**
Destination	**`172.16.2.10`**
Destination Port	**`80`**
Source User	**None**
Protocol	**TCP**
show all potential match rules until first allow rule	not selected
Application	**web-browsing**
Category	**None**
check hip mask	not selected

9. Click **Execute** and then click the test result, **inside-to-vpn**, in the middle column.



Verify that the **Result Detail** includes the **Action** "allow." This result indicates that there probably is not a Security policy issue.

10. (Optional) Your organization's documentation indicates that NAT should not be enabled on this interface. Execute a NAT Policy Match lookup using the following settings:

Parameter	Value
Select Test	**NAT Policy Match**
From	**`inside`**
To	**`vpn-12`**
Source	**`192.168.1.20`**
Destination	**`172.16.2.10`**
Source Port	[blank]
Destination Port	**`80`**
Protocol	**`TCP`**
To Interface	None
Ha Device ID	[blank]

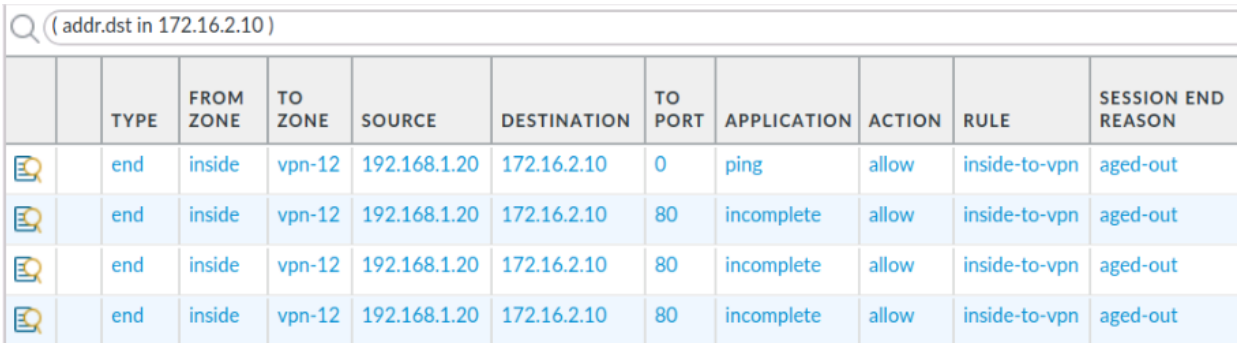
Verify that the Test Result is “No Rule Matched.”

5.1.4 Stop! Try a Top-Down Approach Instead

In the prior several steps, you used some important tools to move “bottom-up” through the basic layers of network and firewall functionality. Nothing is wrong with this approach. Use of these tools may be the only viable options available to you, for example, in the absence of much initial traffic during the deployment of new rules and other configurations.

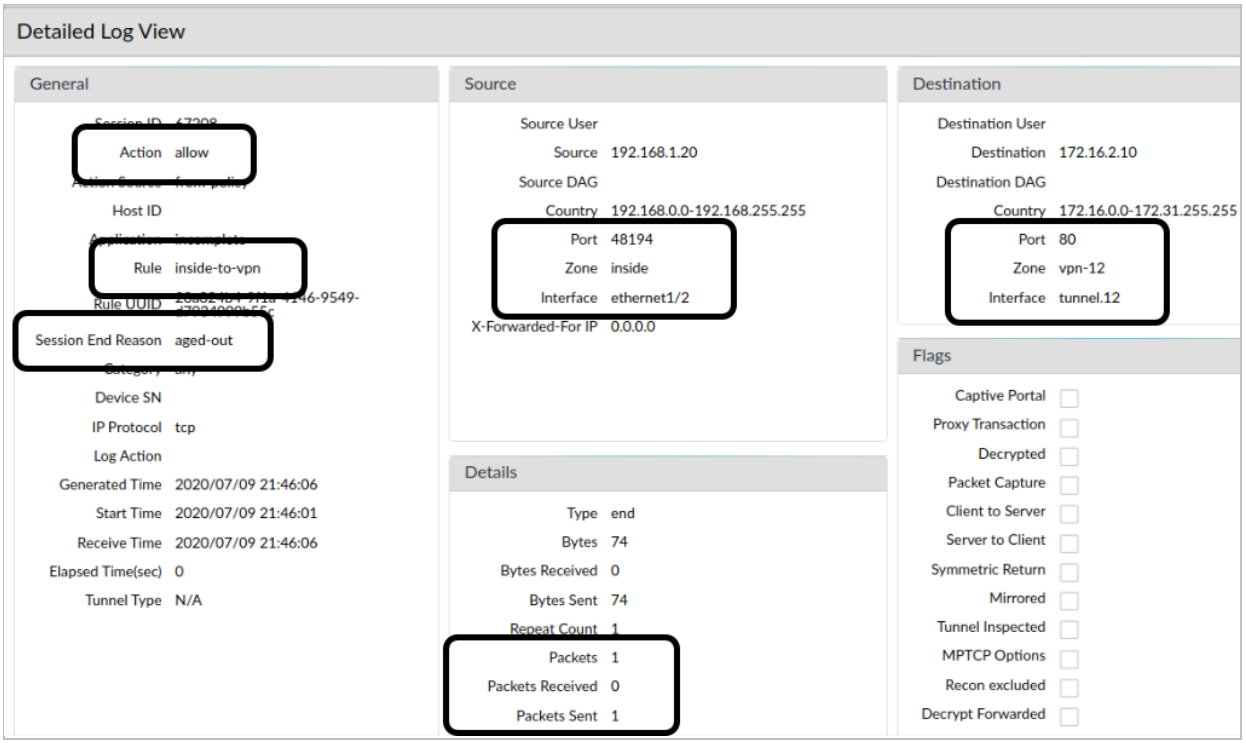
However, a faster option often is to begin troubleshooting with a tool such as the Traffic logs. Traffic logs that are related to the problem, if they exist, will provide an instant view of multiple layers of network and firewall functions in the results of a single query.

11. In the web interface of the firewall, go to **Monitor > Logs > Traffic**.
12. Clear any existing filters.
13. In the table of data, click any **Destination** address.
14. Edit the text in the filter box to match the following filter statement:
    `( addr.dst in 172.16.2.10 )`
15. Press **Enter** or click → (**Apply Filter**):



		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		end	inside	vpn-12	192.168.1.20	172.16.2.10	0	ping	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.10	80	incomplete	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.10	80	incomplete	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.10	80	incomplete	allow	inside-to-vpn	aged-out

16. Click the **Detailed Log View** icon in the left column for a **Traffic** log record that corresponds to a connection attempt to port 80:



17. Analyze the information that the **Detailed Log View** provides and compare it to the information that you derived from the multiple steps that you performed previously.

    The ping to the destination server has produced the only information, in the form of DNS resolution, that you would not have been able to get by first examining the Traffic logs.

    Note: If a NAT rule were to have been applied, the Traffic logs would display the NAT IP addresses and port numbers in the **Source** and **Destination** sections.

18. Click **Close** to close the **Detailed Log View** window.
19. (Optional) Go to the **Policies** > **Security** page and review the existing policy rule names, the naming of the zones, and the configuration of the individual rules.

    Verify that the hit count for the "inside-to-vpn" rule is not zero, which in the lab environment will reflect similar activity that you discovered in the Traffic logs. For the purposes of the lab, the rules provided are open, are non-blocking, and have no Security profiles applied.

	NAME	TAGS	Source ZONE	Destination ZONE	APPLICATION	SERVICE	ACTION	PROFILE	OPTIONS	HIT COUNT
1	inside-to-extranet	none	inside	extranet	any	application-default	Allow	none		2836
2	extranet-to-inside	none	extranet	inside	any	application-default	Allow	none		0
3	inside-to-vpn	none	inside	vpn-12	any	application-default	Allow	none		31
4	vpn-to-inside	none	vpn-12	inside	any	application-default	Allow	none		0
5	dns-no-log	no-log	inside	outside	dns	application-default	Allow	none	none	508
6	inside-to-internet	none	inside	outside	any	application-default	Allow	none		401
7	extranet-to-outside	none	extranet	outside	any	application-default	Allow	none		8
8	danger-simulated-traffic	none	dange...	dang...	any	application-default	Allow			11980
9	intrazone-default	none	any	(intrazone)	any	any	Allow	none	none	69296
10	interzone-default	none	any	any	any	any	Deny	none		-

5.1.5 Check the Health of the VPN Tunnel

The Traffic logs indicate that the firewall has correctly received and routed the problem connections to the correct tunnel interface. The next step is to check the health of the tunnel.

20. In the web interface of the firewall, go to **Network > IPSec Tunnels**.

    Scan the high-level red-light–green-light indicators to check the status of the tunnel.

NAME	STATUS	IKE Gateway/Satellite				Tunnel Interface			
		INTERFACE	LOCAL IP	PEER ADDRESS	STATUS	INTERFACE	VIRTUAL ROUTER	SECURITY ZONE	STATUS
extranet-tunnel	Tunnel Info	ethernet1/3	192.168.50.1/24	192.168.50.10	IKE Info	tunnel.12	lab-vr (Show Routes)	vpn-12	

    Notice that the **IKE Gateway/Satellite** status is green, but the tunnel status is red. This information indicates that the control channel of the tunnel is established, but the functions of the tunnel related to the encapsulation and encryption of data is not operational.

21. Go to **Monitor > Logs > System**.

    The System logs may provide you with information related to phase one and phase two of the establishment of the tunnel. You will expect to see information that confirms that phase one has been completed successfully, as indicated by the green status of the gateway. You hope to find information that indicates why the tunnel is not operational.

22. In the filter box, type:
`(subtype eq vpn)` and press **Enter** or click → (**Apply Filter**):

TYPE	SEVERITY	EVENT	DESCRIPTION
vpn	informational	ike-nego-p2-fail	IKE phase-2 negotiation is failed as initiator, quick mode. Failed SA: 192.168.50.1[500]-192.168.50.10[500] message id:0x956F8AF4. Due to negotiation timeout.
vpn	informational	ike-recv-notify	IKE protocol notification message received: INVALID-ID-INFORMATION (18).
vpn	informational	ike-recv-notify	IKE protocol notification message received: INVALID-HASH-INFORMATION (23).
vpn	informational	ike-recv-notify	IKE protocol notification message received: INVALID-HASH-INFORMATION (23).
vpn	informational	ike-recv-notify	IKE protocol notification message received: INVALID-ID-INFORMATION (18).
vpn	informational	ike-nego-p2-start	IKE phase-2 negotiation is started as initiator, quick mode. Initiated SA: 192.168.50.1[500]-192.168.50.10[500] message id:0x956F8AF4.

Note: The output you see in the lab will contain more entries than are shown in the example. You may need to add a filter to include "eventid eq ike-recv-notify" to display the messages for comparison.

The log data displayed not only confirms that there is a problem with the tunnel, but also indicates that the firewall is the initiator of the VPN connection (ike-nego-p2-fail). The ike-recv-notify messages do not seem to provide an instant identification of the root cause of the issue.

23. Click to open a *new tab* in the testing browser, go to the LIVE Community website (**`live.paloaltonetworks.com`**), and use the LIVE Community search box to search for the following string:
`VPN "invalid-id-information"`

You may quickly be able to find information that indicates that "The most common phase-2 failure is due to Proxy ID mismatch."

At this point, you could attempt to inspect the proxy ID configuration, diagnose the problem as a proxy ID mismatch, and apply a remedy.

You now may review the proxy ID configuration, but to complete the lab activity as designed, do *not* attempt to resolve the problem.

5.1.6 Initiate the VPN Connection from the Remote Network

The following steps will provide you with an opportunity to see the same problem with the firewall as the *responder* to the VPN configuration.

VPN log information generally can be more verbose when you are troubleshooting the responder. The responder can compare information offered by the initiator with *local* information that the responder will not provide to initiators. When certain parameters shared by the initiator fail to match the requirements of the responder, the responder stops communicating.

24. On the student desktop, double-click the **Remmina** shortcut.
25. Double-click the **Server-Extranet** configuration in the **Remmina Remote Desktop Client** window.

    This configuration will connect you to a DMZ server. The server runs VPN software that makes IPsec-tunnel connections to and accepts IPsec-tunnel connections from the firewall.
26. Type:
    `sudo ipsec setup restart` and press **Enter**.

    This command resets the IPsec service on the server. After the service restarts, the server will attempt to *initiate* a VPN connection to the firewall.

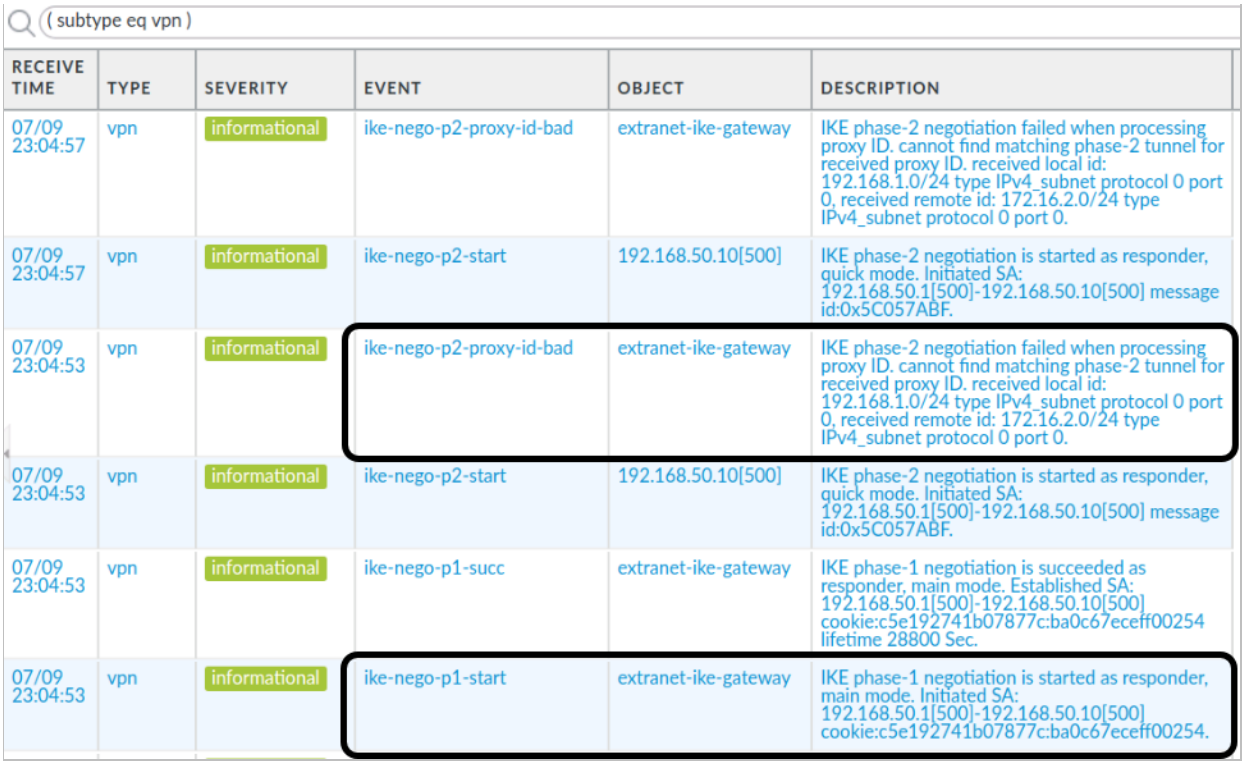
```
paloalto42@extranet1:~$ sudo ipsec setup restart
Stopping strongSwan IPsec…
Starting strongSwan 5.6.2 IPsec [starter]…
paloalto42@extranet1:~$
```

27. Type:
    `exit` and press **Enter**.

    This command will end the SSH session and the Remmina window will close.

5.1.7 Troubleshoot the VPN Connection as the Responder

28. Go to the web interface of the firewall and refresh the **Monitor > Logs > System** page.
29. Find the sequence of log events that begins with **ike-nego-p1-start** and ends with **ike-nego-p2-proxy-id-bad**:



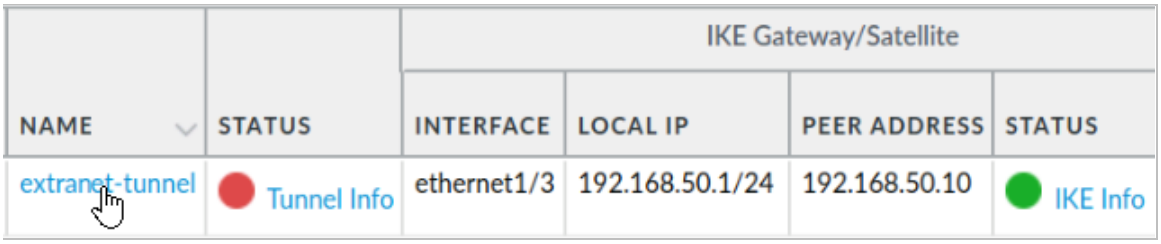
RECEIVE TIME	TYPE	SEVERITY	EVENT	OBJECT	DESCRIPTION
07/09 23:04:57	vpn	informational	ike-nego-p2-proxy-id-bad	extranet-ike-gateway	IKE phase-2 negotiation failed when processing proxy ID. cannot find matching phase-2 tunnel for received proxy ID. received local id: 192.168.1.0/24 type IPv4_subnet protocol 0 port 0, received remote id: 172.16.2.0/24 type IPv4_subnet protocol 0 port 0.
07/09 23:04:57	vpn	informational	ike-nego-p2-start	192.168.50.10[500]	IKE phase-2 negotiation is started as responder, quick mode. Initiated SA: 192.168.50.1[500]-192.168.50.10[500] message id:0x5C057ABF.
07/09 23:04:53	vpn	informational	ike-nego-p2-proxy-id-bad	extranet-ike-gateway	IKE phase-2 negotiation failed when processing proxy ID. cannot find matching phase-2 tunnel for received proxy ID. received local id: 192.168.1.0/24 type IPv4_subnet protocol 0 port 0, received remote id: 172.16.2.0/24 type IPv4_subnet protocol 0 port 0.
07/09 23:04:53	vpn	informational	ike-nego-p2-start	192.168.50.10[500]	IKE phase-2 negotiation is started as responder, quick mode. Initiated SA: 192.168.50.1[500]-192.168.50.10[500] message id:0x5C057ABF.
07/09 23:04:53	vpn	informational	ike-nego-p1-succ	extranet-ike-gateway	IKE phase-1 negotiation is succeeded as responder, main mode. Established SA: 192.168.50.1[500]-192.168.50.10[500] cookie:c5e192741b07877c:ba0c67eceff00254 lifetime 28800 Sec.
07/09 23:04:53	vpn	informational	ike-nego-p1-start	extranet-ike-gateway	IKE phase-1 negotiation is started as responder, main mode. Initiated SA: 192.168.50.1[500]-192.168.50.10[500] cookie:c5e192741b07877c:ba0c67eceff00254.

Use the log data to confirm that the firewall now is starting IKE phase-1 and phase-2 processes as the responder. The event name **ike-nego-p2-proxy-id-bad** and the description of this event provide a clear indication of a problem that must be fixed. “Proxy ID bad” means that the proxy ID addresses configured on the firewall do not match the addresses sent from the remote host.

Either end of the connection could be misconfigured. However, with a problem like proxy ID, when the connection attempt fails, the responder does not share the responder’s configuration back to the initiator. Thus, only the responder can say precisely what the problem is. Only the responder has enough information to compare *both* proxy ID configurations.

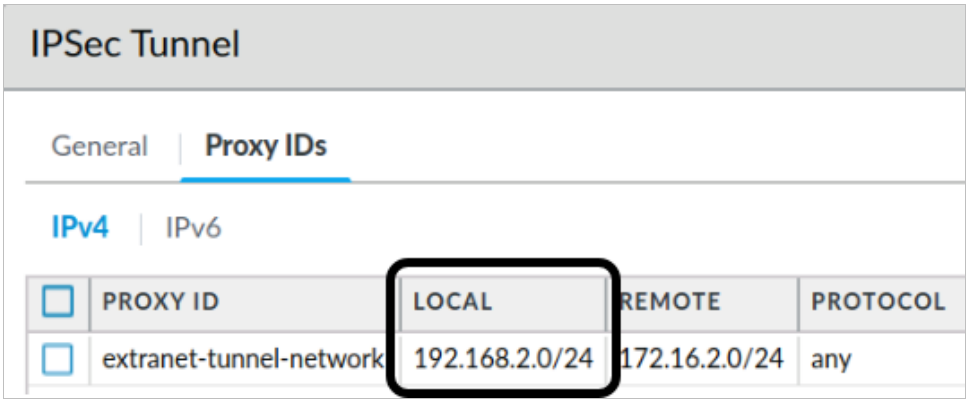
5.1.8 Check Proxy ID Settings and Correct the Problem

30. Go to **Network > IPSec Tunnels** and click **extranet-tunnel**:



The firewall will display the **IPSec Tunnels** configuration dialog.

31. Go to the **Proxy IDs** tab and review the configuration:

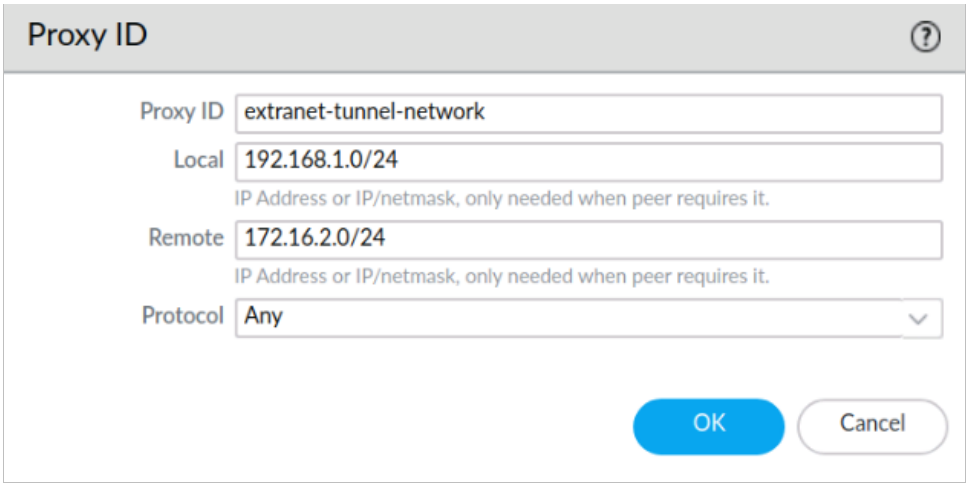


The correct address for the local network is **`192.168.1.0/24`**.

32. In the **Proxy ID** column, click **extranet-tunnel-network**.

The firewall will display the **Proxy ID** configuration dialog.

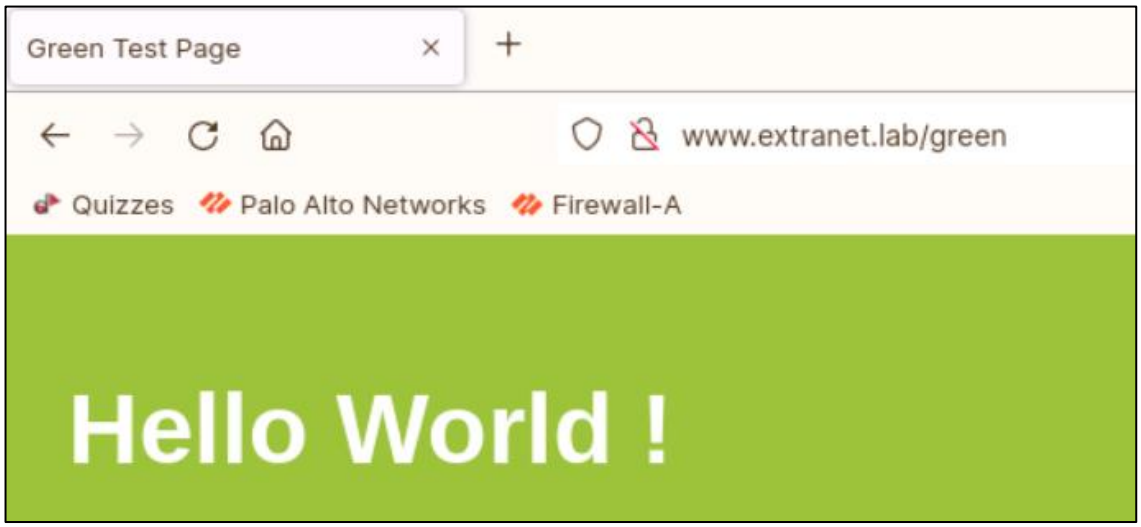
33. Change the **Local** network setting to **`192.168.1.0/24`** and click **OK**:



34. Click **OK** on the **IPSec Tunnels** configuration dialog.
    The window will close.
35. **Commit** the configuration.

5.1.9 Verify the Solution

36. Go to the testing browser and reload the following page:
    `http://www.extranet.lab/green`
    Important: Make sure you use “http://” for the resource specification.



37. Go to the web interface of the firewall and if needed refresh the **Network > IPSec Tunnels** page.
    The status of the tunnel now should be green.

NAME	STATUS	IKE Gateway/Satellite				
		INTERFACE	LOCAL IP	PEER ADDRESS	STATUS	INTERFACE
extranet-tunnel	Tunnel Info	ethernet1/3	192.168.50.1/24	192.168.50.10	IKE Info	tunnel.12

38. Click **Tunnel info** next to the green light in the **Status** column:

Tunnel Info - extranet-tunnel

1 item

NAME	LOCAL IP	LOCAL PORT	PEER IP	REMOTE IP	REMOTE PORT	PKT ENCAP	PKT DECAP	TID	PROTOCOL
extranet-tunnel:extranet-tunnel-network	192.168.50.1	0	192.168.50.10	172.16.2.0/24	0	8	5	1	0

The firewall does not populate this window with information unless a tunnel is established. After a tunnel is established, if you select the tunnel name, then you can click to **Restart** the initiation of the tunnel.

39. Click **Close** to close the **Tunnel Info - extranet-tunnel** window.

5.1.10 Clean Up Your Lab Environment

40. Close the testing browser.
41. Use the **exit** command to close the **Terminal** window on the student desktop.

Stop. This is the end of the lab.

5.2 Lab: Host-Inbound VPN Traffic—Case B

Lab Objectives

- Use the web interface data to check VPN health
- Use log data to identify and diagnose a phase-1 tunnel-establishment issue
- Use the information collected to fix the problem

Lab Scenario

IT administrators are creating helpdesk tickets because they cannot access a certain web server via SFTP. You know that administrator traffic to the web server normally is routed through a VPN tunnel to your main data center. You also know that this traffic recently was redirected back to the main data center after a maintenance cycle.

You need to find the problem and fix it. Is the problem in the routing configuration, the VPN connection, the network connection at the remote site, the host server, or perhaps the target web application?

5.2.1 Apply a Baseline Configuration to the Firewall

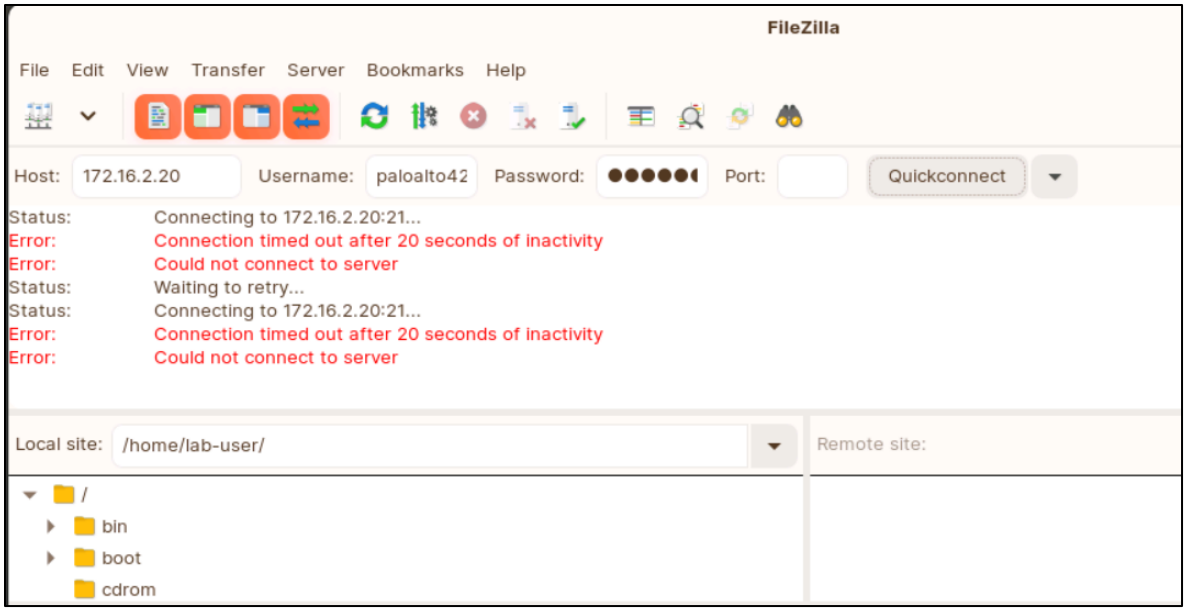
1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-05b.xml**

   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. After the load task is complete, use the web interface to **Commit** the configuration.

5.2.2 Verify the Problem with SFTP Access to the Web Server

3. On the student desktop, double-click **FileZilla**.
4. Type **172.16.2.20** in the Host box, **paloalto42** in the Username box, **Pal0Alt0!** in the Password box, and then select Quickconnect.



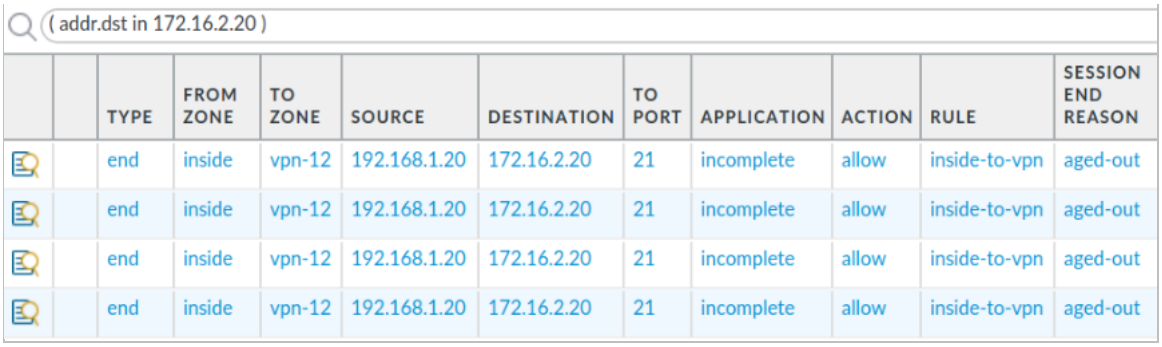
Note the destination host address **172.16.2.20**.

FileZilla will display the progress of the connection attempt and eventually fail to connect to the server.

5. Minimize the **FileZilla** program:

5.2.3 Review the Traffic and System Logs

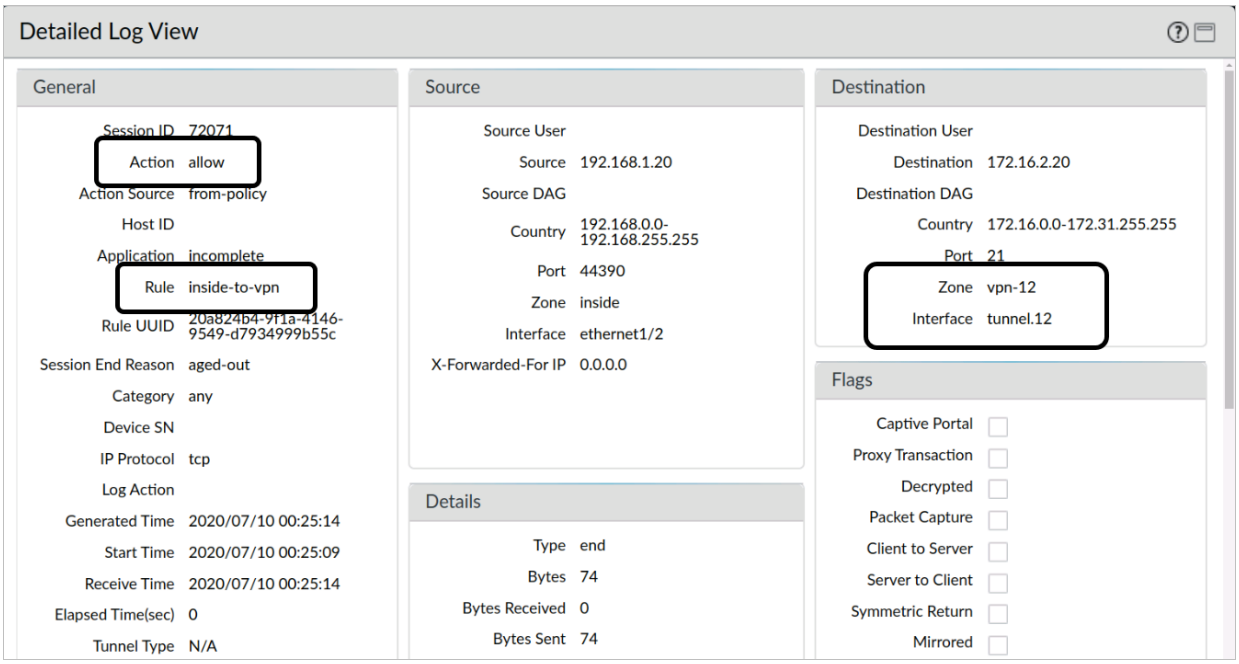
6. In the firewall web interface, go to **Monitor > Logs > Traffic.**
7. Clear any existing filters.
8. In the table of data, click any **Destination** address.
9. Edit the text in the filter box to match the following filter statement:
   `( addr.dst in 172.16.2.20 )`



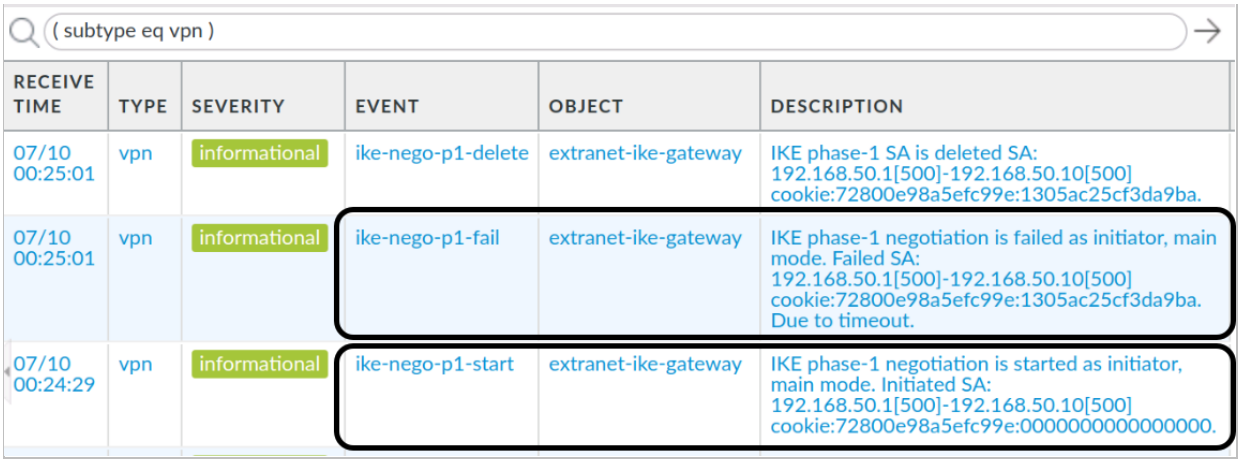
		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		end	inside	vpn-12	192.168.1.20	172.16.2.20	21	incomplete	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.20	21	incomplete	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.20	21	incomplete	allow	inside-to-vpn	aged-out
		end	inside	vpn-12	192.168.1.20	172.16.2.20	21	incomplete	allow	inside-to-vpn	aged-out

10. Click the **Detailed Log View** icon in the left column to review the session details provided by the **Traffic Log View**.

    Confirm that the firewall has logged the traffic, has applied a Security rule that allows the traffic, has routed the traffic to the correct interface, and has not blocked the traffic based on any actions that might be taken by a Security profile.



11. Click **Close** to close the **Detailed Log View** window.
12. Check the system logs. Go to **Monitor > Logs > System**
13. Verify and apply the filter:
    `( subtype eq vpn )`



Important: Depending on the timing of activity in the lab environment, the DMZ server can be aggressive in re-initiating lost tunnel connections. If the DMZ server has re-initiated the connection, you now will see System log information produced by the firewall as the responder. To cause the firewall to initiate the connection, follow the steps in the subsequent section titled "(Optional) Cause the Firewall to Initiate the Connection."

14. Analyze the information provided.

    As the initiator, the firewall reports that the "phase-1 negotiation is failed" and cites the reason as "Due to a timeout." There are two key pieces of information to notice. First, the logs indicate

that the *negotiation* has failed – not the basic connection attempt. Second, timeout failures typically mean the remote server simply did not reply.

If you were to look in the Traffic logs for the traffic between endpoints of the tunnel, that is, 192.168.50.1 and 192.168.50.10, why would you *not* be able to find this traffic?

Answer: The transit-traffic pattern is *intrazone* traffic, which is not logged by default.

5.2.4 Check the High-Level Health Indicators for the Tunnel

15. Go to **Network > IPSec Tunnels**.

    Scan the high-level red-light–green-light indicators to check the status of the tunnel:

		IKE Gateway/Satellite				
NAME	STATUS	INTERFACE	LOCAL IP	PEER ADDRESS	STATUS	INTERFACE
extranet-tunnel	Tunnel Info	ethernet1/3	192.168.50.1/24	192.168.50.10	IKE Info	tunnel.12

    Notice that the tunnel status and **IKE Gateway/Satellite** status are red.

16. Click **IKE Info** and observe that the firewall displays no details.

    On this page, the firewall populates the status-details windows with additional information only after a successful connection has been established.

17. Click to **Close** the window.
18. Click **Tunnel Info** and observe that the firewall also displays no details

    Click to **Close** the window.

5.2.5 Troubleshoot as the Responder

19. On the student desktop, double-click the **Remmina** shortcut.
20. Double-click the **Server-Extranet** configuration in the **Remmina Remote Desktop Client** window.

    This configuration will connect you to the DMZ server that runs VPN software that makes IPsec-tunnel connections to and accepts IPsec-tunnel connections from the firewall.

21. Type:
    `sudo ipsec setup restart` and press **Enter**.

    This command resets the IPsec service on the DMZ server. After the service restarts, the DMZ server will attempt to *initiate* a VPN connection to the firewall.

```
paloalto42@extranet1:~$ sudo ipsec setup restart
Stopping strongSwan IPsec…
Starting strongSwan 5.6.2 IPsec [starter]…
paloalto42@extranet1:~$
```

22. Type:
    `exit` and press **Enter**.

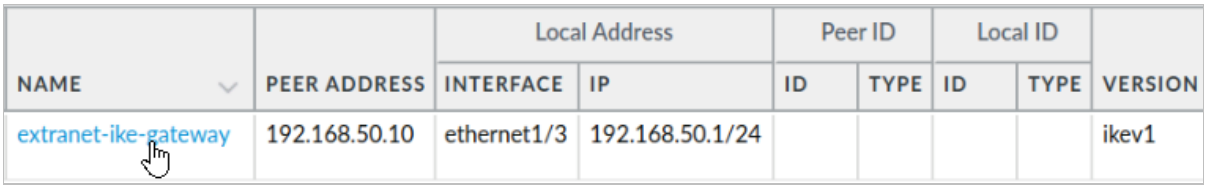
    This command will end the SSH session and the Remmina window will close.

23. Go to the web interface of the firewall and refresh the **Monitor > Logs > System** page.

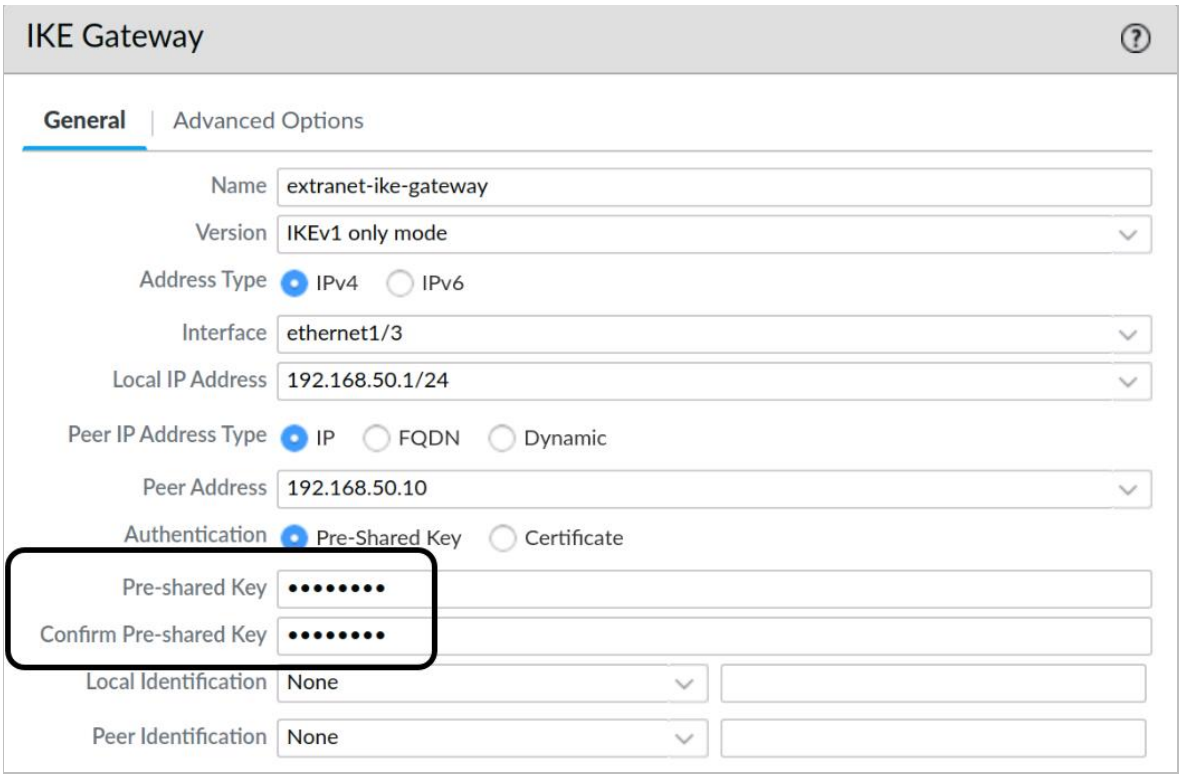
What does the log information indicate?

5.2.6 Reset the Pre-Shared Key and Verify Functionality

24. Go to **Network > Network Profiles > IKE Gateways**:



25. Click **extranet-ike-gateway**:

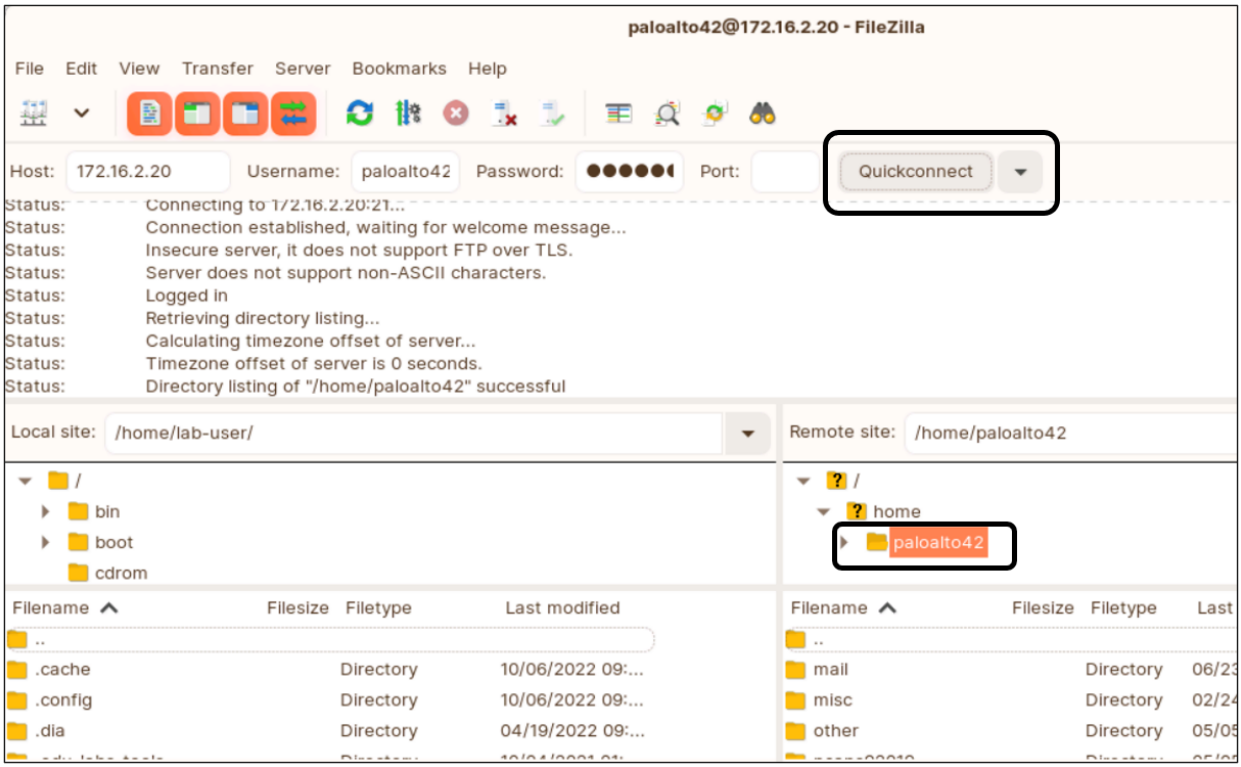


26. Set the pre-shared key to:
    `paloalto`
27. Click **OK**.
28. Click to **Commit** the configuration.

    After you commit the configuration, if you then refresh the **Network > IPSec Tunnels** page, the tunnel status indicator might stay red. To cause the firewall to attempt to establish the tunnel, you can send some traffic to the tunnel. If the status is now green, you now can attempt to connect to the required SFTP destination and determine whether it now is reachable.
29. Restore (or open) the **FileZilla** window and click Quickconnect again to re-establish the connection.

    Verify that the connection succeeds.

The /home/paloalto42 directory should be displayed in the Remote site pane of the application:



30. Close the **FileZilla** program.
31. Go to the web interface of the firewall and refresh the **Network > IPSec Tunnels** page.
    The status of the tunnel now should be green:

		IKE Gateway/Satellite				
NAME	STATUS	INTERFACE	LOCAL IP	PEER ADDRESS	STATUS	INTERFACE
extranet-tunnel	Tunnel Info	ethernet1/3	192.168.50.1/24	192.168.50.10	IKE Info	tunnel.12

32. (Optional) Review the information provided by the **Tunnel Info** and **IKE Info** links.
33. (Optional) Review the **System** and **Traffic** logs to see the negotiation success messages and confirm that FTP traffic to 172.16.2.20 was passed through the tunnel.

5.2.7 (Optional) Cause the Firewall to Initiate the Connection

Depending on the timing of activity in the lab environment, the DMZ server can be aggressive in re-initiating lost tunnel connections. You can use the following steps to cause the firewall to initiate the VPN connection. Sometimes a third-party technical issue or organizational division of responsibility can thwart the diagnostic process, unless you can cause the firewall to initiate the connection.

34. Open **Remmina** and double-click on Firewall-A.

35. In the CLI type:
`test vpn[Tab][Tab]i[Tab]`

This procedure displays the options available for the **`test vpn`** command. All available options for **`test vpn`** begin with "**`i`**." Consequently, after you press **Tab** the second time, the autocomplete feature of the CLI adds an "**`i`**" to the command line. If any options started with a non-unique letter (or letter combination), the CLI simply would display the list of options. In this case, you must press **Tab** three times, whereas typically you would have to press **Tab** only twice.

```
admin@firewall-a> test vpn i[Tab]
> ike-sa     only negotiate IKE SA
> ipsec-sa   negotiate IPSec SA (and IKE SA when necessary)
```

The two command options shown provide you with control over the initiation of phase-1 and phase-2 VPN connections. The following steps include the use of both commands.

36. Type:
`ke[Tab] [Tab]`.

This procedure displays the options available for the **`test vpn ike-sa`** command.

```
admin@firewall-a> test vpn ike-sa [Tab]
+ gateway   test for given IKE gateway
  |         Pipe through a command
  <Enter>   Finish input
```

Notice the firewall provides a **`gateway`** option so that you can specify an individual gateway.

37. Type:
`gate[Tab] [Tab]`

This procedure will add **`gateway`** to the current command string and then list the gateways that are configured on the system.

```
admin@firewall-a> test vpn ike-sa gateway [Tab]
  extranet-ike-gateway   extranet-ike-gateway
  <value>                test for given IKE gateway
```

38. Type:
`extra[Tab]` and press **Enter**.

This procedure will add the name of the target gateway to the current command string and execute the command. The command itself initiates phase 1 of the VPN connection.

```
admin@firewall-a> test vpn ike-sa gateway extranet-ike-gateway [Enter]

Start time: Nov.07 07:16:31
Initiate 1 IKE SA.
```

The command in the next step also will initiate the phase-1 negotiation (if necessary) and phase-2 negotiation.

39. Type:
    `test vpn ipsec-sa [Tab]`
    This procedure shows the options for the **`test vpn ipsec-sa`** command, which negotiates IPSec SA and, when necessary, IKE SA.

```
admin@firewall-a> test vpn ipsec-sa [Tab]
+ tunnel   test for given VPN tunnel
  |        Pipe through a command
  <Enter>  Finish input
```

40. Type:
    `tun[Tab] [Tab]`
    This procedure shows the available configurations that you can specify for the tunnel option. In the current lab configuration, only one tunnel configuration is available.

```
admin@firewall-a> test vpn ipsec-sa tunnel [Tab]
  extranet-tunnel:extranet-tunnel-network   extranet-tunnel:extranet-tunnel-network
  <value>                                   test for given VPN tunnel
```

41. Type:
    `extr[Tab]` and press **Enter**.
    This procedure uses autocomplete to append the name of the tunnel configuration to the command string and executes the command.

```
admin@firewall-a> test vpn
 ipsec-sa tunnel extranet-tunnel:extranet-tunnel-network [Enter]

Start time: Nov.07 07:45:09
Initiate 1 IPSec SA for tunnel extranet-tunnel:extranet-tunnel-network.
```

42. Check the logs (ideally on the responder) for information about the results of the connection attempt.
    However, in the lab, you likely will be using these commands to compare for yourself the difference between log data generated as initiator and log data generated as responder.

5.2.8 Clean Up Your Lab Environment

43. Close any open Remmina SSH connections.

Stop. This is the end of the lab.

5.3 (Optional) Troubleshoot VPN connectivity independently – Case C

This lab is optional, please first check with your instructor if you should complete this exercise.

Lab Objectives

- Use the tools and techniques discussed during the lesson to independently troubleshoot and resolve a VPN connectivity issue.

Lab Scenario

One of the system administrators is reporting an issue that he cannot ping an IP address 172.16.2.11 which is reachable via an IPSEC VPN between the FireWall and the Extranet-Server. The IPSEC VPN is called “VPN-to-DMZ-Server”.

In this lab you will independently troubleshoot and solve the connectivity via an IPSEC VPN using the tools and techniques discussed during the lesson. All issues should be resolved on the FireWall and not on Client-A or the Extranet-Server. This lab does not include any step-by-step instructions nor solution guide. Please check with your instructor if you require any assistance.

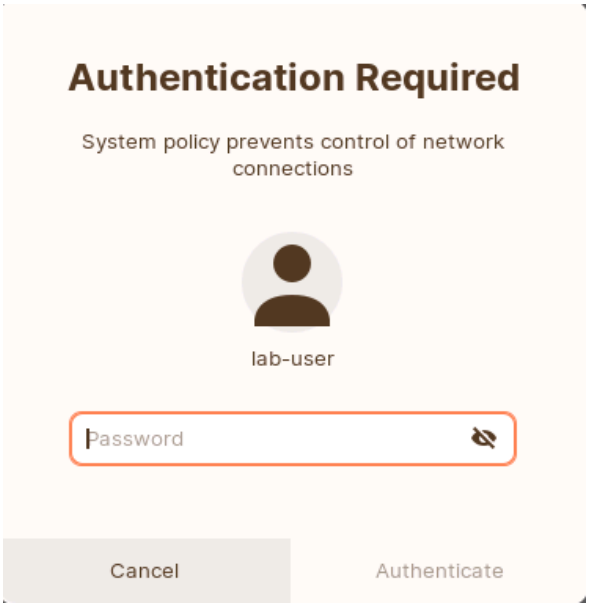
5.3.1 Apply a Baseline Configuration to the Firewall

1. Use the Firefox web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-05c.xml**

   Import the file from the **/home/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. After the load task is complete, use the web interface to **Commit** the configuration.

   You might get a popup window warning you that system policy prevents control of network connections.



Use the password “Pal0Alt0!” and click “Authenticate” if you get the above pop-up window

5.3.2 Troubleshoot and Resolve

1. Identify and correct this problem so that you can ping 172.16.2.11 from Client-A. The Extranet-Server regularly tries to establish the VPN. If you want to reset the connection manually, then you can log into the Extranet-Server (Remmina profile “Server-Extranet”) and reset the VPN using the command **sudo ipsec restart**.

Stop. This is the end of the Troubleshoot VPN connectivity independently – Case C lab.

6.1 Lab: Transit Traffic—App-ID and Torrents

Lab Objectives

- Create an application-aware Security policy rule to block torrents
- Test application blocking with different configurations
- Identify how the App-ID “web-browsing” is matched to a session
- Identify how URL filtering and App-ID work together to create better security
- Review session-end descriptors and how to interpret them

Lab Scenario

This lab requires you to resolve typical challenges related to the interpretation of Traffic logs and how to get the results you expect. You will be required to diagnose and implement solutions for allowing or blocking a particular kind of traffic, such as torrent traffic.

6.1.1 Apply a Baseline Configuration to the Firewall

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-06.xml**

   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. After the load task is complete, use the web interface to **Commit** the configuration.
3. Verify that the Policy Rule 2 is disabled.

   Note: On the **Policies > Security** page, **Policy Rule 2** is disabled:

	NAME	TAGS	Source ZONE	Destination ZONE	APPLICATION	SERVICE	ACTION
1	egress-outside-app-id	egress	inside	outside	dns google-base shutterfly ssl web-browsing	application-default	Allow
2	*egress-outside*	*egress*	*inside*	*outside*	*any*	*application-default*	*Allow*
3	internal-dmz-ftp	internal	inside	dmz	ftp	application-default	Allow
4	intrazone-default	none	any	(intrazone)	any	any	Allow
5	interzone-default	none	any	any	any	any	Deny

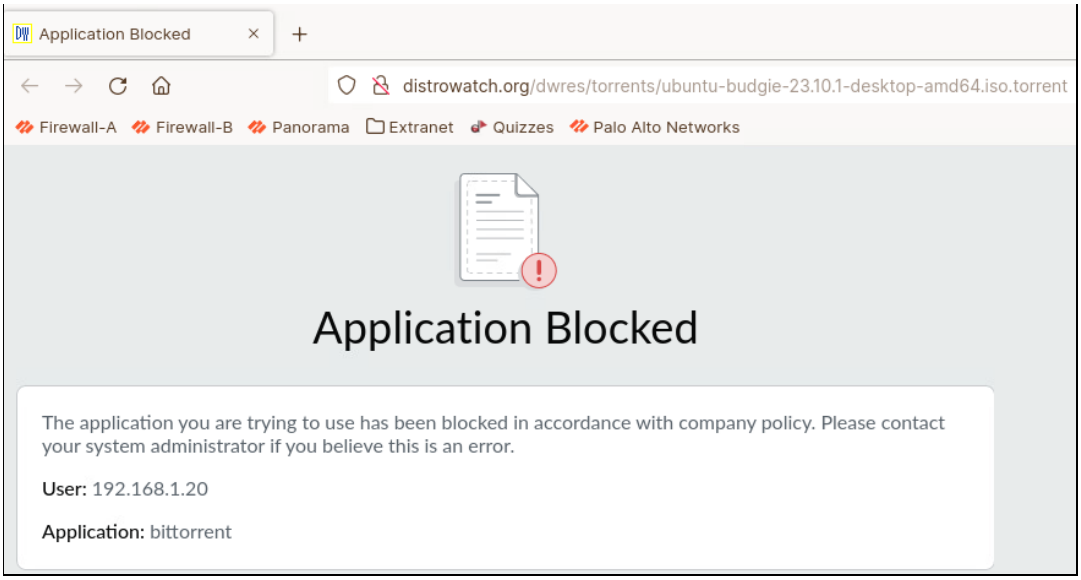
6.1.2 Attempt to Download Torrent File

4. On the student desktop, open Firefox.
5. Connect to **distrowatch.org**
6. Select the Torrent Downloads link from the upper right corner of the page.
7. Scroll down the list of Projects until you locate entries for Ubuntu.
8. Click the link for any Ubuntu torrent file:

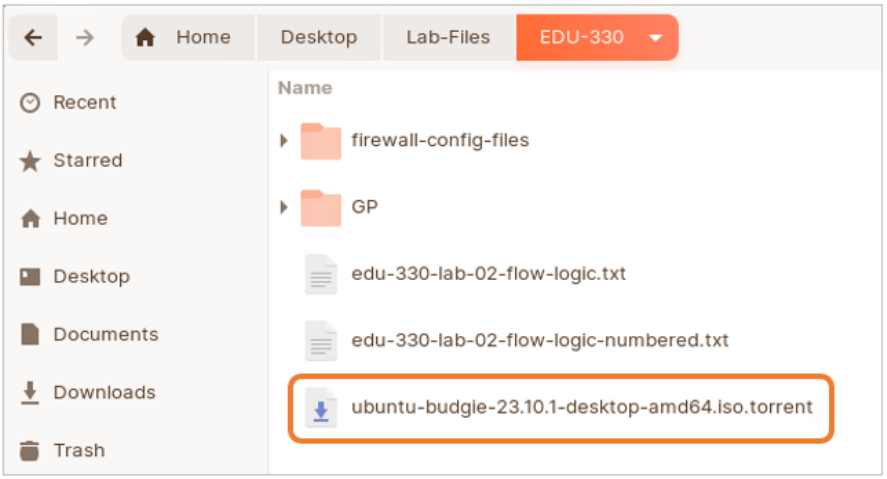
Ubuntu Budgie	ubuntu-budgie-23.10.1-desktop-amd64.iso.torrent
Ubuntu	ubuntu-23.10.1-desktop-amd64.iso.torrent
MX Linux	MX-23.1_x64.iso.torrent
Xubuntu	xubuntu-23.10-desktop-amd64.iso.torrent
Ubuntu Unity	ubuntuunity-23.10-desktop-amd64.iso.torrent
Ubuntu Studio	ubuntustudio-23.10-dvd-amd64.iso.torrent

Note that the version and date you see in this list will differ. You do not need to download these exact files – you can try downloading any Ubuntu torrent file for this lab.

9. The firewall will block the application:



10. Close Firefox.
11. On the client desktop, open the **`Lab-Files`** folder.
12. Open the **`EDU-330`** folder.
13. Double-click the ubuntu torrent file listed in the folder:

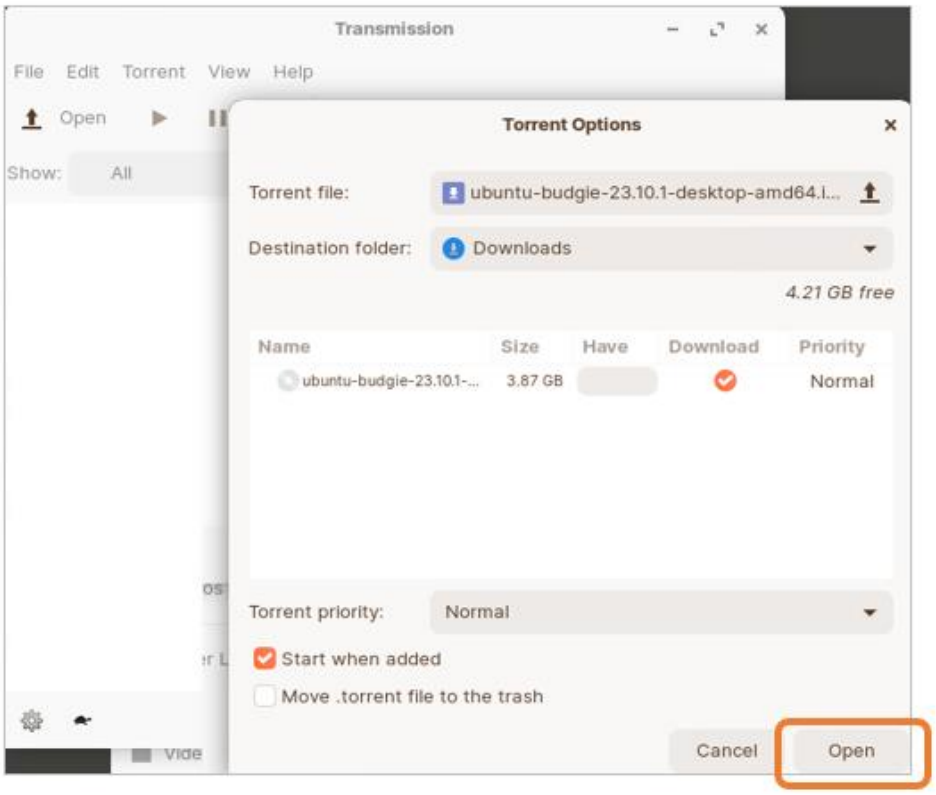


The file you see may differ from the example shown above.

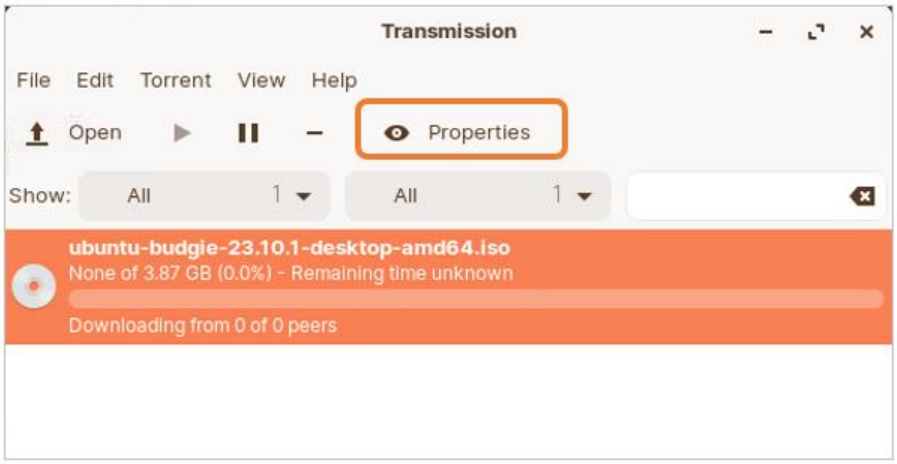
Since the firewall blocked your earlier attempt to download a torrent file, this *.torrent file has been placed on the client host so you can see how the firewall handles bittorrent peer traffic.

Double-clicking the file will open the Transmission torrent application.

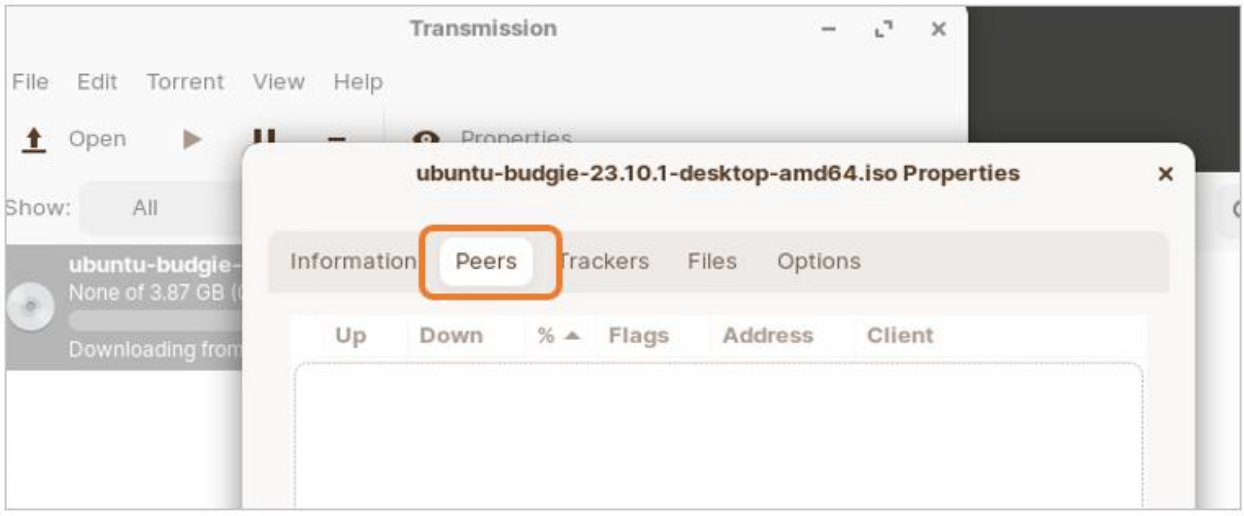
14. In the Torrent Options window, leave the settings unchanged and click **Open**.



15. In Transmission, highlight the entry for the torrent file and click **Properties**.



16. Select the **Peers** tab.



The **Peers** list likely will show no entries and no indication of activity.

The **Transmission** torrent application apparently cannot reach external peers, even though explicitly you have not created any Security policy rules to block torrents.

Is the firewall actually blocking connections to torrent peers and file transfers? If so, which aspects of the current configuration are producing this result? Subsequent lab steps will help you to answer these questions.

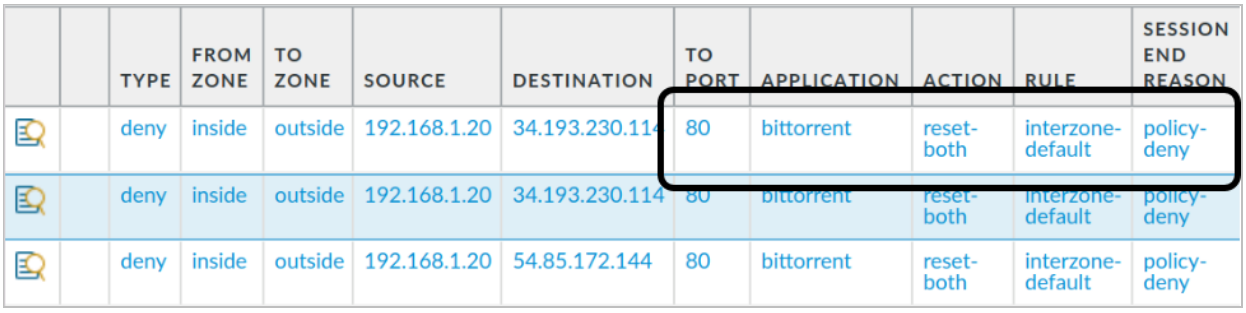
17. Click **Close** at the bottom of that window.
18. Right-click the name of the file that the torrent program is attempting to download and select **Delete Files and Remove**.

    Removing the torrent from the list will stop ongoing connection attempts and thereby reduce the number of Traffic log entries to search through. Notice that there is no indication that torrent files are being downloaded.
19. Click **Delete** to confirm.
20. Close the Transmission application.

6.1.3 Examine Traffic Logs and App-ID Results

21. Go to the web interface of the firewall.
22. Click **Monitor > Logs > Traffic** and inspect the Traffic logs. Add or remove columns as needed.
23. Filter the Traffic logs for the source address of the student desktop:
    `( addr.src in 192.168.1.20 )`

24. Use your knowledge of filters and how to navigate the log viewer to find a log entry for which the **bittorrent** application has been detected:



		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		deny	inside	outside	192.168.1.20	34.193.230.114	80	bittorrent	reset-both	interzone-default	policy-deny
		deny	inside	outside	192.168.1.20	34.193.230.114	80	bittorrent	reset-both	interzone-default	policy-deny
		deny	inside	outside	192.168.1.20	54.85.172.144	80	bittorrent	reset-both	interzone-default	policy-deny

Notice that the rule **interzone-default** is blocking the bittorrent application. Why is the "interzone-default" rule blocking bittorrent?

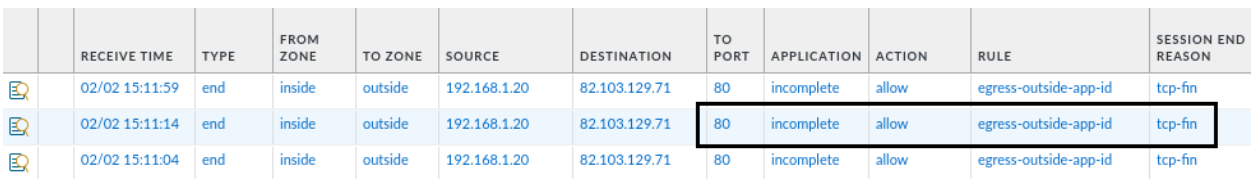
25. In the **Application** column, find a log entry with **not-applicable** listed.
You should have two or more logged instances:

		TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		drop	inside	outside	192.168.1.20	197.57.142.252	49229	not-applicable	deny	interzone-default	policy-deny
		drop	inside	outside	192.168.1.20	95.87.35.187	6881	not-applicable	deny	interzone-default	policy-deny
		drop	inside	outside	192.168.1.20	5.228.123.15	6881	not-applicable	deny	interzone-default	policy-deny

The firewall assigns the application name **not-applicable** if the traffic matches to a rule that explicitly denies the destination *port* (or *service*). The firewall also will assign the application "not-applicable" when an *implicit* block occurs for traffic with a destination port that is not otherwise allowed by the sum of all Security policy rules, regardless of the actual application.

In the example, the destination port 6881 is not a default (or *well-known*) port for any of the applications that explicitly are allowed by the Security policy rules. The "drop" result of the "deny" action specified by the matching Security rule is the product of a total Security policy in which each individual rule is limited to the *application-default* service or a custom *Service* object or *Service Group* object that defines a limited number of ports for each specified application. In this Security policy, no rule permits the use of "any" application and no rule permits an allowed application to use "any" port for interzone traffic.

26. Try to find one or more log entries that lists the **Application** type as **incomplete**:



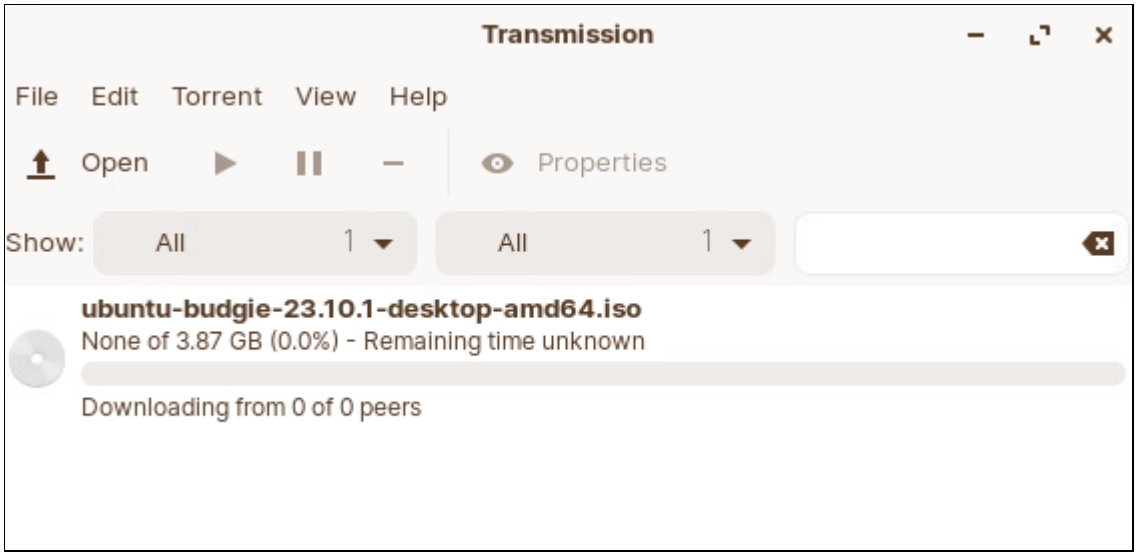
		RECEIVE TIME	TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		02/02 15:11:59	end	inside	outside	192.168.1.20	82.103.129.71	80	incomplete	allow	egress-outside-app-id	tcp-fin
		02/02 15:11:14	end	inside	outside	192.168.1.20	82.103.129.71	80	incomplete	allow	egress-outside-app-id	tcp-fin
		02/02 15:11:04	end	inside	outside	192.168.1.20	82.103.129.71	80	incomplete	allow	egress-outside-app-id	tcp-fin

An application entry of **incomplete** means that the TCP handshake did not complete. In such cases, no payload data is transmitted; thus, no App-ID signature matching is performed.

An application entry of **insufficient-data** for a TCP connection means that the three-way handshake completed but the data sent subsequently, typically just a single packet, did not match an App-ID signature. You likely will *not* have an example of "insufficient-data" in the Traffic logs of your current lab environment.

6.1.4 Enable Traffic

27. Go to **Policies** > **Security**, enable the **egress-outside** rule, and **Commit** the changes.
    Note that the "egress-outside" rule is set to "Allow." After you enable this rule, torrents and other traffic that match to this rule will be allowed. The "interzone-default" rule will not be applied.
28. Return to the **`/home/lab-user/Desktop/Lab-Files/EDU-330/`** folder and double-click on the torrent file again to launch the **Transmission** application.
29. In Transmission, verify that the torrent is listed:

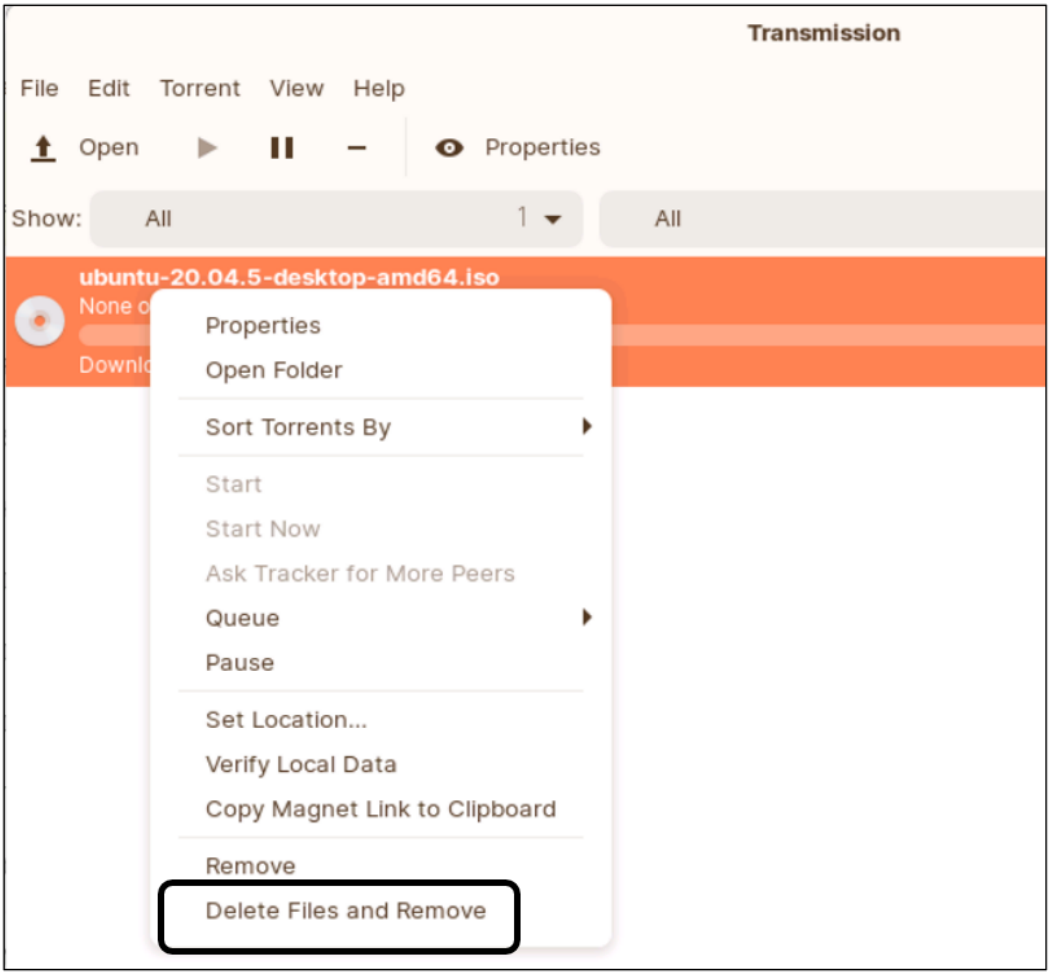


30. Go to the web interface of the firewall.
31. Click **Monitor** > **Logs** > **Traffic** and inspect the Traffic logs.
32. Filter the Traffic logs for the bittorrent application:
    `( app eq 'bittorrent' )`
33. Use the refresh button to update the Traffic log until you see several entries for bittorrent traffic:

(app eq 'bittorrent')

		RECEIVE TIME	TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
		11/02 18:07:46	end	inside	outside	192.168.1.20	195.191.244.15	1145	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:46	end	inside	outside	192.168.1.20	5.189.188.23	46913	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:36	end	inside	outside	192.168.1.20	91.134.159.168	51413	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:36	end	inside	outside	192.168.1.20	89.149.207.193	28013	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:36	end	inside	outside	192.168.1.20	213.136.79.27	12251	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:36	end	inside	outside	192.168.1.20	38.60.245.4	51291	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:36	end	inside	outside	192.168.1.20	95.111.230.250	60442	bittorrent	allow	egress-outside	aged-out
		11/02 18:07:01	end	inside	outside	192.168.1.20	5.75.137.186	32942	bittorrent	allow	egress-outside	aged-out

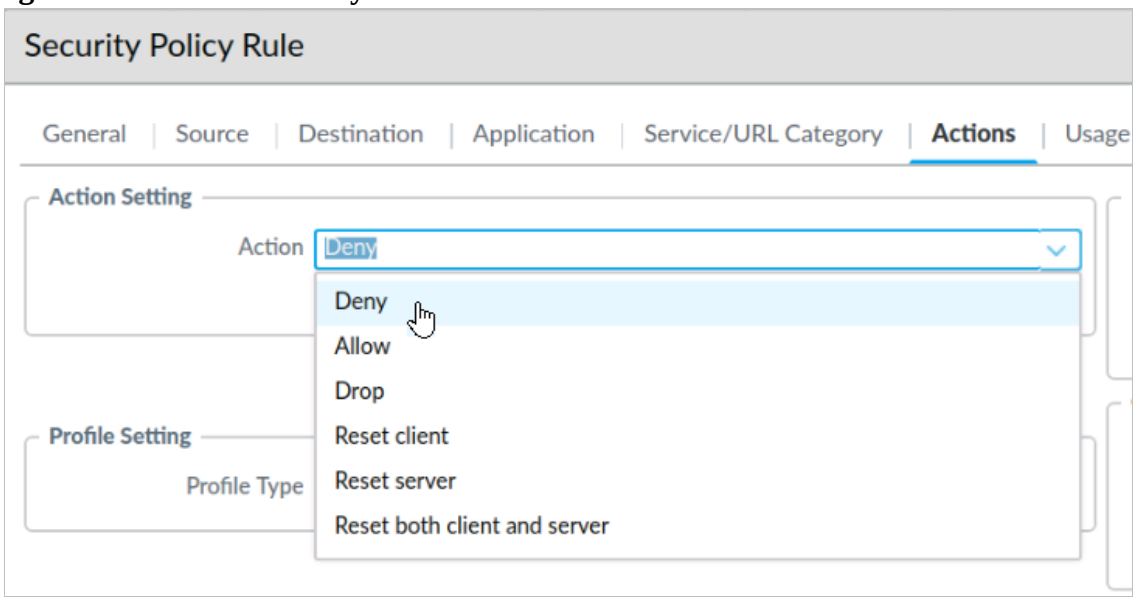
34. Note the name of the rule which matches bittorrent traffic – **`egress-outside`**.
35. In Transmission, right-click the name of the file being downloaded, select **Delete Files and Remove,** and **Delete** again in the confirmation.



36. Leave Transmission open.

6.1.5 Set the Matching Policy Rule to “Deny” and Test

37. In the firewall web interface, go to **Policies > Security**, change the **Action Setting** on the **egress-outside** rule to **Deny** and click **OK**:



38. **Commit** the changes.
39. On the client Desktop, open the Lab-Files/EDU-330 folder and double-click the *.torrent file.
40. Click **Open** in the Torrent Options window to start the torrent download in Transmission.
41. In the firewall web interface, examine the Traffic Logs again:

(app eq 'bittorrent')

RECEIVE TIME	TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
11/02 18:15:41	deny	inside	outside	192.168.1.20	1.4.150.130	5432	bittorrent	reset-both	egress-outside	policy-deny
11/02 18:15:41	deny	inside	outside	192.168.1.20	5.79.148.175	35562	bittorrent	reset-both	egress-outside	policy-deny
11/02 18:15:41	deny	inside	outside	192.168.1.20	37.61.78.140	32168	bittorrent	reset-both	egress-outside	policy-deny
11/02 18:15:41	deny	inside	outside	192.168.1.20	101.69.118.28	33032	bittorrent	reset-both	egress-outside	policy-deny
11/02 18:15:41	deny	inside	outside	192.168.1.20	119.167.234.22	6882	bittorrent	reset-both	egress-outside	policy-deny

Which rule now is blocking the torrents?

Answer: The “egress-outside “policy rule is blocking the torrents.

Can you use Firefox to access torrent-related sites?

Answer: Yes, if the application is recognized as **web-browsing** or **ssl**.

Can you successfully download a torrent file?

Answer: No. The torrent traffic is blocked by the “egress-outside” rule.

Many sessions related to Transmission activity are identified as **web-browsing**. What likely is happening inside most of these sessions?

Answer: A TCP handshake on port 80 is completed. The torrent client makes an HTTP request, which the firewall can use to identify the application as "web-browsing." But the destination server then either immediately closes the session or simply does not respond. Most of these sessions include only three or four received packets.

6.1.6 Create a Policy Rule to Block Torrents

42. Use your knowledge of the firewall to modify the Security rule **egress-outside** to *block* torrents explicitly.

    Use the **Application** filter to ensure all torrents can be blocked.

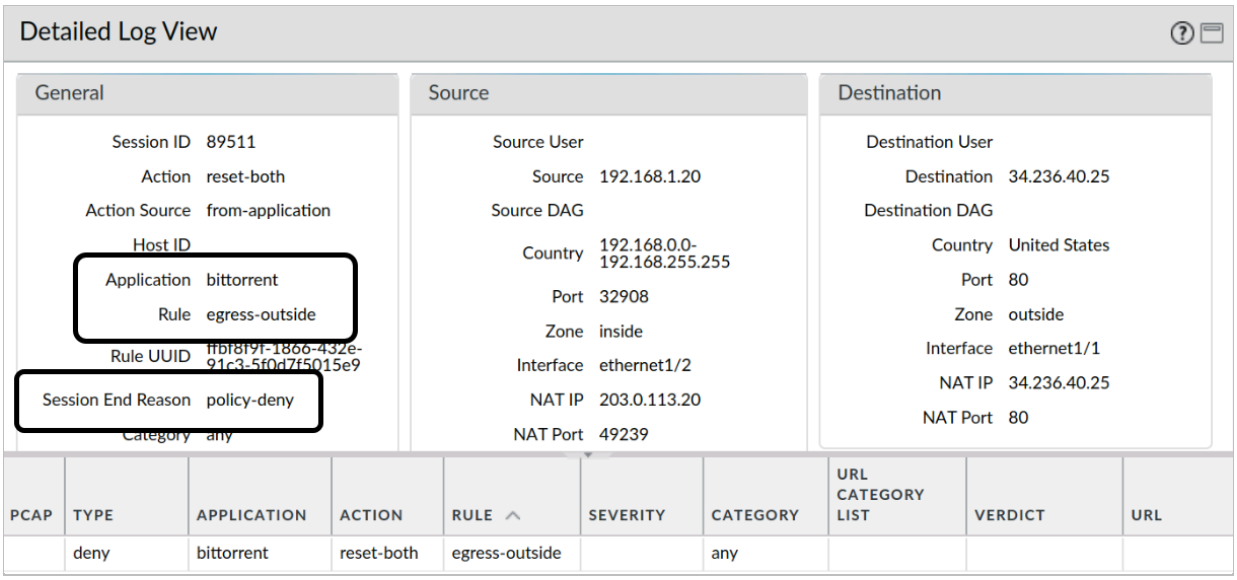
43. Run tests to verify that your rule is working.
44. Use the Traffic logs to identify the policy rules that are matched to sessions that end in **policy-deny**:

	TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
	deny	inside	outside	192.168.1.20	34.236.40.25	80	bittorrent	reset-both	egress-outside	policy-deny
	deny	inside	outside	192.168.1.20	18.235.131.250	80	bittorrent	reset-both	egress-outside	policy-deny
	deny	inside	outside	192.168.1.20	34.193.230.114	80	bittorrent	reset-both	egress-outside	policy-deny
	deny	inside	outside	192.168.1.20	34.193.230.114	80	bittorrent	reset-both	egress-outside	policy-deny
	deny	inside	outside	192.168.1.20	34.193.230.114	80	bittorrent	reset-both	egress-outside	policy-deny
	deny	inside	outside	192.168.1.20	54.209.162.80	80	bittorrent	reset-both	egress-outside	policy-deny

Note: The example is for illustrative purposes only. Your logs may look somewhat different.

Traffic that results in "policy-deny" may have an **Action** listed as "reset-both" or "deny," depending on which rule is related. Formulating answers to the following questions will help you better understand the difference between "drop" and "deny."

- How do the various actions and session-end listed in the Traffic log relate to the traffic?
- Can you still browse to different torrent sites, and can you still download torrent files?
- What can you do to prevent browsing for and downloading standalone torrent files?
- Would blocking torrent sites using categories with URL filtering help?



6.1.7 Add File Blocking to the Security Profile Setting

Torrent *client* activity generates network data that can be detected as an *application*. Individual files that typically end with the extension ".torrent" contain metadata about how a target file can be downloaded by subsequent use of a torrent client.

Torrent metadata files potentially can be hosted on variously categorized or uncategorized sites that are not excluded by the running Security policy.

A Security profile that includes File Blocking for individual torrent files themselves can provide an additional layer of defense against torrent-related activity.

45. In the web interface, go to **Objects > Security Profiles > File Blocking**:

NAME	LOCATION	RULE NAME	APPLICATIONS	FILE TYPES	DIRECTION	ACTION
basic file blocking	Predefined	Block high risk file types	any	7z, bat, chm, class, cpl, dll, exe, hlp, hta, jar, ocx, PE, pif, rar, scr, torrent, vbe, wsf	both	block
		Continue prompt encrypted files	any	encrypted-rar, encrypted-zip	both	continue
		Log all other file types	any	any	both	alert
strict file blocking	Predefined	Block all risky file types	any	7z, bat, cab, chm, class, cpl, dll, exe, flash, hlp, hta, jar, msi, Multi-Level-Encoding, ocx, PE, pif, rar, scr, tar, torrent, vbe, wsf	both	block
		Block encrypted files	any	encrypted-rar, encrypted-zip	both	block
		Log all other file types	any	any	both	alert

Note that both Security profile options for File Blocking that are built into the default firewall configuration include "torrent" as a file type to block.

46. Use your knowledge of the firewall to add the **strict-file-blocking** profile to the **egress-outside** policy rule.

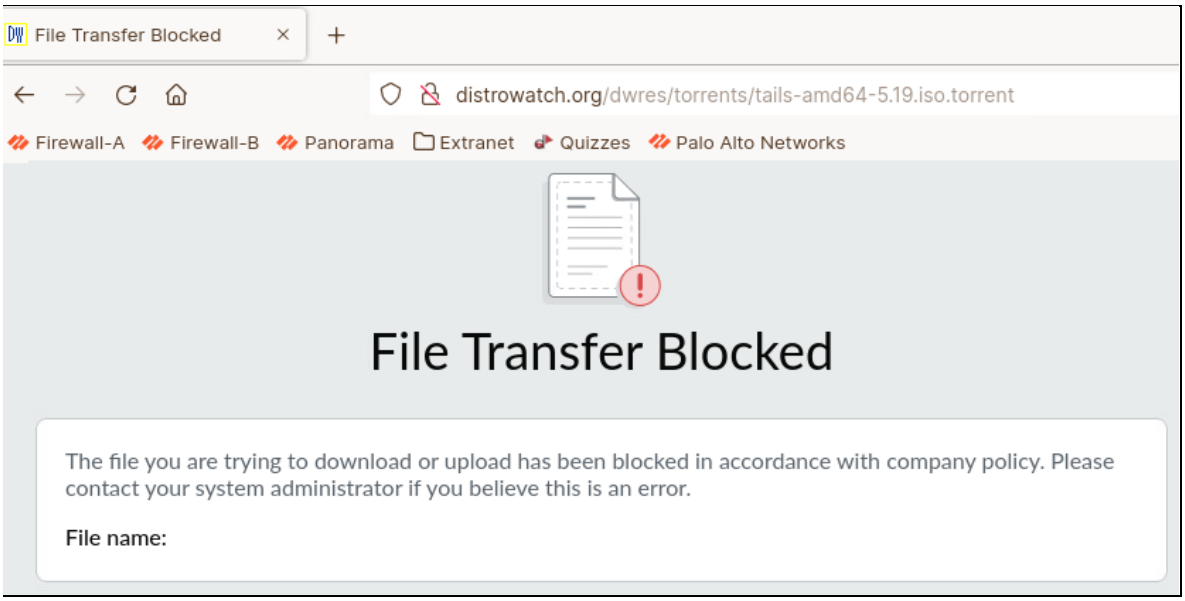
    Why would this profile be ineffective if you added it to the **egress-outside** rule as it is currently configured?

    Answer: You need to apply the File Blocking profile to traffic that is *allowed* as web-browsing (or other traffic) but that has *not* already been identified as *bittorrent*.

47. Change the **Action** for the egress-outside rule from **Deny** to **Allow**.
    This change means that the firewall will not deny bittorrent traffic but will block *.torrent file downloads.

48. **Commit** the changes.

49. Run tests to discover how this rule changes the way the firewall processes torrent-related traffic. Refer to previous steps if needed.



These exercises are designed to illustrate how you can configure the firewall with layers of protection to prevent unwanted traffic such as bittorrent. You can allow only certain applications; you can explicitly block applications by creating a Deny rule and adding the applications you want to block; you can use Security Profiles to block file type downloads; you can use a URL Filtering profile to block categories of websites; or you can combine one or more of these methods in the same set of Security Policy rules.

6.1.8 Clean Up Your Lab Environment

50. Go to the Transmission application. Click **File** > **Quit**.

51. Then click **Yes** to confirm that you want to exit the application.

52. Close Firefox.
53. Close any open File Manager windows on the client desktop.

Stop. This is the end of the lab.

6.2 Lab: Transit Traffic—Blocking Tor

Lab Objectives

Configure the firewall to:

- Allow internal hosts to connect to the internet using standard browsers
- Block attempts to use anonymization tools such as the Tor (The Onion Router) network

Lab Scenario

This lab focuses on another typically unwanted class of applications: Tor and similar anonymizers:

- The Tor network conceals user identity by proxying encrypted connections through a network of servers that fragment and randomize connection events in such a way that data connections cannot be easily traced back to the user. If you attempt to trace Tor traffic backward from destination-server to source-client, you will see connection requests from random nodes on the Tor network, instead of the true source node of the user.

6.2.1 Lab Challenge and Checklist

Palo Alto Networks has created App-IDs such as **tor** and **tor2web** to identify Tor connections. Like any other anonymizer, Tor uses different techniques to bypass your security. This lab asks you to test whether blocking **tor** and **tor2web** with the use of App-ID is enough to block the unwanted traffic.

Consider options from the following task list to help ensure that you can block Tor application traffic in your lab environment:

- Create a Security policy rule called **egress-deny-torrents** to block Tor based on App-ID
- Use App-ID application filters such as: **tor** and **tor2web**
- Block risky URL categories
- Deny unknown applications
- Block untrusted and expired certificates with a Decryption profile
- Turn on SSL decryption
- Configure source-based and destination-based control using EDLs

Blocking evasive application such as Tor requires a combination of methods. Use as many capabilities of the firewall as needed to properly block Tor.

Sometimes a session can end for multiple reasons. For example, after the firewall drops a session upon detecting a threat, a host might send a TCP FIN message to terminate the same session. Also, the Traffic log might record events for multiple sessions (for example, ping), each with a separate end reason. In these cases, the session end reason field displays only the highest priority reason.

Stop. First, try to solve the lab problem on your own.
The steps in the following section provide an example diagnosis and solution.
Do not proceed without permission from your instructor.

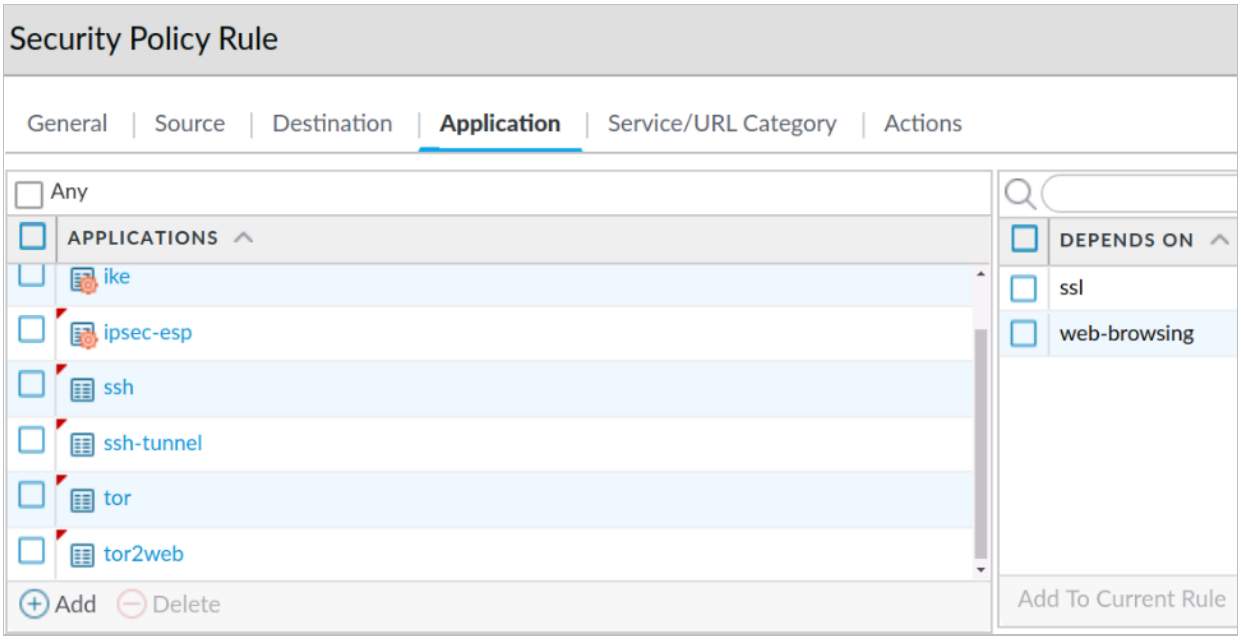
6.2.2 Lab Solution: Security Policy to Block Tor App-ID

In the following steps, you will create a Security policy called **egress-deny-tor-browser** to block internet access to the following applications:

- http-proxy
- ike
- ipsec-esp
- ssh
- ssh-tunnel
- tor
- tor2web

1. In the web interface of the firewall, go to **Policy > Security**.
2. Create a rule called **egress-deny-tor-browser** that denies access to the applications in the list.

   Tip: Ensure that the **Service** configuration is set to **application-default**.



Note: You might consider also blocking “torch”-related applications.

	NAME	TAGS	Source ZONE	Destination ZONE	APPLICATION	SERVICE	ACTION
1	egress-outside-app-id	egress	inside	outside	dns google-base shutterfly ssl web-browsing	application-default	Allow
2	egress-outside	egress	inside	outside	bittorrent	application-default	Deny
3	egress-deny-tor-browser	none	inside	outside	http-proxy ike ipsec-esp ssh ssh-tunnel tor tor2web	application-default	Deny

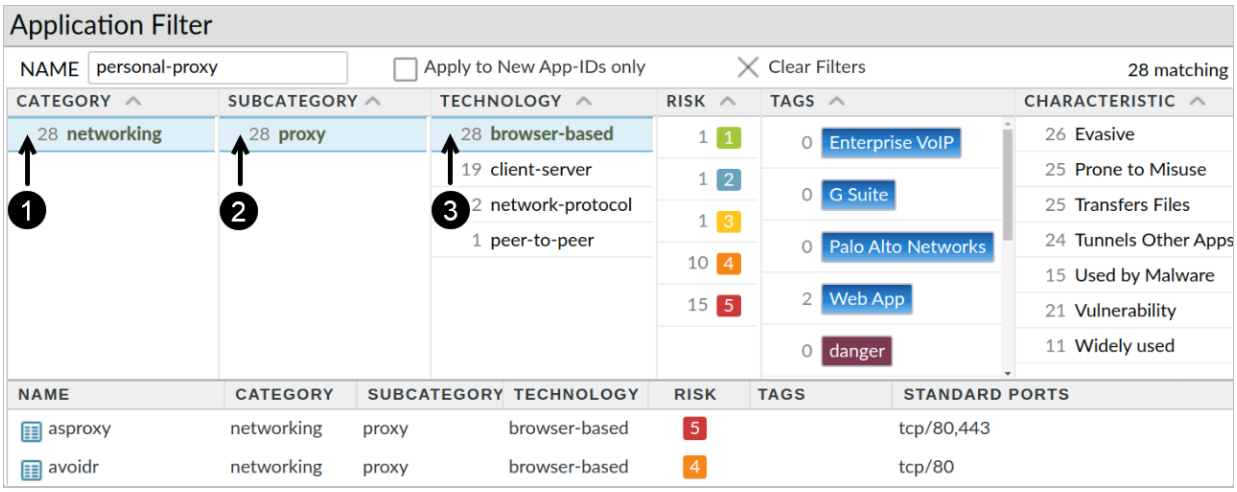
6.2.3 Lab Solution: Use Application Filters

High-quality avoidance software is available and is constantly being improved. User demand for tools that bypass network restrictions is strong. The Application Filter feature can help block applications based on a dynamic categorization of behavior; it does not require you to add new or modified individual App-IDs to Security policy rules manually.

The Application Filter capability dynamically groups applications based on category.

3. Go to **Objects > Application Filters** and create a new group of applications (example name **`personal-proxy`**) based on:
   - The category **networking**
   - Subcategory **proxy**
   - Technology **browser-based** (select **Show Technology Column** at bottom of application Filter window)
   - Characteristic all (or undefined)

   This filter includes applications such as “psiphon” and “tor2web.”
4. Click **OK**.

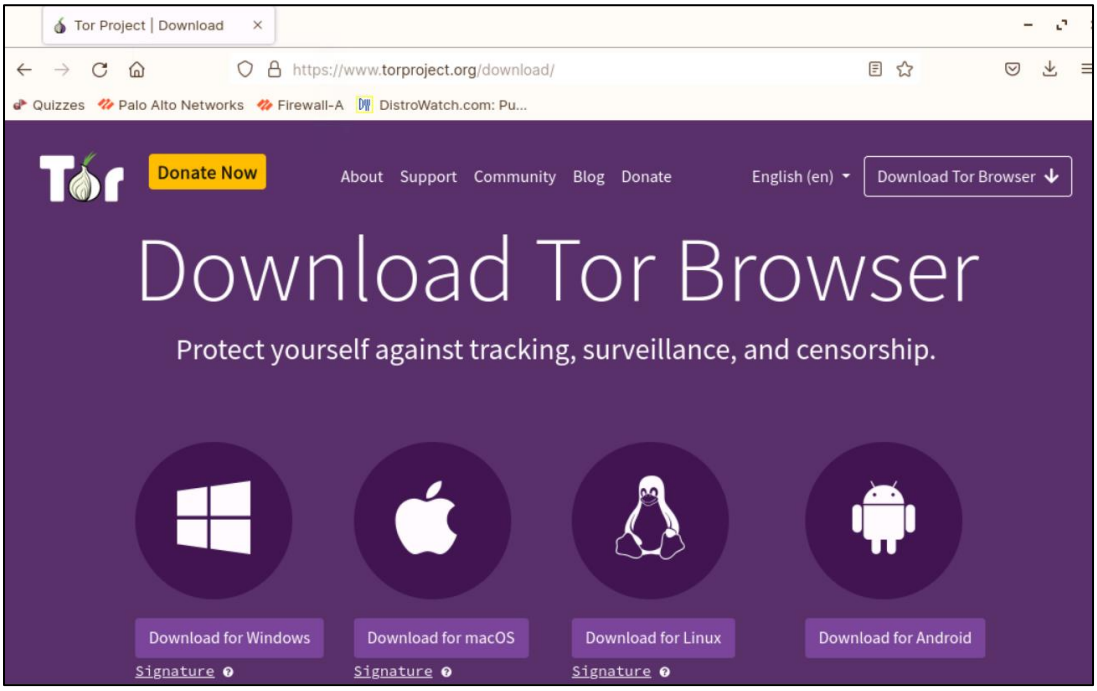


5. Go to **Policies > Security** and create a Security policy rule to block applications that match to the application filter **personal-proxy** and set the action to **Deny**.

   Note: When you whitelist (as opposed to blacklist) applications in your Security policy, use "application-default" for the Service. The firewall compares the port used with the list of standard or typical ports for that application. If the port used is not a default port for the application, the firewall will drop the session and write "appid policy lookup deny" in the log.

6. Assess whether Tor still can be used by accessing **https://www.torproject.org/download/**.

   One way to test whether Tor is being blocked as desired would be to request an update to the Tor browser (if available):



   What else must be done to block Tor?

Note: As the precision and coverage of App-ID signatures increases, the effectiveness of Security policy rules that use the basic App-ID signatures to detect and respond to target application types also increases.

In the lab environment, various potential evasion and obfuscation techniques that applications such as Tor now can or will be able to perform are unlikely to be seen.

Solutions provided in subsequent sections of this lab activity provide a conceptual roadmap of recommended practices for building a layered defense against multiple threat vectors for which any one application, such as Tor, is just an example.

6.2.4 Lab Solution: Block Risky URL Categories

7. Create a URL Filtering profile that, at minimum, blocks access to the following categories:
   - copyright-infringement
   - dynamic-dns
   - malware
   - parked
   - phishing
   - proxy-avoidance-and-anonymizers
   - questionable
   - unknown
8. Go to **Objects > Security Profiles > URL Filtering**. Find each category and block access to them.
9. Associate the URL Filtering profile to a Security policy rule.

6.2.5 Lab Solution: Deny Unknown Applications

Security professionals often recommend that you should block any applications that are categorized as “unknown-tcp,” “unknown-udp,” and “unknown-p2p.”

10. Implement a rule that denies access to “unknown” applications.
11. Test the effect of this rule on blocking the use of Tor.

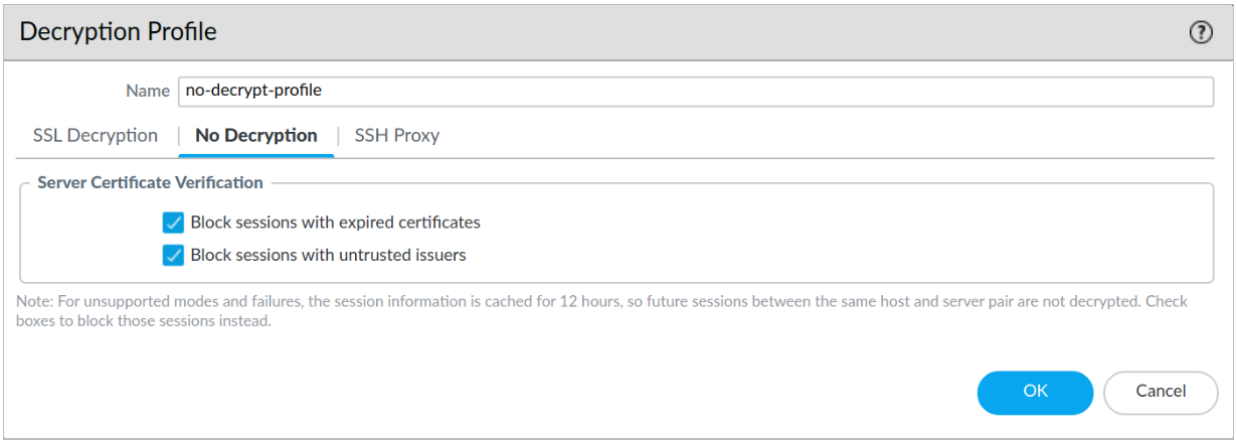
    Note: If your users need to access sanctioned applications that are detected as “unknown-tcp” or “unknown-udp,” create a Security policy that allows that traffic only on the specific ports used by the sanctioned application.

6.2.6 Lab Solution: Blocking Untrusted and Expired Certificates with a Decryption Profile

The following steps will create a Decryption profile as part of a no-decrypt rule to limit Tor from being able to make connections.

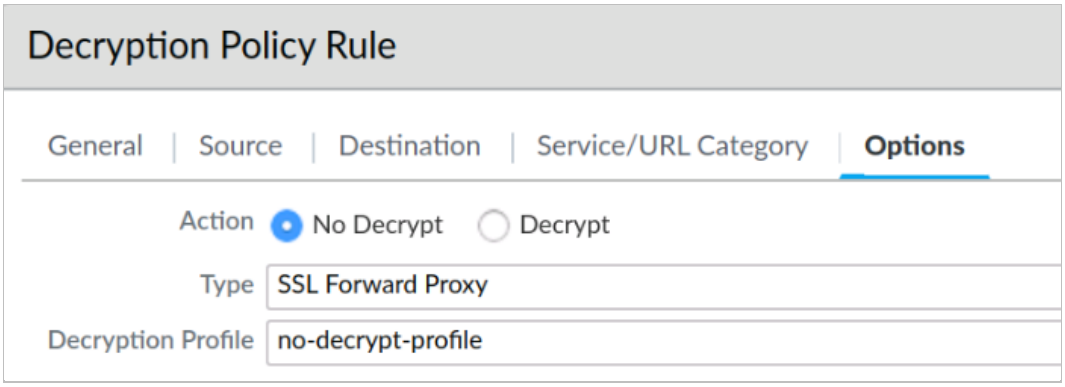
> You can block connections with certificate problems without decrypting SSL. This action can be quite effective in blocking Tor.

12. Go to **Objects > Decryption Profile**.
13. Click **Add**.
14. Name the profile **`no-decrypt-profile`**.
15. On the **No Decryption** tab, select **Block sessions with expired certificates** and **Block sessions with untrusted issuers** and click **OK**:

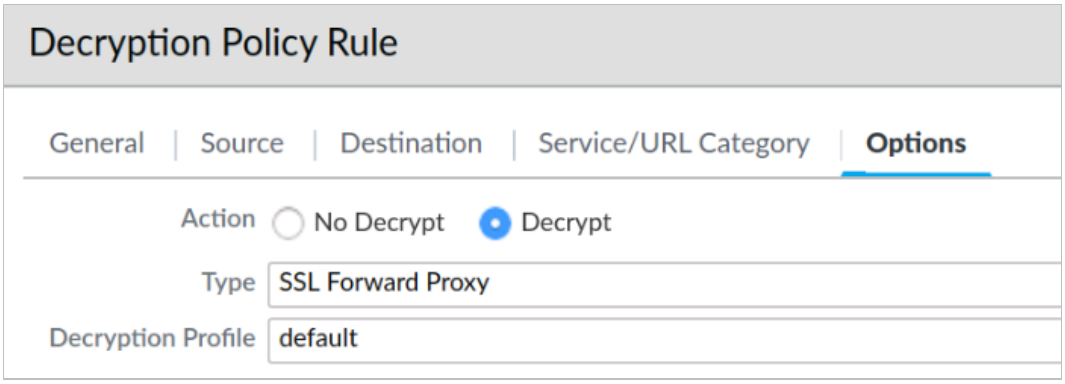


16. Go to **Policies > Decryption**.
17. Click **Add** and name the rule **`no-decrypt`**.
18. Add a brief description that includes the following: what was changed, why was the change made, who made the change, and the date and time.
19. Set the **Source** zone to **inside** and the **Destination** zone **outside**.
20. Click the **Service/URL Category** tab and add URL categories that should not be decrypted, such as “financial-services.”
21. Click the **Options** tab.

22. Set the **Action** to **No Decrypt** and the **Decryption Profile** to **no-decrypt-profile** and then click **OK**:



23. Click **Add** to create a second Decryption policy rule and name it **`decrypt`**.
24. Add a brief description that includes the following: what was changed, why was the change made, who made the change, and the date and time.
25. Set the **Source** zone to **inside** and the **Destination** zone **outside**.
26. Click the **Options** tab.
27. Set the **Action** to **Decrypt** and the Decryption profile to **default** and then click **OK**:



6.2.7 Lab Solution: Create Decryption Profile for Decrypted Traffic

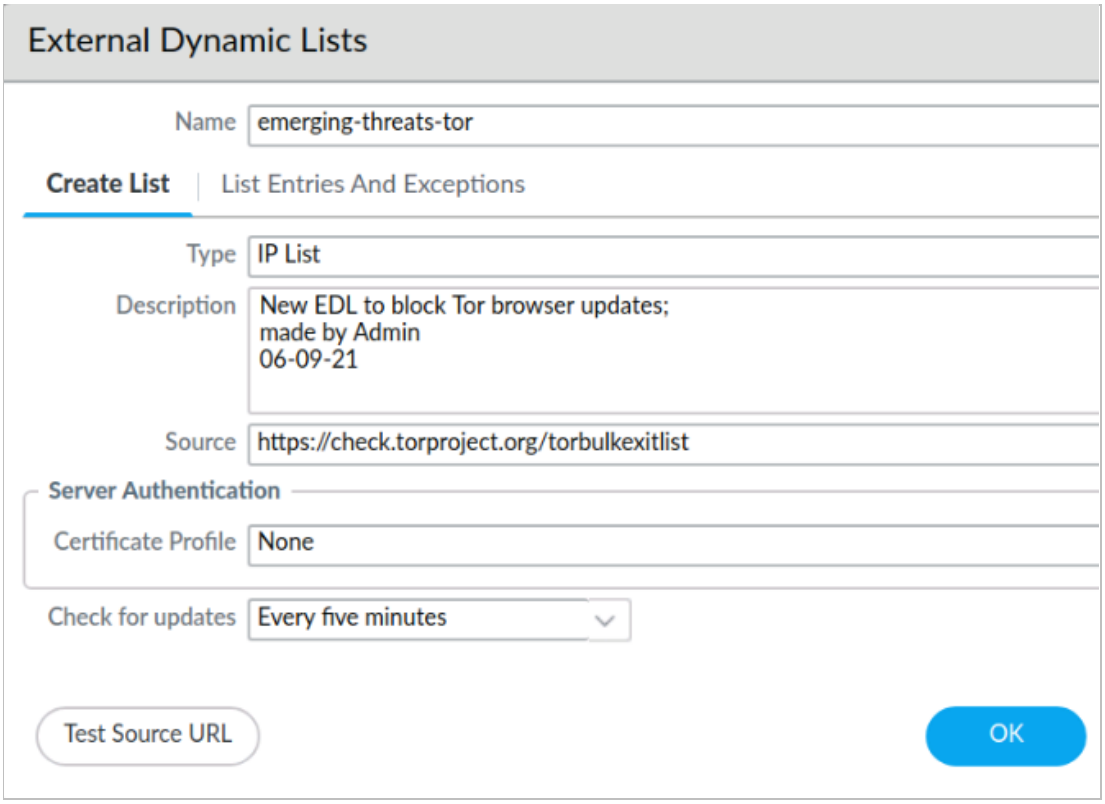
28. Go to **Objects > Decryption Profile**.
29. Click **Add** and give the profile a name, such as **`decrypt-profile`**.
30. Select the options for **Server Certificate Verification** and **Unsupported Mode Checks** on the SSL Decryption tab.
31. Finish the profile configuration with any other selections of your choosing.
32. Go to **Policies > Decryption**.
33. Associate the **decrypt** policy to the Decryption profile you just created.

6.2.8 Lab Solution: Use an External Dynamic List (EDL)

In addition to the precautions taken in previous steps, you can use the firewall's support for EDLs to block other types of connections related to the Tor network. Security policy rules that define source or destination traffic based on an EDL of IP addresses maintained by external threat researchers can be used to block and/or log traffic that matches entries in the list. Use of the correct EDL may help block Tor browser updates and possibly other Tor-related activity.

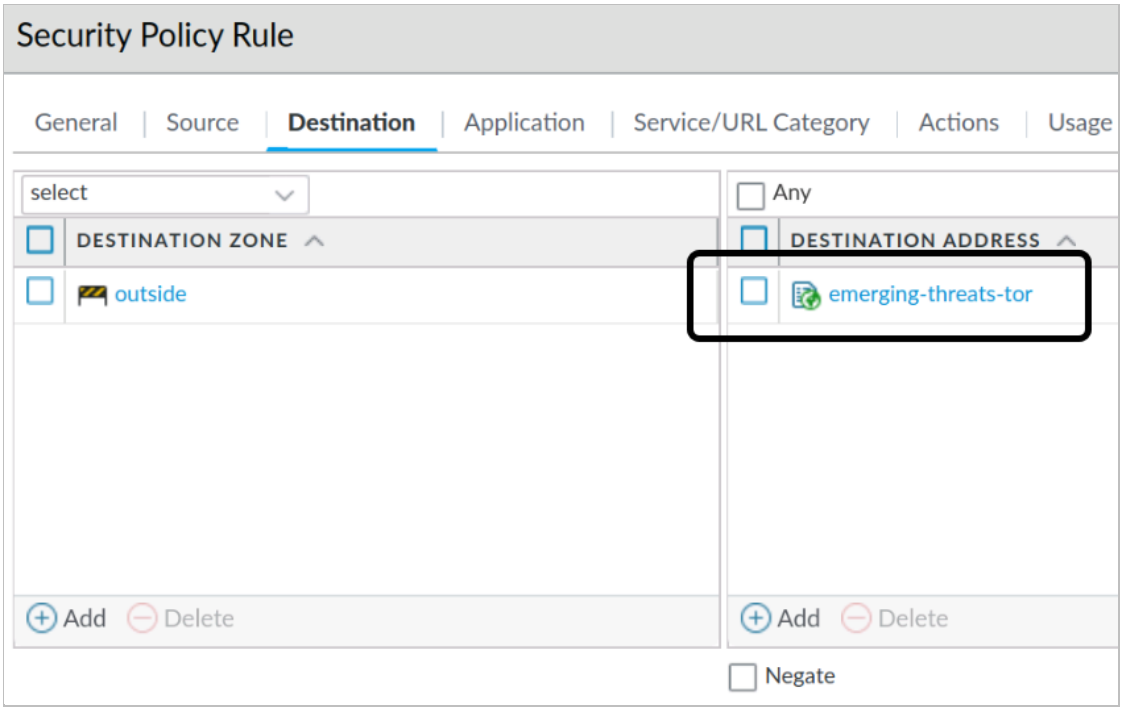
For example, the URL "https://check.torproject.org/torbulkexitlist" hosts a frequently updated EDL of Tor exit nodes, routers, and relays. The firewall can download the EDL at a configured interval and, thereby, dynamically update the match conditions for a policy rule.

34. Go to **Objects > External Dynamic Lists**.
35. Click **Add** and name the EDL entry **`emerging-threats-tor`**.
36. Enter the following URL in the **Source** field:
    `https://check.torproject.org/torbulkexitlist`
37. Add a brief description that includes the following: what was changed, why was the change made, who made the change, and the date and time.
38. Select **Test Source URL** to verify connectivity to the URL.
39. Close the popup confirmation message.
40. Leave all other setting unchanged.



41. Click **OK** to close External Dynamic Lists dialog box.

42. Go to **Policies > Security**.
43. **Add** a new security rule that uses the EDL as the definition for the **Destination Address** with the **Destination Zone** set to outside:



44. Set the **Action** to **deny**.
45. Complete the rule configuration with any other selections of your choosing.
46. **Commit** the configuration.
47. Test to assess whether Tor can be used. Note if attaching the EDL makes any difference.

6.2.9 Clean Up Your Lab Environment

48. Close all open testing browser tabs and windows. Clear any filters in logs.

    Leave open the browser and the connection to the web interface of the firewall.

49. If you have an open CLI connection to the firewall, leave the connection open for the next lab.

Stop. This is the end of the lab.

6.3 (Optional) Troubleshoot Internet connectivity

This lab is optional, please first check with your instructor if you should complete this exercise.

Lab Objectives

- Use the tools and techniques discussed during the lesson to independently troubleshoot transit traffic passing through the FireWall

Lab Scenario

Users are complaining that they are not able to access the Internet. In this lab, you will troubleshoot and solve three problems on your own using the tools and techniques discussed during the lesson. All issues should be resolved on the FireWall and not on Client-A. This lab does not include any step-by-step instructions nor solution guide. Please check with your instructor if you require any assistance.

6.3.1 Check Running Services

50. Use the Firefox web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-06b.xml**

    Import the file from the **/home/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

51. After the load task is complete, use the web interface to **Commit** the configuration.

6.3.2 Troubleshoot and Resolve

52. Identify and correct this problem so that you can access **www.bbc.com** from Client-A
53. After you have resolved the Internet connectivity problem, users still complain that certain websites like http://apache.org only load partially or slowly. If you are not able to replicate the issue, then please try other websites like http://example.com, http://w3.org, http://gnu.org, http://sectigo.com

Stop. This is the end of the Troubleshoot Internet connectivity lab.

7. Lab: System Services

Lab Objectives

- Use the CLI to display system services status
- Use the CLI to start, stop, and restart a service
- Locate and display service (daemon) logs

Lab Scenario

You want to perform a basic health check of your system:

- Are all services running?
- What is the current CPU use?

Also, you want to walk through the processes required to change the log level for one or more services, and you want to see how to restart a service.

7.1 Check Running Services

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-07.xml**
   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.
2. After the load task is complete, use the web interface to **Commit** the configuration.
   Verify the **Result** reported is "Successful" and the **Details** include "Configuration committed successfully." Use **Remmina** to open an SSH session to **Firewall-A**. If you have an existing session already active, use it.
3. Click **Close**.
4. Verify that the CLI is in *operational mode*. The prompt sign should be "**>**".
   If the prompt sign is "**#**," the CLI is in configuration mode; type **`exit`** and press **Enter**.
5. Type:
   `show system software status` and press **Enter**.
   This command lists core firewall services and service groups and their state. When a service is listed as "stopped," additional information may be provided:

```
admin@firewall-a> show system software status [Enter]

Slot 1, Role mp
---------------------------------------
Type     Name                State    Info
Group    all                 running
Group    base                running
```

```
Group    batch                running
Group    batch_secondary      running
Group    chassis              running
Group    data_plane           running
[. . .]
Process  all_task             running  (pid: 3258)
Process  authd                running  (pid: 3450)
Process  bfd                  running  (pid: 3466)
Process  brdagent             running  (pid: 3071)
Process  chasd                running  (pid: 2871)
Process  comm                 running  (pid: 3528)
[. . .]
```

Are all services running?

6. Find the authentication service (**`authd`**) and note (write down) its process ID (**`pid`**) by using **`/authd`**.

   (Optional) Note one or more other services of interest to you. Your task simply will be to correlate the basic status provided here with the CPU and memory information that you will display in step 8.

7. After you run **`show system software status`**, if the firewall does not return you to the command prompt, press the **spacebar** as needed (or **q**) to return to the command prompt.

8. Type:
   `show system resources` and press **Enter**.
   This command displays system-level and process-level details about resource use:

```
admin@firewall-a> show system resources [Enter]

top - 21:43:52 up 11 days,  6:52,  1 user,  load average: 0.00, 0.05, 0.05
Tasks: 130 total,   2 running, 128 sleeping,   0 stopped,   0 zombie
Cpu(s):   0.7 us,  1.0 sy,  0.2 ni, 97.9 id,  0.2 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem:   7967.1 total,     136.1 free,   4369.7 used,   3461.4 buff/cache
MiB Swap:  4000.0 total,    3469.3 free,    530.5 used.   1123.5 avail Mem

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 7949       30  10 1564m 1.3g 1.3g S 23.9 30.8   0:00.12 panio
 2555       15  -5 90716 3944  872 S  4.0  0.1  22:28.94 sysd
 3258       20   0 1589m 1.4g 1.3g S  4.0 31.3 762:42.38 pan_task
 3486       30  10  136m 9256 1220 S  4.0  0.2   4:21.27 python
 3094       20   0     0    0    0 R  2.0  0.0  29:12.04 kni_single
    1       20   0 16548  152    0 S  0.0  0.0   0:05.16 init
    2       20   0     0    0    0 S  0.0  0.0   0:00.01 kthreadd
[. . .]
```

9. Look for the **authd** service.

   Which percentage of CPU and memory is the **authd** service currently using?

10. Type:
    show system resources follow and press **Enter**.
    This command displays system-level and process-level details about resource use:

```
admin@firewall-a> show system resources follow [Enter]
top - 20:44:19 up 8 days,  5:31,  1 user,  load average: 0.94, 0.45, 0.33
Tasks: 254 total,   2 running, 252 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.3 us,  1.0 sy,  0.0 ni, 91.4 id,  0.7 wa,  1.5 hi,  0.0 si,  0.0 st
MiB Mem :   7967.1 total,    135.0 free,   4364.7 used,   3467.4 buff/cache
MiB Swap:   4000.0 total,   3469.5 free,    530.5 used.   1130.0 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
 6772 root      20   0   67.1g   2.7g   2.7g S  10.0  34.6   1272:08 pan_task
 5288 root      20   0       0      0      0 S   1.7   0.0 194:10.22 kni_single
 3354 root       0 -20 3257056   2.7g   2.7g S   0.7  34.6  37:48.03 md_apps
 5438 nobody    20   0   71244   5300   1396 S   0.3   0.1   6:51.32 redis-server
 5444 nobody    20   0   59468   1688   1380 S   0.3   0.0   6:38.44 redis-server
[. . .]
```

    What does pressing the **spacebar** do when you use the **follow** option?

    Which key can you press to learn more about how to manipulate this information? (Answer: **h**)

11. Press **q** to quit the program and return to the command prompt.

7.2 Review the Logs for a Specific Service

One way to review firewall logs is to generate a Tech Support File and offline review the logs collected. However, there may be times when you already are focused on a specific service (daemon) and know what to look for in a specific log file. The CLI provides **less**, **grep**, and **tail** commands to display and search log data.

12. Type:
    less mp-log (add a space if you did not use autocomplete) and press **Tab** and **Enter** and then type **y**.
    This procedure displays a list of all available management-plane logs:

```
admin@firewall-a> less mp-log
Display all 173 possibilities? (y or n)
agent                          appweb3-panmodule.log
bfd.log                        botnet.log
brdagent.log                   brdagent.log.old
cgroups.log                    cgroups_details.log
chasd.log                      check_plugin_compat.log
configd.log                    configd.log.old
content_telemetry.log          contentd.log
cryptod.log                    cryptod.log.old
csad.log                       curlog_out_content
```

```
curlog_out_sig_av             curlog_out_sig_iot
curlog_out_sig_wildfire       dagger.log
device_certgen.log            device_telemetry.log
[. . .]
```

Notice the various logs available. Note that on VM-Series firewalls all logs are stored on the management plane. On firewalls that have separate physical data planes, you can display all available data-plane logs with the command **`less dp-log [Tab]`**.

13. In the list of available log options, locate the **`md_info.log`** file.
14. Type
    `less mp-log md_info.log` and press **Enter**.
    This command displays the log file **`md_info.log`**:

```
admin@firewall-a> less mp-log md_info.log [Enter]

--------------------------------------------------------------------------------
2023-04-12 14:15:26.611 +0000 INFO: Main Script Log Initialized
--------------------------------------------------------------------------------
2023-04-12 14:15:27.324 +0000 INFO: sysd: initialized
2023-04-12 14:15:27.331 +0000 INFO: sysd: process running with pid 10627
2023-04-12 14:15:32.726 +0000 INFO: all: initialized
2023-04-12 14:15:32.726 +0000 INFO: gdb: initialized
2023-04-12 14:15:32.727 +0000 INFO: sysdagent: initialized
2023-04-12 14:15:32.730 +0000 INFO: configd: initialized
[. . .]
2023-04-12 14:15:32.740 +0000 INFO: ha_agent: initialized
/Script
```

The log output is displayed through the **`less`** command. The log data that you see traces the system startup process for those elements of the application stack that the master daemon launches. The file is partitioned with a human-readable header. Your output will differ somewhat from the example shown.

15. Type:
    `/Script` and press **Enter**.
    This **`less`** command searches for "Script," which appears in the log-section heading "Main Script Log Initialized."
16. Press the **`n`** key.
    This **`less`** command searches for the *next* instances of "Script."
17. Press **`n`** repeatedly until you see "**`Pattern not found (Press RETURN)`**" at the bottom of the terminal window:

```
2023-04-12 14:15:26.611 +0000 INFO: Main Script Log Initialized
--------------------------------------------------------------------------------
2023-04-25 15:13:35.855 +0000 INFO: dnsproxy: initialized
2023-04-25 15:13:35.916 +0000 INFO: redis_csad: initialized
2023-04-25 15:13:35.917 +0000 INFO: csad: initialized
2023-04-25 15:13:35.917 +0000 INFO: pppoe: initialized
```

```
2023-04-25 15:13:35.920 +0000 INFO: dsms: initialized
2023-04-25 15:13:35.927 +0000 INFO: redis_dscd: initialized
2023-04-25 15:13:36.035 +0000 INFO: data_plane: initialized
Pattern not found  (press RETURN)
```

18. Press **Enter**.

For illustration, we will focus on the **authd** process.

19. Type:
    /authd and press **Enter**.
    This procedure searches for the next instance of "authd" in the log file from the top of the window:

```
2023-04-25 15:13:39.311 +0000 INFO: authd: initialized
2023-04-25 15:13:39.312 +0000 INFO: satd: initialized
2023-04-25 15:13:39.736 +0000 INFO: gdb: process running with pid 3722
2023-04-25 15:13:39.873 +0000 INFO: monitor: process running with pid 3723 [. . .]
```

You should see a log entry that identifies the time the service was initialized. If at any point you need to return to the top of the log, press **g**.

20. Press **n** to see the next match:

```
2023-04-25 15:13:56.742 +0000 INFO: authd: process running with pid 5613
2023-04-25 15:13:56.744 +0000 INFO: dhcp: process running with pid 5614
2023-04-25 15:13:56.746 +0000 INFO: dnsproxy: process running with pid 5615
2023-04-25 15:13:56.748 +0000 INFO: gp_broker: process running with pid 5616
```

You should see a log entry that identifies the time the firewall detected the process running, along with its process ID. You can press **n** again to check for more entries for this service.

The last PID reported in **md_info.log** (**5613** for **authd** in this example) will match the currently running process as displayed using the **show system resources** command.

Did what you write down in step 6 match your PID in step 20?

21. Press **q** to return to the command prompt.

7.3 Change the Debug Level for a Service

You rarely should need to change the debug log level for a service on the firewall. You generally should change the log level for a service only if directed to do so by a current Knowledge Base article or by a technical support engineer as part of an open case.

WARNING: On a firewall that is in production, if you change the log level of a service to a higher logging level than the default, you should change the log level back to the default as soon you have logged the activity required.

In the following steps you will observe a baseline level of log activity. Then you will compare this baseline to elevated, debug-level, log activity. First, you will monitor the log for the Device Server (**devsrvr**) service.

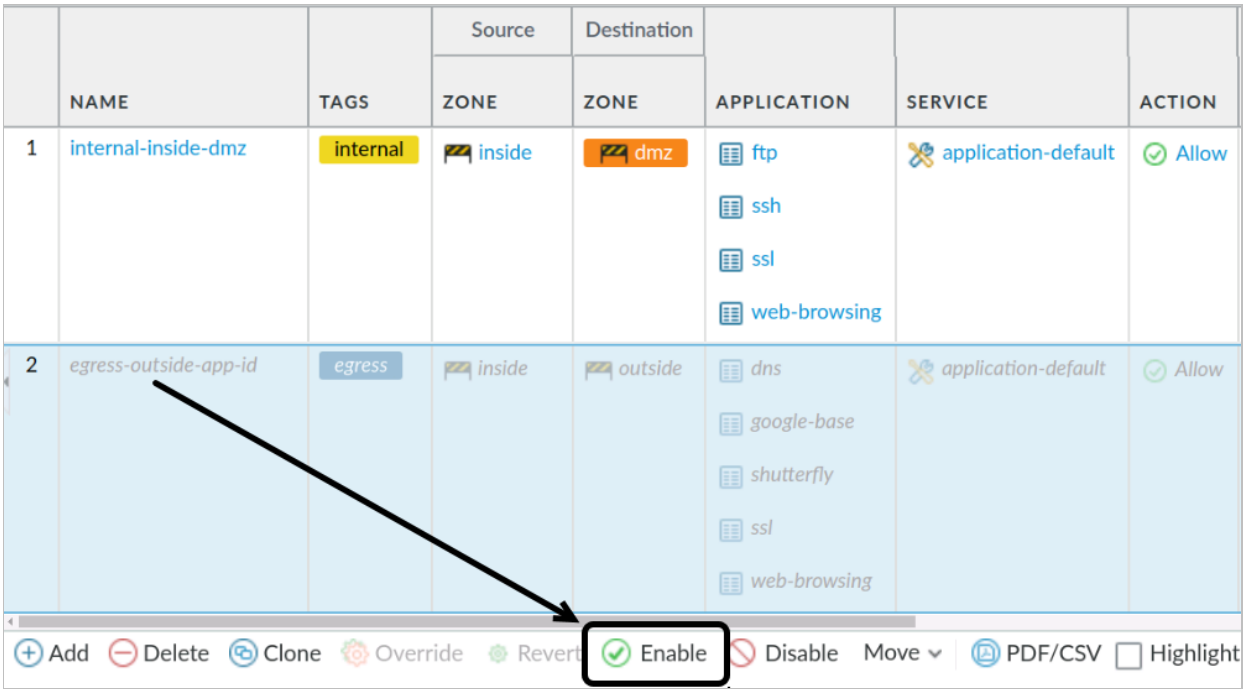
The Device Server handles, among other things, the transfer of configuration information to the data plane. Thus, you can perform a **Commit** action while monitoring the **devsrv.log** file to see real-time log activity.

22. Type:
    tail follow yes mp-log devsrv.log and press **Enter**.
    This command prints new log entries to the screen in near real time for as long as the command is left running:

```
admin@firewall-a> tail follow yes mp-log devsrv.log [Enter]
2023-04-25 16:34:10.868 +0000 content is installed or skip phase1 enabled
2023-04-25 16:34:10.869 +0000 Config commit for devsrvr only commit done
2023-04-25 16:36:09.864 +0000 update str:
[{
"action":"delete",
"rslt":["ggya.download", "coughstuffs.com", "advertiserexe.ru", "ngrpn.com",
"kutpaambalaj.com", "lfchanraomo.com",
[. . .]
```

    Note: The initial output from the command will be different from what is shown in the example.

23. In the firewall web interface, go to **Policies > Security**.
24. In the first column on the left, click inside the table cell labeled "**2**" to highlight the **egress-outside-app-id** rule:



25. With the second rule in the table highlighted, go to the bottom of the content-display area and click **Enable**.

    The typeface of the rule name should change from italics to standard Roman font.

26. **Commit** the configuration.
27. Return your focus to the **Remmina** window that is running the **`tail`** command.

    Watch the log output. Notice the various activities involved in the commit process:

```
[. . .]
2023-04-25 00:03:41.128 +0000 Config commit phase0 started
2023-04-25 00:03:41.536 +0000 pan_ha_is_sync_needed: needed=0, is_peer_up=0, sta[...]
2023-04-25 00:03:41.539 +0000 Config commit phase0 done
2023-04-25 00:03:42.924 +0000 Config commit phase1 started
2023-04-25 00:03:42.924 +0000 flags 0x10002, content 0x0, not devsrvr only, not [...]
[. . .]
2023-04-25 00:03:46.244 +0000 Config commit phase1 done
2023-04-25 00:03:46.264 +0000 flags 0x0, content 0x1, not devsrvr only, not cont[...]
2023-04-25 00:03:46.264 +0000 Config commit phase2 started
[. . .]
2023-04-25 00:03:47.329 +0000 Last committed config saved
[. . .]
2023-04-25 00:03:47.346 +0000 config is committed
2023-04-25 00:03:47.347 +0000 Config commit phase2 done
[. . .]
^C
admin@firewall-a>
```

28. After the **Commit** action in the web interface is complete, press **Ctrl+C** to return to the command prompt in the SSH connection.

In the following steps you will change the debug level of the Device Server log.

29. Type:
    `debug device-server show` and press **Enter**.
    This command displays the current debug level for the **`device-server`** service:

```
admin@firewall-a> debug device-server show [Enter]

debug level: info
  Features:
    config  : basic
```

    Notice that the current debug level is **`info`**. Many debug-log functions provide options for configuration *features* that allow you to target the collection of information based on a specific functional component or feature set. “Basic” typically is the default configuration feature and is designed to provide the debug information that you will need for most situations.

30. Type:
    `debug device-server on debug` and press **Enter**.
    This command changes the log level to “debug”:

```
admin@firewall-a> debug device-server on debug [Enter]

debug level: debug
  Features:
    config  : basic
```

31. Type:
    `tail follow yes mp-log devsrv.log` and press **Enter**.
32. Use the web interface to **disable** the **egress-outside-app-id** rule.
33. **Commit** the configuration change.

    Close the notification window from the previous commit, if it is still open. Take any other steps or substeps necessary to complete the task.

34. As the **Commit** is processing, return your focus to the **Remmina** window that is running the **`tail`** command and watch the log output.

    Look for lines tagged with "**`debug:`**".

    These tagged lines are the extra information that you have enabled by changing the log setting:

```
admin@firewall-a> tail follow yes mp-log devsrv.log [Enter]
[. . .]
2023-04-25 16:47:29.401 +0000 debug: pan_urlfiltering_get_application
  (pan_urlfiltering_handler.c:1126): application id (0)
2023-04-25 16:47:29.401 +0000 debug: pan_urlfiltering_get_application
  (pan_urlfiltering_handler.c:1145): app_id(0) application(undecided)         [. . .]
[. . .]
2023-04-25 16:48:09.714 +0000 Config commit phase1 started
2023-04-25 16:48:09.714 +0000 flags 0x10002, content 0x0, not devsrvr only, not
  content only
[. . .]
2023-04-25 16:48:09.717 +0000 debug: parse_plaintext_license_key(pan_license.c:1512):
  The bundle_id is 20528898
2023-04-25 16:48:10.287 +0000 debug: pan_tdb_do_file_2_version(pan_tdb_comp.c:109):
  version = 0x2e3109c
2023-04-25 16:48:10.298 +0000 debug: pan_tdb_do_load_serialize(pan_tdb_ser.c:498):
  pan_tdb_do_load_serialize: pan_regex_load_aho, len = 0x11ed3a
[. . .]
```

Note: Your output will differ somewhat from what is shown here.

35. Press **Ctrl+C** to exit the tail program and return to the command prompt:

```
[. . .]
2023-04-25 16:53:15.270 +0000 pan_controller_proc::pan_ctrl_shutdown_release(): Free
  held shutdown permission
2023-04-25 16:53:15.270 +0000 pan_controller_proc::pan_ctrl_shutdown_release():
  Current shutdown permission status 0
^C
admin@firewall-a>
```

Important: After you generate the log data that you need, normally the next thing you should do is change the debug level back to its default level, which in this case would be **info**. However, for the purposes of this lab only, leave the log level set to "debug" so that you can see the effect that restarting a process has on debug levels.

A restart of a process for which you have changed debug log level will reset the service's log level to the default. A restart of the firewall will reset all service log levels to their default values.

7.4 Restart a Service

WARNING: The following lab activity directs you to restart services to illustrate the skills and concepts that typically will be required only in advanced troubleshooting scenarios. In practice, do not restart data-plane services without guidance from a technical support engineer or a Knowledge Base article that is directly relevant to your situation.

In most cases, if you restart the Device Server, Management Server, or another management-plane service, you will not cause any impact to data-plane traffic.

36. Type:
show system software status | match devsrvr and press **Enter**.
This command shows the status of the Device Server:

```
admin@firewall-a> show system software status | match devsrvr [Enter]
Process  devsrvr              running  (pid: 3325)
```

Make a note of the PID so you can compare it to the new PID after restarting the service.

After you run a command, the firewall will put it in the command line buffer so that, after you execute the next command, you can quickly rerun it by pressing the **Up Arrow** and then **Enter**.

If you are prepared to rerun the command *quickly*, you will be able to catch the change in running status that occurs after restarting a service.

37. Type:
debug software restart process device-server and press **Enter**.
This command restarts the Device Server service:

```
admin@firewall-a> debug software restart process device-server [Enter]

Process devsrvr was restarted by user admin
```

38. Press the **Up Arrow twice** and then press **Enter**.
This procedure reruns the **show system software status | match devsrvr** command. Note the status:

```
admin@firewall-a> show system software status | match devsrvr
Process  devsrvr              stopping  (pid: 3325) - User Stop
```

39. Wait about five seconds, then press the **Up Arrow** and then press **Enter**.
    This procedure reruns the **`show system software status | match devsrvr`** command.
    Repeat this process until you see the service running with a new process ID (**`pid`**) number:

```
admin@firewall-a> show system software status | match devsrvr
Process  devsrvr              running  (pid: 10060)
```

40. Type:
    `debug device-server show` and press **Enter**.
    This command displays the current debug level for the **`device-server`** service:

```
admin@firewall-a> debug device-server show

debug level: info
  Features:
    config  : basic
```

    Notice that, after the service is restarted, the current debug level has been reset to **`info`**. Note that restarting the service is *not* the method you should use to revert a custom log-level setting.
    In the web interface, close the **Commit** results dialog, if you have not already done so.

7.5 Restart a Service and Monitor a Data-Plane Session

In the following steps, you will:

- Establish an SSH connection,
- Restart the Management Server service (which will disconnect you from the CLI and web interface),
- Reconnect to the web interface, and
- Verify that the SSH session still exists.

41. On the **Desktop**, double-click **Remmina** and then double-click the **Server-Extranet**.
    This process will establish an SSH connection from the client workstation to the Extranet Server.

42. In the **Remmina CLI** of **Firewall-A**, type:
    `show session all filter application ssh` and press **Enter**.
    This command shows active sessions that match the application ID "**`ssh`**":

```
admin@firewall-a> show session all filter application ssh [Enter]

--------------------------------------------------------------------------------
ID    Application  State Type Flag Src[Sport]/Zone/Proto (translated IP[Port])
Vsys                                  Dst[Dport]/Zone (translated IP[Port])
```

```
--------------------------------------------------------------------------------
3252  ssh          ACTIVE FLOW     192.168.1.20[50588]/inside/6  (192.168.1.20[50588])
vsys1                              192.168.50.10[22]/dmz  (192.168.50.10[22])
```

Write down the session ID number. Your number will be different from the one in the example.

43. Type:
    Type **`debug software restart process management-server`** and press **Enter**.
    This command will restart the Management Server process:

```
admin@firewall-a> debug software restart process management-server [Enter]

Process mgmtsrvr was restarted by user admin
2023-04-25 18:24:06.510 +0000 Error:  pan_read_full(comm_utils.c:107): srvr: fatal
recv error. sock=18 err=Connection reset by peer (104)
admin@firewall-a>
```

    On restart of the Management Server, the CLI itself will be disconnected from the Management Server process. Immediately after you execute the command to restart the service, you can, for example, attempt to execute a command such as **`show system software status`**, but typically you will be returned to the command prompt without the command having been run. Within 10 to 15 seconds, the SSH session will be terminated and you will be disconnected from the CLI.

44. Check the web interface.
    You may see a status message that indicates that the web interface also is disconnected.

45. Click **OK** to clear the error message if you received one.
    The Management Server service usually takes fewer than one to two minutes to restart. Depending on the firewall model and variations in the configuration of management services, a restart of the Management Server service may take longer. If you use the refresh button on the firewall web interface, you may get the Logged Out window for the firewall.

46. Wait one or two minutes and then attempt to reload the web interface and log in.
    After the firewall disconnects the CLI session, the **Remmina** window may close automatically. Before closing **Remmina**, it may display a notification that says, “Connection closed by remote host.”

47. Click **OK** if the notification is displayed.
    The Remmina window will close.

48. In **Remmina**, double-click the entry for **Firewall-A** to reconnect by SSH.

49. Type:
    `show system software status` and press **Enter**.

50. Ensure all processes are running.

7.6 Investigate the Event

Here you will verify that the mgmtsrvr process has been restarted on the Server-Extranet by using the firewall CLI and System Log.

51. In the firewall CLI, type:
    `show session all filter application ssh` and press **Enter**.

    This command shows active sessions that match the application ID **`ssh`**.

    You should see the same session information that you saw earlier in the activity, including the session ID, session state (ACTIVE), and port numbers.

52. Select the Remmina connection to the **Server-Extranet**.
53. Verify that the **Server-Extranet** is not displaying any notifications about having been disconnected.
54. Close the Remmina connection to the **Server-Extranet**.
55. In the firewall CLI, run the **`show session all filter application ssh`** command.

    Tip: Press the **Up Arrow** and then **Enter**.

    About 15 to 30 seconds may elapse before the session is removed from the session monitor. If you rerun the command until you see "**No Active Sessions,**" you will demonstrate that the session that you were monitoring was, in fact, the session for the Server-Extranet connection because the end of the session will correlate to the time of your closure of the Server-Extranet, plus the TIME_WAIT period for the session.

56. Type:
    `grep mp-log md_info.log pattern mgmtsrvr` and press **Enter**.

    This command displays all lines in the **`md_info.log`** file with a match to "**mgmtsrvr**":

```
admin@firewall-a> grep mp-log md_info.log pattern mgmtsrvr [Enter]
[. . .]
2023-04-25 18:24:02.498 +0000 INFO: mgmtsrvr: received user restart
2023-04-25 18:24:02.498 +0000 INFO: mgmtsrvr: User restart reason - triggered by CLI
2023-04-25 18:24:02.498 +0000 INFO: mgmtsrvr: received user stop
2023-04-25 18:24:42.502 +0000 INFO: mgmtsrvr: hasn't exited properly, sending SIGKILL
2023-04-25 18:24:43.140 +0000 INFO: mgmtsrvr: exited, Core: False, Exit signal: [...]
2023-04-25 18:24:44.147 +0000 INFO: mgmtsrvr: process running with pid 18271
```

    Note: Your output will differ somewhat from what is shown here.

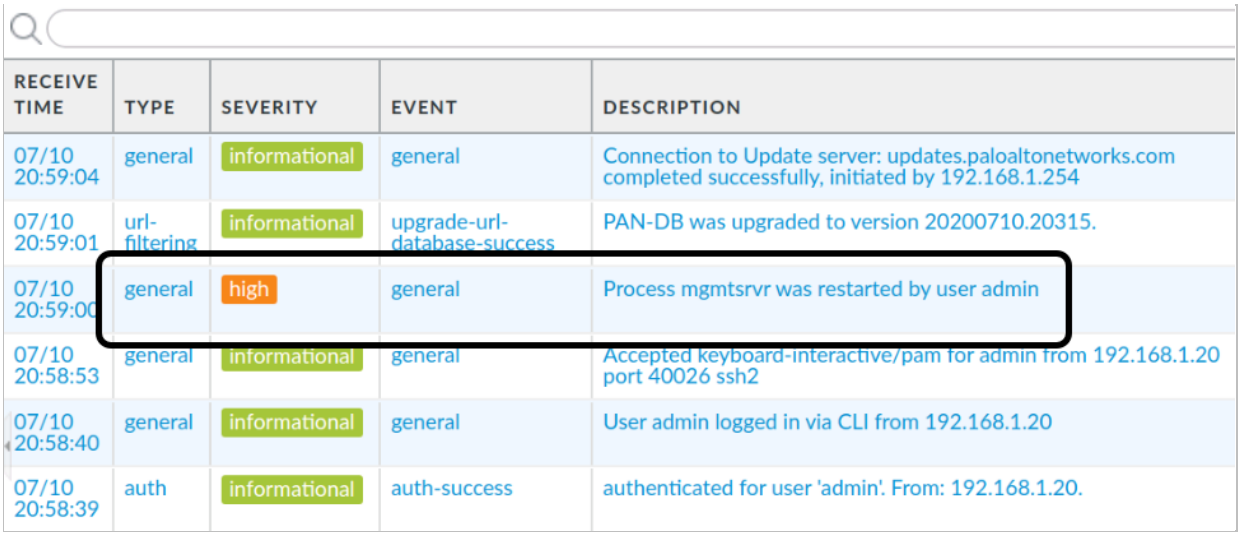
    This is useful information. However, to find it, you would have to know to look in the **`md_info.log`**. If you examine this log, you will not find which user of the CLI issued the restart. Instead, examine the standard System log of the firewall.

57. Type:
`show log system severity greater-than-or-equal high direction equal backward | match mgmtsrvr` and press **Enter**.
This command filters System log data for severity and sorts by newest entries first:

```
admin@fw-a> show log system severity greater-than-or-equal high direction equal backward |
match mgmtsrvr

2023-04-25 18:24:02 high  general  general 0  Process mgmtsrvr was restarted by user admin
```

This output gives us both the time of the event and an associated user. Your next task will be to look for this information in the web interface.

58. Log in to the firewall web interface and go to **Monitor > Logs > System**.
59. Find the log entry that correlates to the entry you located in the CLI:



RECEIVE TIME	TYPE	SEVERITY	EVENT	DESCRIPTION
07/10 20:59:04	general	informational	general	Connection to Update server: updates.paloaltonetworks.com completed successfully, initiated by 192.168.1.254
07/10 20:59:01	url-filtering	informational	upgrade-url-database-success	PAN-DB was upgraded to version 20200710.20315.
07/10 20:59:00	general	high	general	Process mgmtsrvr was restarted by user admin
07/10 20:58:53	general	informational	general	Accepted keyboard-interactive/pam for admin from 192.168.1.20 port 40026 ssh2
07/10 20:58:40	general	informational	general	User admin logged in via CLI from 192.168.1.20
07/10 20:58:39	auth	informational	auth-success	authenticated for user 'admin'. From: 192.168.1.20.

Note: Your web interface in the lab may show several additional columns that are removed in the example to display the relevant parameters in print more clearly.

Imagine that you are investigating a case involving a “spontaneous” disconnect or other change in system status on the firewall. When the operational state of the firewall changes suddenly, the System log is the first place you should look for indicators of the cause. If the change was the result of a change in system settings, you should be able to find that action in the System logs.

Based on what you see reviewing this event, what might you filter for to check for changes to the system that could help you in other situations in your work environment?

7.7 Clean Up Your Lab Environment

61. Close any open testing browser tabs and windows. Clear any filters in logs.

    Leave open the browser connection to the web interface of the firewall.

62. If you have an open CLI connection to the firewall, leave the connection open for the next lab.

Stop. This is the end of the lab.

8. Lab: SSL Decryption

Lab Objectives

- Use automation tools that are built into the firewall to toggle SSL decryption on or off without the need to change the running configuration of the firewall
- Demonstrate the use of a Dynamic Address Group (DAG) in combination with tags and log processing to change policy rule matches

Lab Scenario

To troubleshoot an SSL decryption issue with a specific site or internal client, you can determine whether the problem is with the site itself or with something in the decryption process by loading the target site without SSL interception of the connection by the firewall. However, reconfiguring and committing the firewall policies that would be required to change how the firewall processes such traffic typically would involve a formal change-control process. There would most likely also be a limited maintenance window for the implementation of the changes.

This lab activity will guide you through the following tasks:

- Create a tag and use this tag to define the membership of a DAG.
- Create a Decryption policy rule that matches to addresses in the DAG and applies the "no-decrypt" action.
- Create two custom Vulnerability signatures to detect custom keyword triggers that you can use as "commands" embedded within URL requests to generate Threat logs with a severity level of *informational*.
- Configure a Log Forwarding profile that triggers an action to add or remove the tag for the DAG whenever the firewall generates a Threat log that matches to a corresponding custom Vulnerability signature.
- Apply the Log Forwarding profile to a Security policy rule.

At the end of the lab activity, you will be able to cause the firewall to change the Decryption policy rule that the firewall applies to the client without the need to make any further changes to the firewall configuration.

8.1 Apply a Baseline Configuration to the Firewall

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-08.xml**

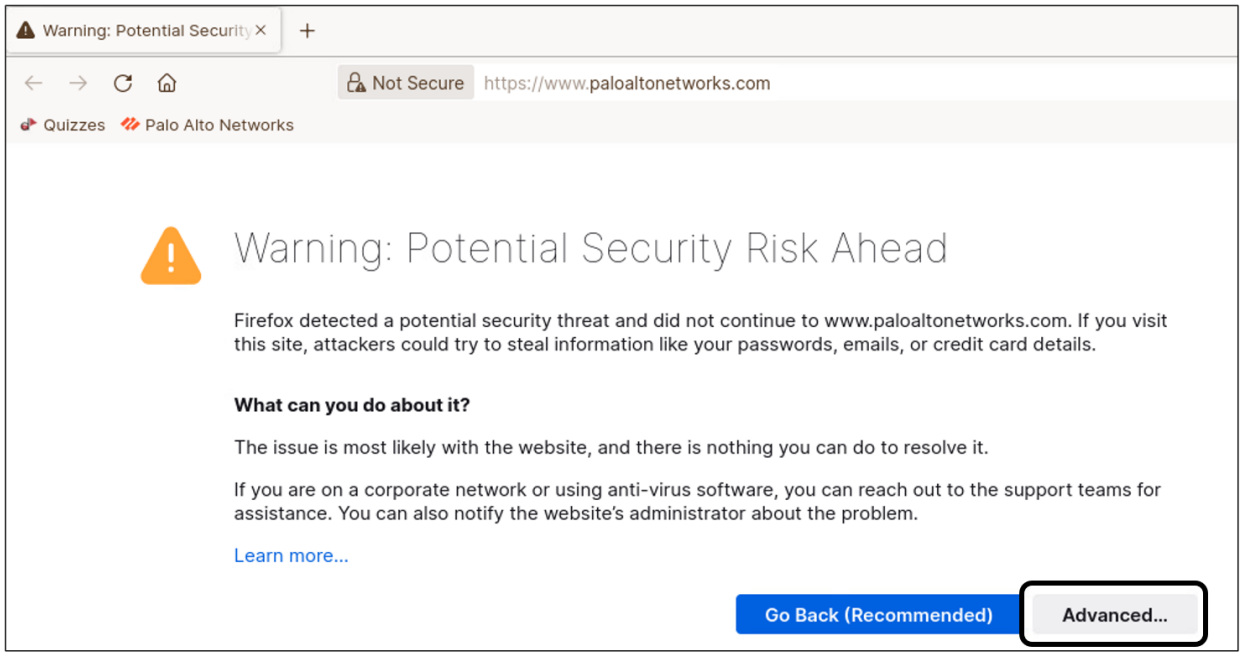
   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.

2. After the load task is complete, use the web interface to **Commit** the configuration.

8.2 Verify the Functionality of SSL Decryption

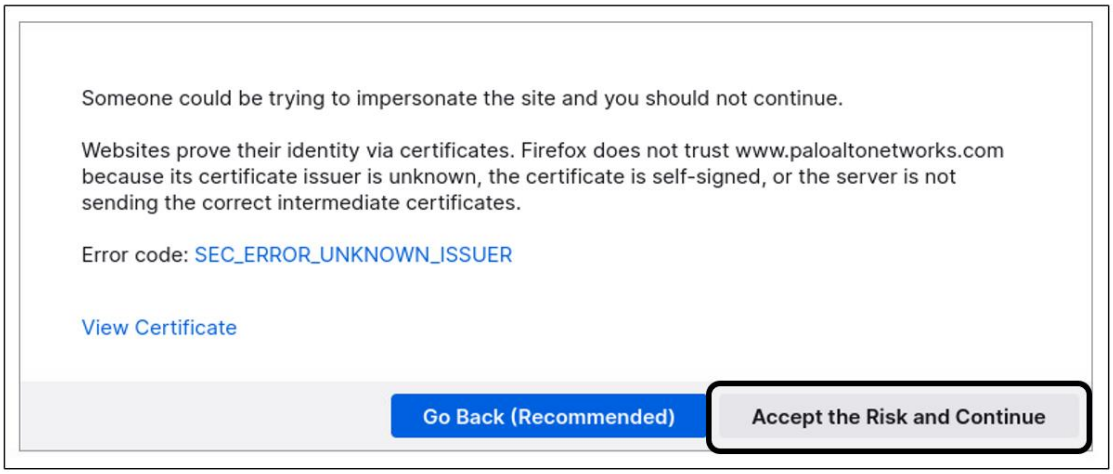
For this lab, use Firefox for testing and use Chromium to configure the firewall. The instructions for this lab are written to illustrate working with certificates specifically in Firefox; however, the overall process and principles for working with other browsers are similar.

3. Launch **Firefox** and go to:
   `www.paloaltonetworks.com`



The browser will report a problem with the security certificate. The firewall has intercepted the session for decryption, but the browser does not trust the certificate that the firewall has sent.

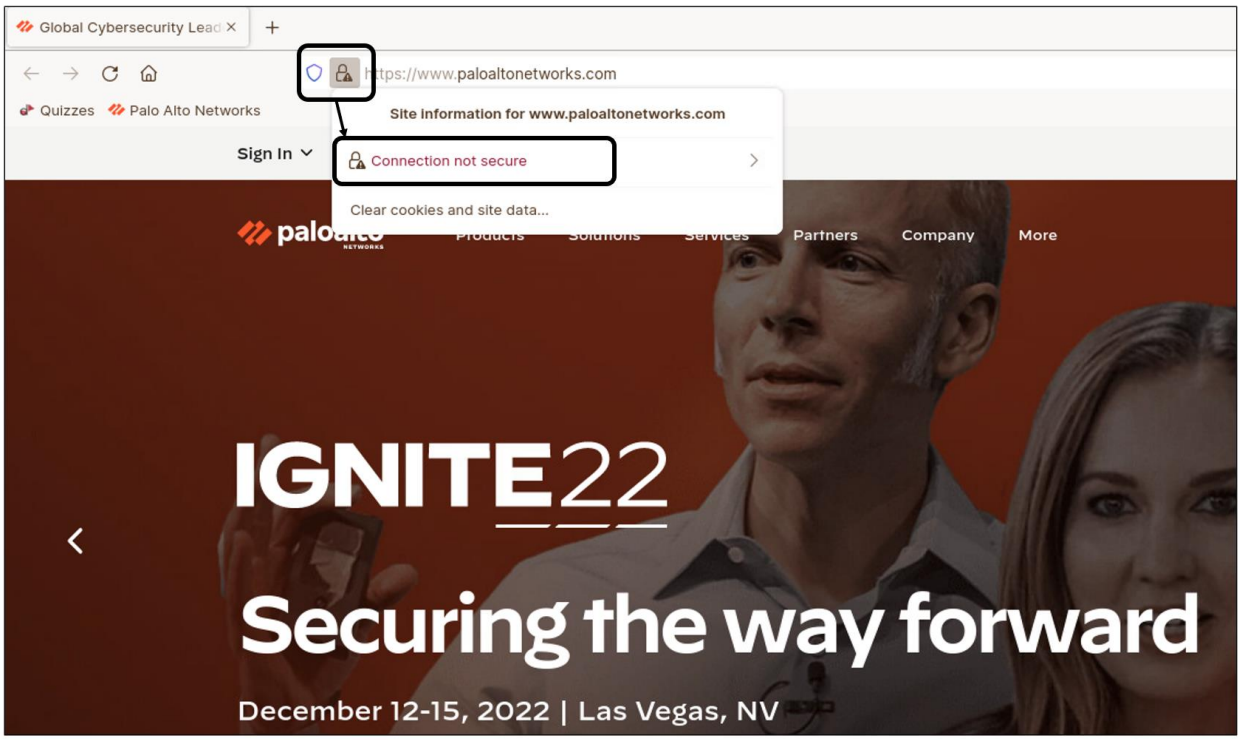
4. Click **Advanced**.
5. Click **Accept the Risk and Continue**.



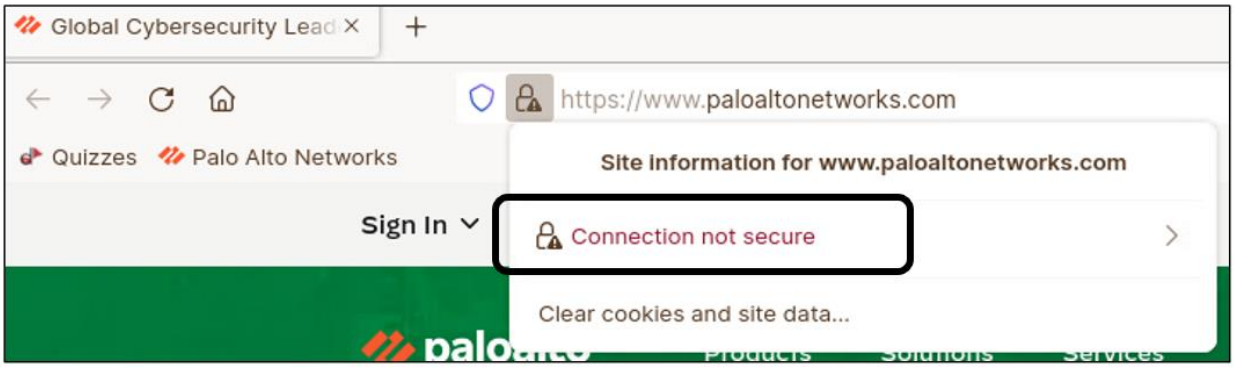
The browser should connect to the site, render the target page, and display in the address bar a notification of a certificate error:

Note: The graphics and other page content displayed may be different than shown here.

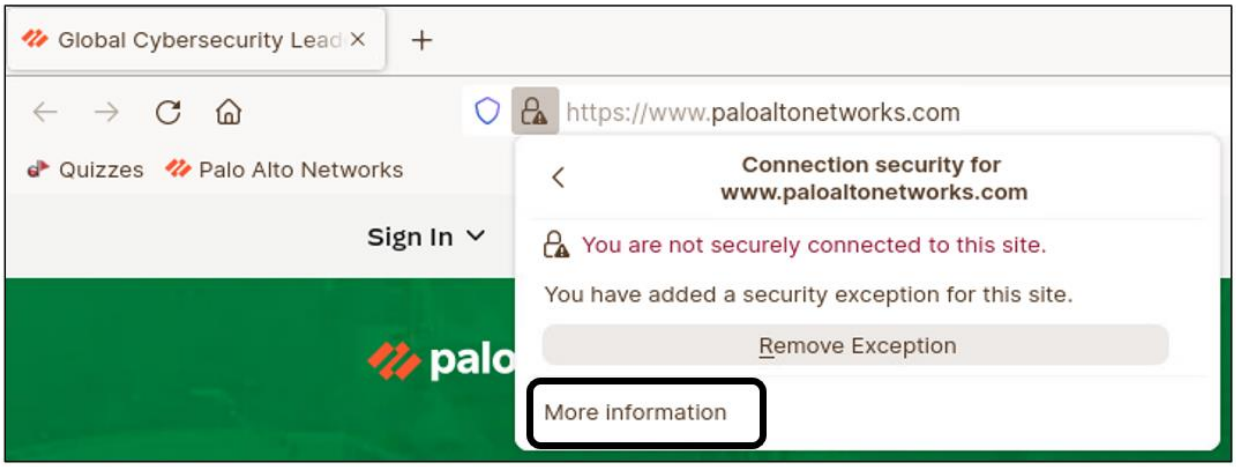
6. Click the padlock next to the URI and note the message **Connection not secure**.



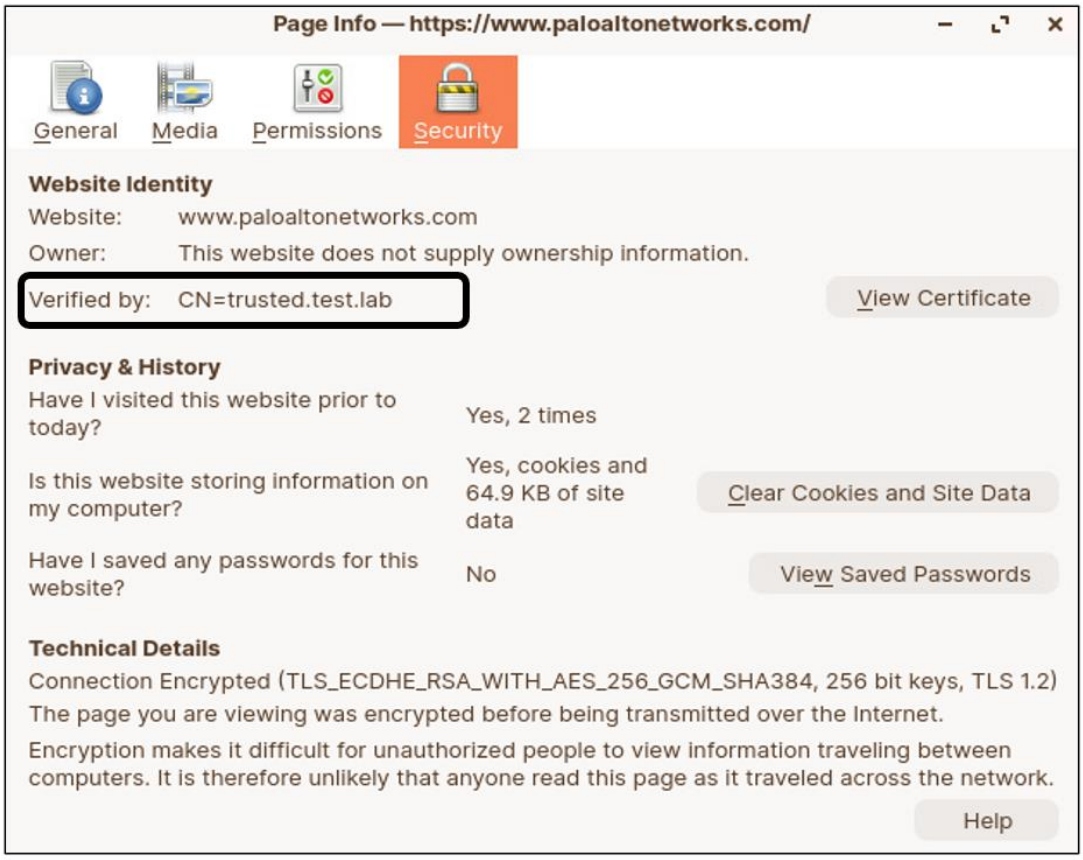
7. Click the arrow to the right of **Connection not secure**.



8. Select **More Information**.



9. Verify that the certificate is issued by "trusted.test.lab," which is the name used by the forward trust certificate that is configured for the firewall:



10. Close the **Certificate** window.
11. Close **Firefox**.

Note: For the purpose of the lab, you do not need to install on the browser the forward trust certificate from the firewall. If the certificates are left uninstalled, you will be able to use the certificate warning that Firefox displays as an initial indication of whether or not the firewall is intercepting a connection for SSL decryption.

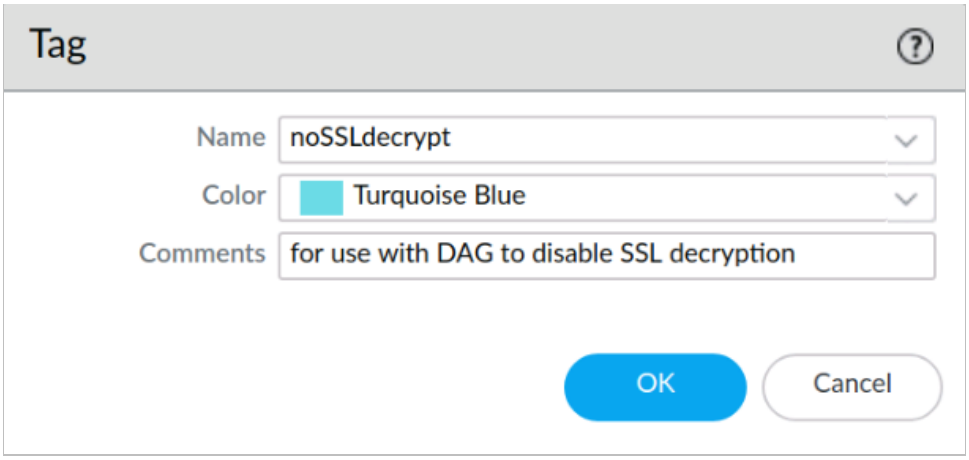
8.3 Create a Tag and a Dynamic Address Group

You will create a tag that will be used to indicate that any IP addresses that are associated with the tag are *not* to be subject to SSL decryption.

12. In the web interface of the firewall, go to **Objects > Tags**.
13. Click **Add** at the bottom of the display area.
14. Configure the tag using the following specifications:

Parameter	Value
Name	**noSSLdecrypt**
Color	**Turquoise Blue**

Parameter	Value
Comments	`for use with DAG to disable SSL decryption`



15. Click **OK**.

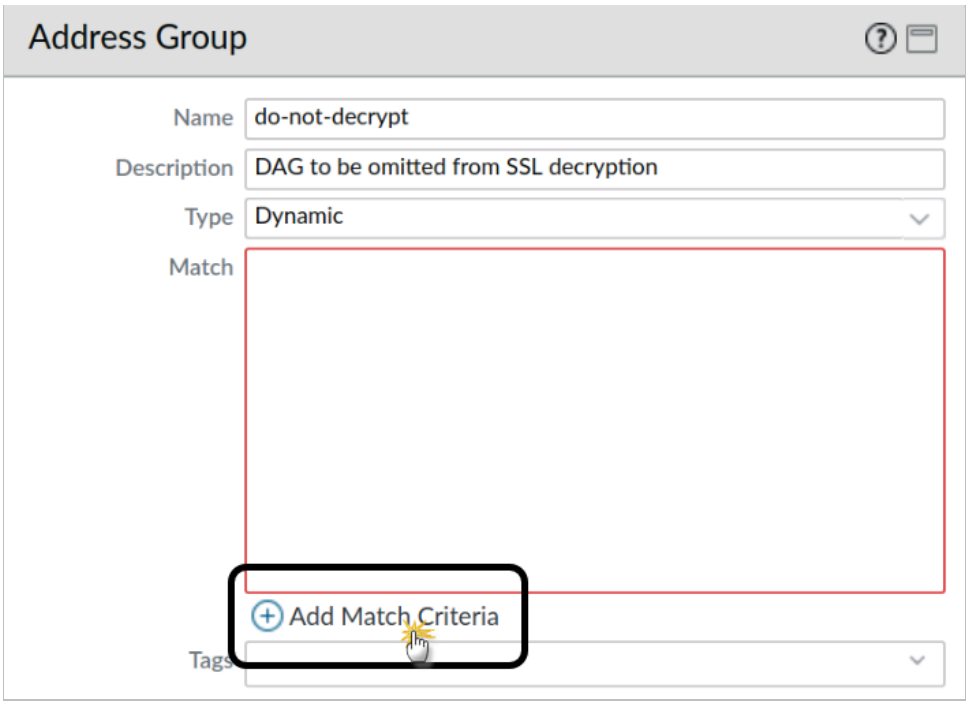
    Your list of tags should look similar to the following:

	NAME	LOCATION	COLOR	COMMENTS
☐	Sanctioned	Predefined	Olive	
☐	empty	Predefined		
☐	danger		Purple	
☐	egress		Blue	
☐	dmz		Orange	
☐	internal		Yellow	
☐	noSSLdecrypt		Turquoise Blue	for use with DAG to disable SSL decryption

16. Go to **Objects > Address Groups**.
17. Click **Add** at the bottom of the display area.
18. Configure the first three parameters of the new **Address Group** as follows:

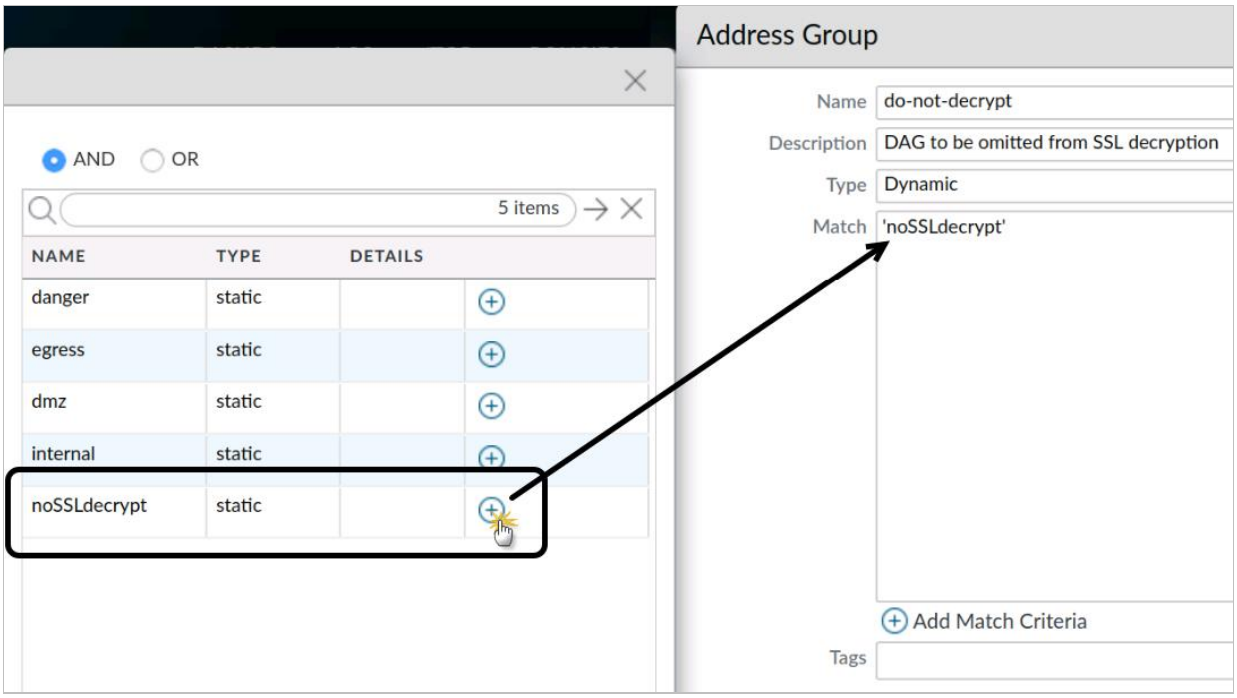
Parameter	Value
Name	`do-not-decrypt`
Description	`DAG to be omitted from SSL decryption`
Type	**Dynamic**

19. Click **Add Match Criteria**:



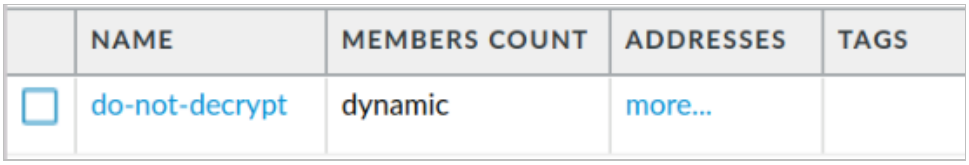
Click the **plus sign (+)** on the row that corresponds to the **noSSLdecrypt** tag.

The **Match** parameter **noSSLdecrypt** is added to the Address Group configuration:



20. Click **OK**.

The **do-not-decrypt** address group is added to the **Address Groups** list:



	NAME	MEMBERS COUNT	ADDRESSES	TAGS
☐	do-not-decrypt	dynamic	more...	

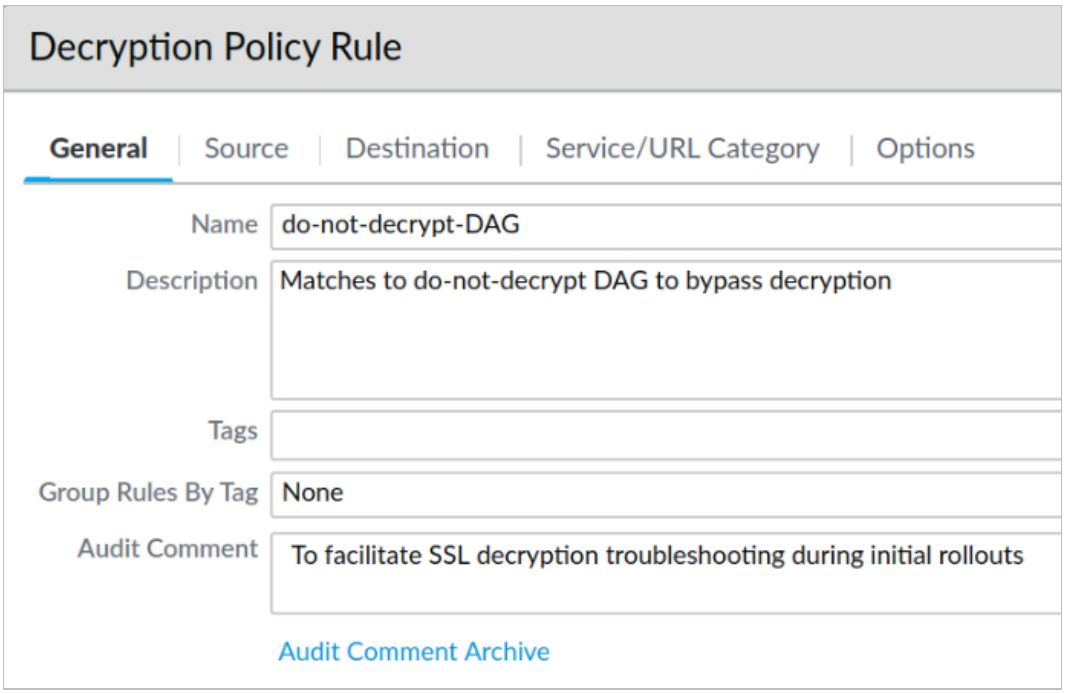
8.4 Create a Decryption Policy Rule

The current firewall configuration includes a Decryption policy rule that decrypts SSL connections from *any* internal source IP address to a short list of URLs defined by a custom URL Category list object named “test-decryption.” The current Decryption policy also includes a rule that excludes the management interface of the firewall from SSL decryption.

You need to add a Decryption policy rule that excludes from decryption all sessions with a source IP address that is a member of the DAG “do-not-decrypt.”

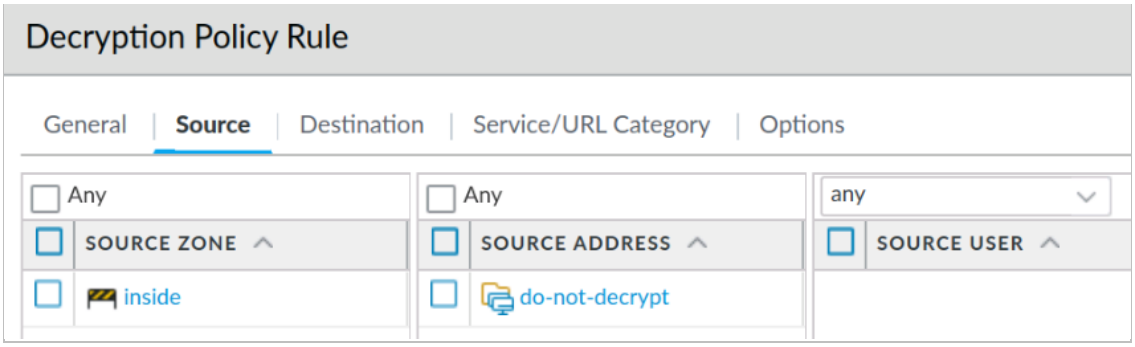
21. Go to **Policies > Decryption**.
22. Click **Add** at the bottom of the display area.
23. Configure the **General** tab of the new **Decryption Policy Rule** using the following specifications:

Parameter	Value
Name	**do-not-decrypt-DAG**
Description	**Matches to do-not-decrypt DAG to bypass decryption**
Tags	[blank]
Group Rules By Tag	**None**
Audit Comment	**To facilitate SSL decryption troubleshooting during initial rollouts**



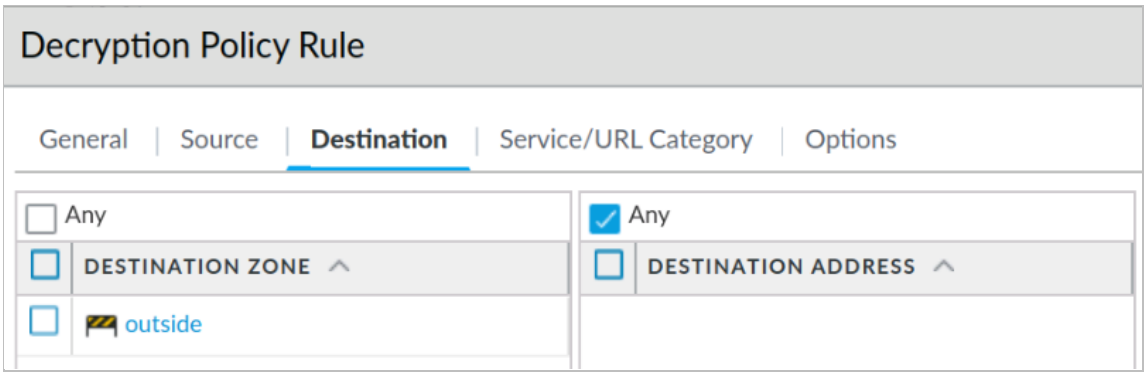
24. Configure the **Source** tab using the following specifications:

Parameter	Value
Source Zone	**inside**
Source Address	**do-not-decrypt**
Source User	[leave] **any**



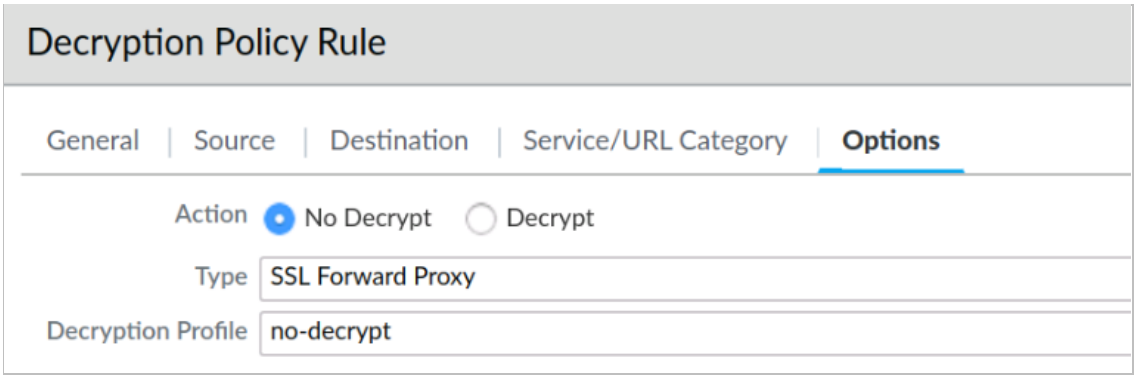
25. Configure the **Destination** tab using the following specifications:

Parameter	Value
Destination Zone	**outside**
Destination Address	[leave] **any**



26. Do not configure the **Service/URL Category** tab. Accept the default settings for "any" service and "any" URL category.
27. Configure the **Options** tab using the following specifications:

Parameter	Value
Action	**No Decrypt**
Type	**SSL Forward Proxy**
Decryption Profile	**no-decrypt**



28. Click **OK**.
29. Highlight the "do-not-decrypt-DAG" rule.
30. Click the **Move** button to move this rule to the first or second position in the list so that it is *above* the "decrypt-url-cat" rule.

Note: If you had selected the top rule before you clicked to **Add** the new rule, then you would not need to move the new rule. The new rule already would be in the second position in the list.

		Source		Destination				
	NAME	ZONE	ADDRESS	ZONE	URL CATEGORY	ACTION	TYPE	DECRYPTION PROFILE
1	no-decrypt-fw-mgmt-if	inside	192.168.1.254	outside	any	no-decrypt	ssl-forward-proxy	default
2	do-not-decrypt-DAG	inside	do-not-decrypt	outside	any	no-decrypt	ssl-forward-proxy	no-decrypt
3	decrypt-url-cat	inside	any	outside	test-decryption	decrypt	ssl-forward-proxy	default

You now have completed the configuration of a Decryption policy that can exclude a client from SSL decryption if the IP address of the client is a current member of a specific DAG.

8.5 Create Custom Vulnerability Signatures

Next, you need to build the infrastructure of objects and rules that will enable the firewall to detect the equivalent of external commands. You will send these "commands" to the firewall in the form of custom keyword triggers that you type into an HTTP request. The firewall will detect and respond to the presence of those keywords by tagging (or untagging) the source IP address of the requesting node. The tag defines membership within the DAG that your Decryption policy currently exempts from SSL decryption.

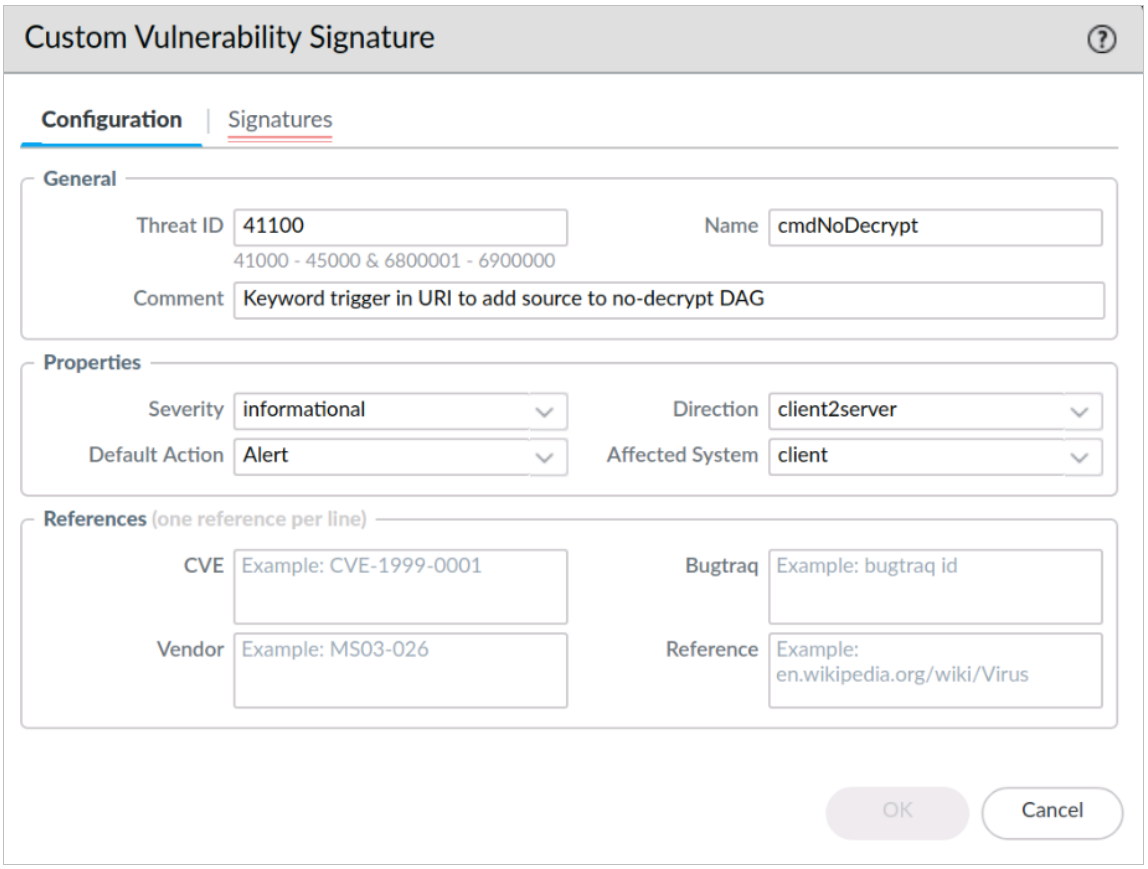
To enable the firewall to detect your command requests, you need to create two custom Vulnerability signatures:

- The first signature will match to the text string pattern "/cmdnodecrypt/" in the URI of an HTTP request.
- The second signature will match to the text string pattern "/cmdendnodecrypt/" in the URI of an HTTP request.

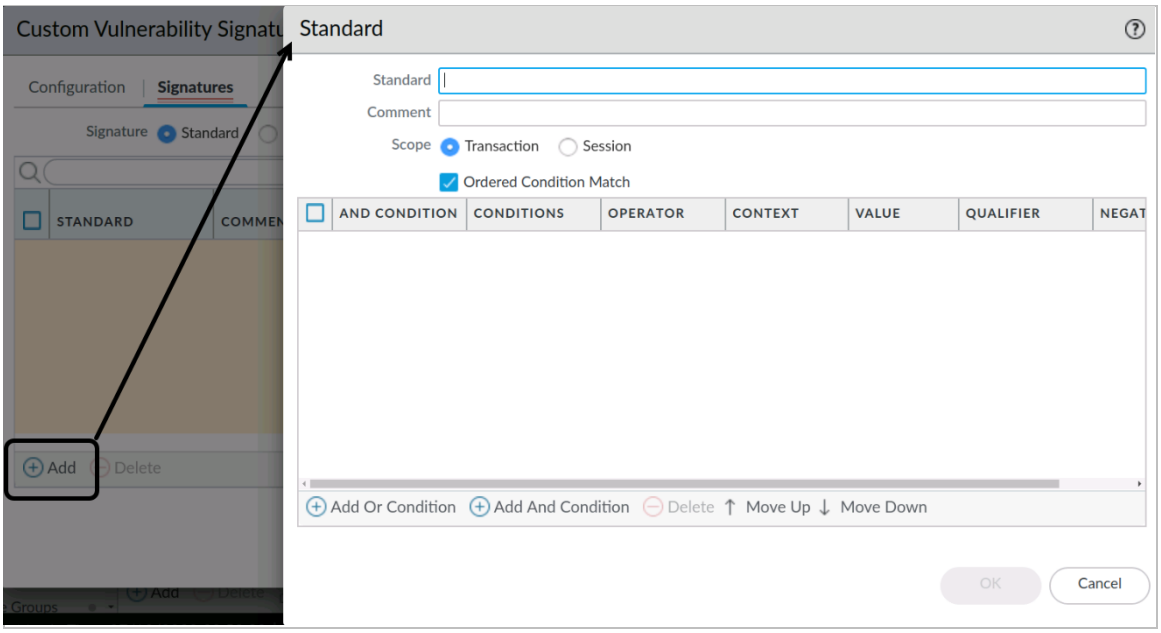
31. Go to **Objects > Custom Objects > Vulnerability**.
32. Click **Add** at the bottom of the display area.
33. Configure the parameters on the **Configuration** tab of the new **Custom Vulnerability Signature** using the following specifications:

Parameter	Value
Threat ID	**41100**
Name	**cmdNoDecrypt**
Comment	**Keyword trigger in URI to add source to no-decrypt DAG**
Severity*	**informational**
Direction	**client2server**
Default Action	**Alert**
Affected System	**client**
References: All	[blank]

***Note:** For use in production environments that include multiple network, security, incident response, and other groups, you would need to work with such groups to ensure that they know the meaning of the alerts that you will generate.

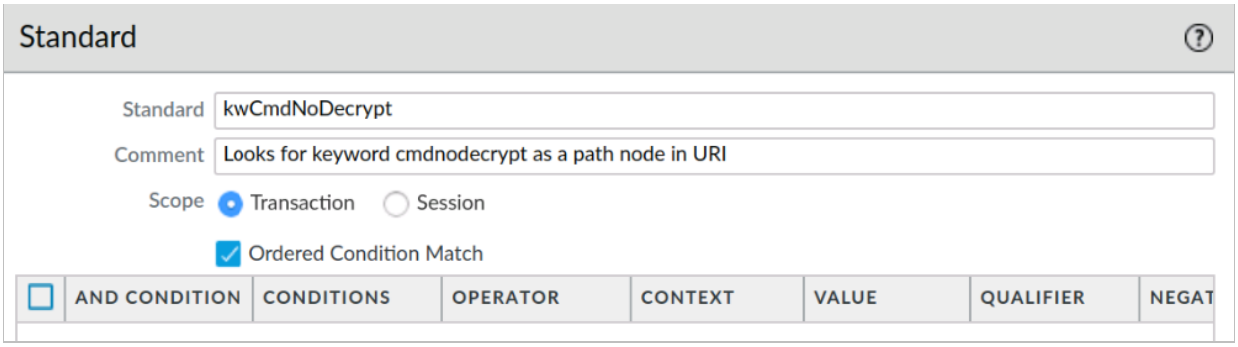


34. On the **Signatures** tab, accept the default **Standard** option, and click **Add**:

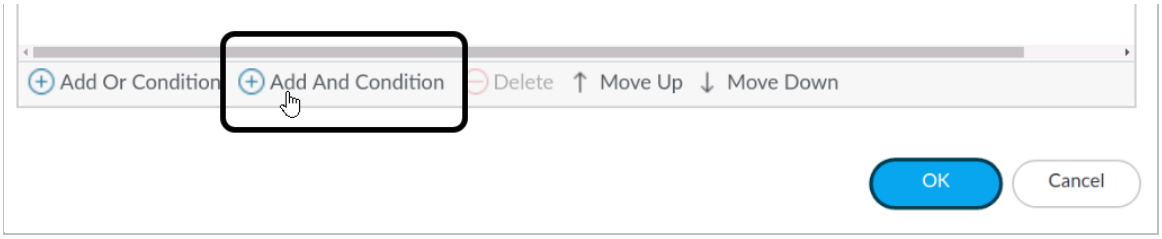


35. Configure the first four parameters of the new **Standard** signature as follows:

Parameter	Value
Standard	**kwCmdNoDecrypt**
Comment	**Looks for keyword cmdnodecrypt as a path node in URI**
Scope	**Transaction**
Ordered Condition Match	**Selected**



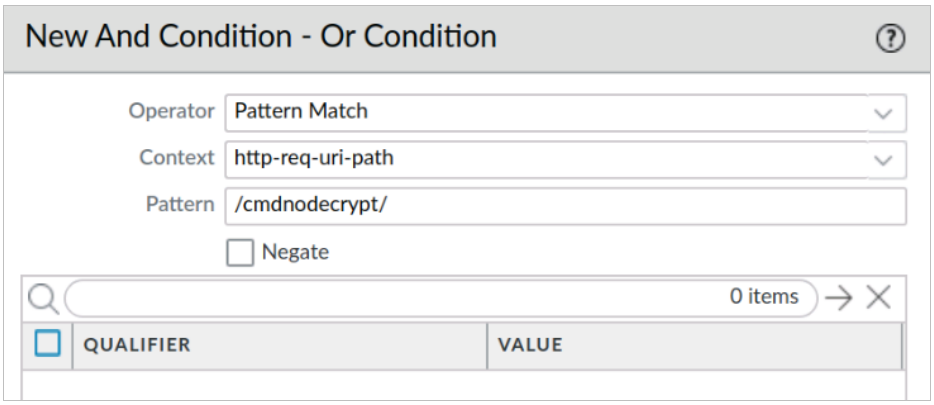
36. Click **Add And Condition** at the bottom of the content area of the **Standard** window:



Note: Be sure to select **Add And Condition** as shown in the preceding dialog box.

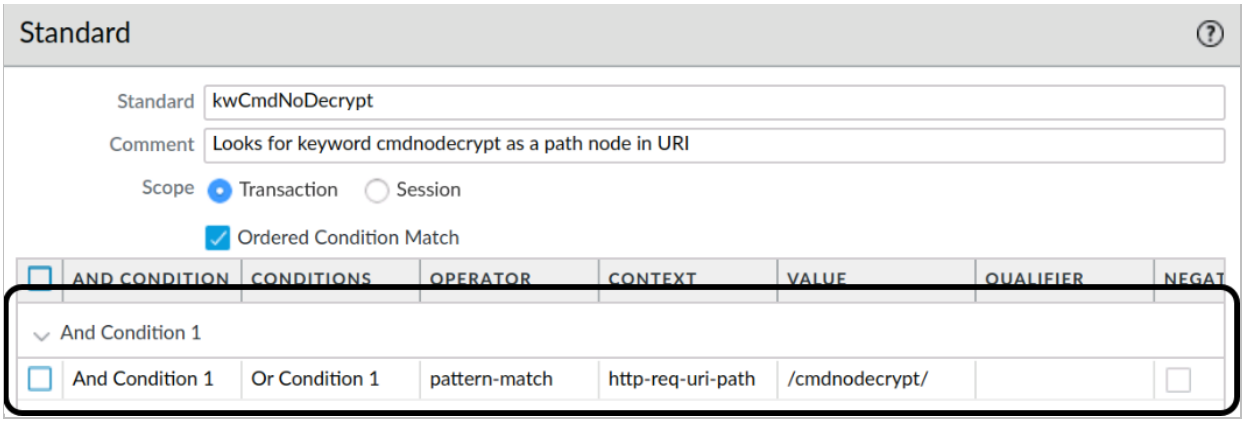
37. Configure the **New And Condition - Or Condition** using the following specifications:

Parameter	Value
Operator	**Pattern Match**
Context	**http-req-uri-path**
Pattern	**/cmdnodecrypt/**
Negate	not selected
Qualifier list	[blank]



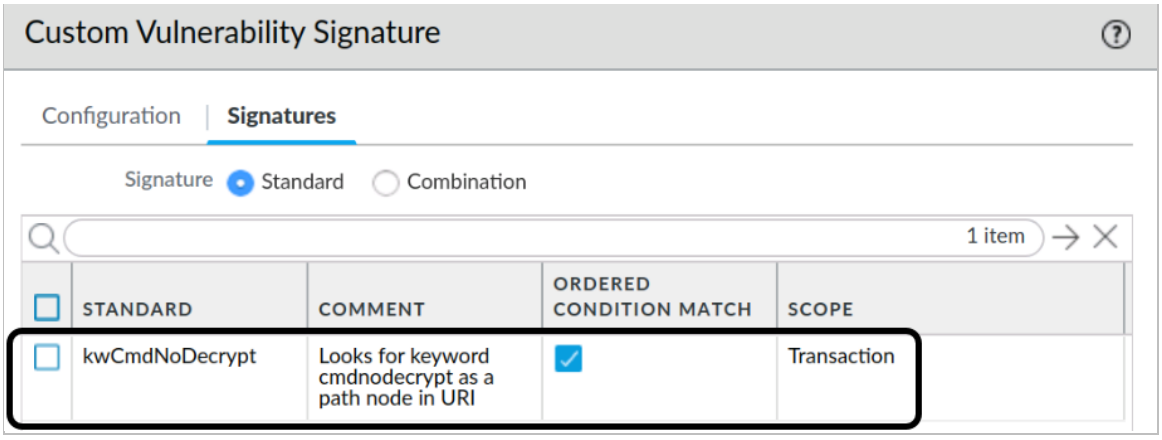
38. Click **OK**.

    The **Standard** page should be populated with the new pattern-match condition:



39. Click **OK**.

    The **Signatures** tab on the **Custom Vulnerability Signature** page should be populated with the new standard signature:



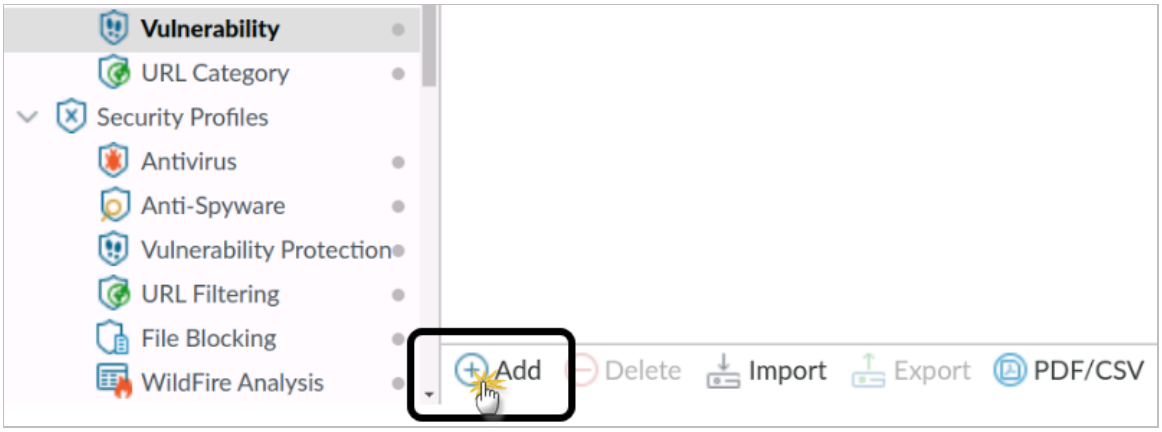
40. Click **OK**.

    The **Objects > Custom Objects > Vulnerability** page should be populated with the new threat ID, threat name, and other information:

NAME	THREAT ID	SEVERITY	DIRECTION	DEFAULT ACTION	AFFECTED SYSTEM	COMMENT
cmdNoDecrypt	41100	informational	client2server	alert	client	Keyword trigger in URI to add source to no-decrypt DAG

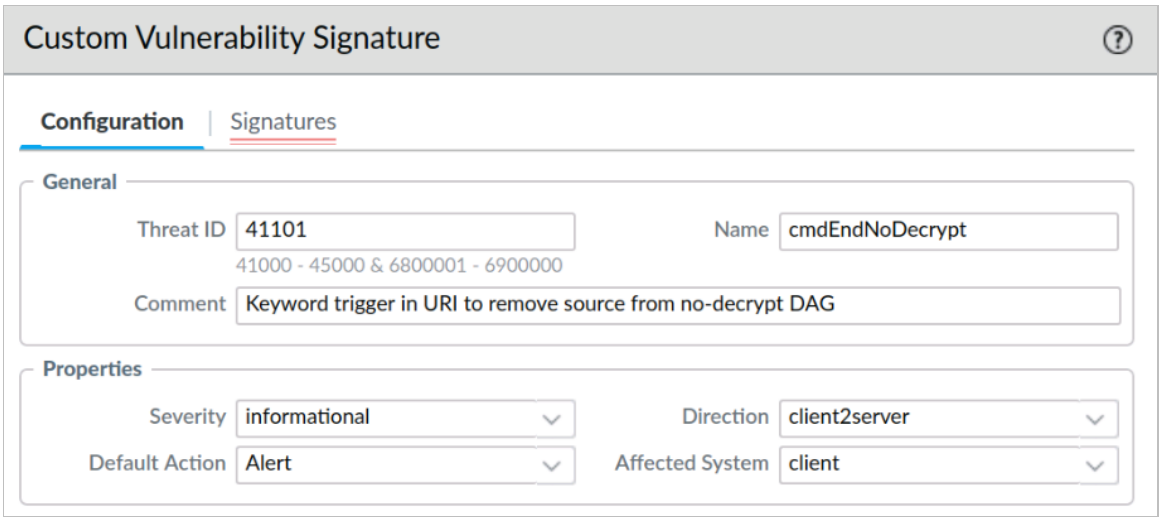
Next, you will create a second custom Vulnerability signature that detects the text string "/cmdendnodecrypt/" and that will be used to trigger the removal of source IP addresses from the "do-not-decrypt" DAG.

41. Click **Add** at the bottom of the display area:

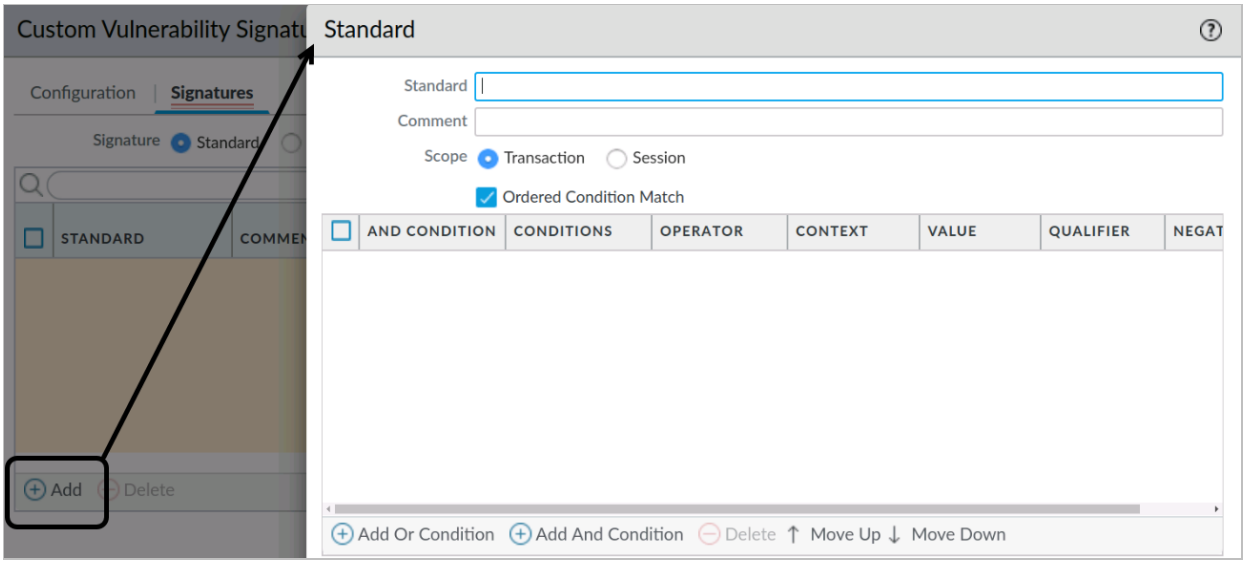


42. Configure the parameters on the **Configuration** tab of the new **Custom Vulnerability Signature** using the following specifications:

Parameter	Value
Threat ID	**41101**
Name	**cmdEndNoDecrypt**
Comment	**Keyword trigger in URI to remove source from no-decrypt DAG**
Severity	**informational**
Direction	**client2server**
Default Action	**Alert**
Affected System	**client**
References: All	[blank]

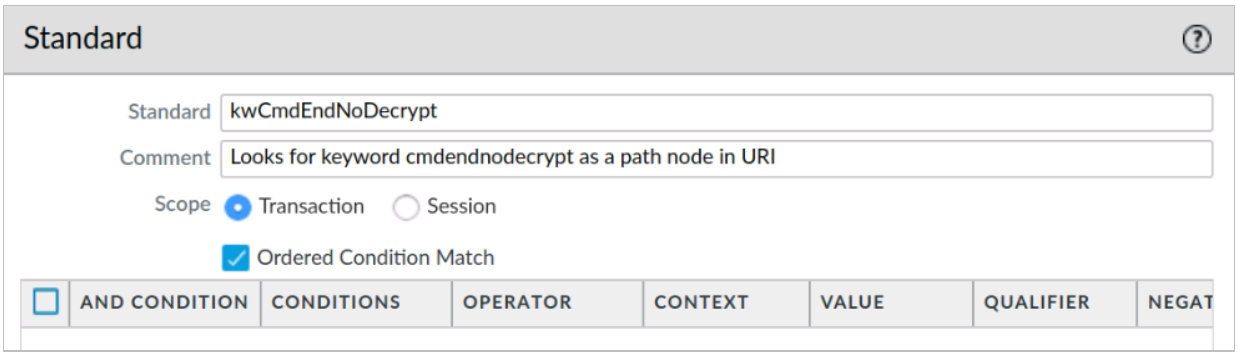


43. On the **Signatures** tab, accept the default **Standard** option, and click **Add**:

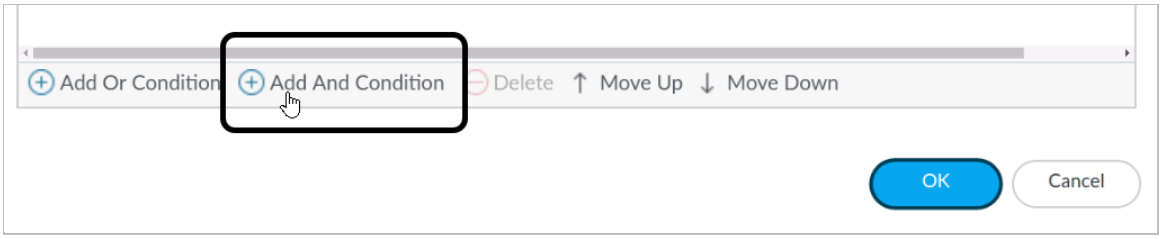


44. Configure the first four parameters of the new **Standard** signature as follows:

Parameter	Value
Standard	**`kwCmdEndNoDecrypt`**
Comment	**`Looks for keyword cmdendnodecrypt as a path node in URI`**
Scope	**Transaction**
Ordered Condition Match	**Selected**

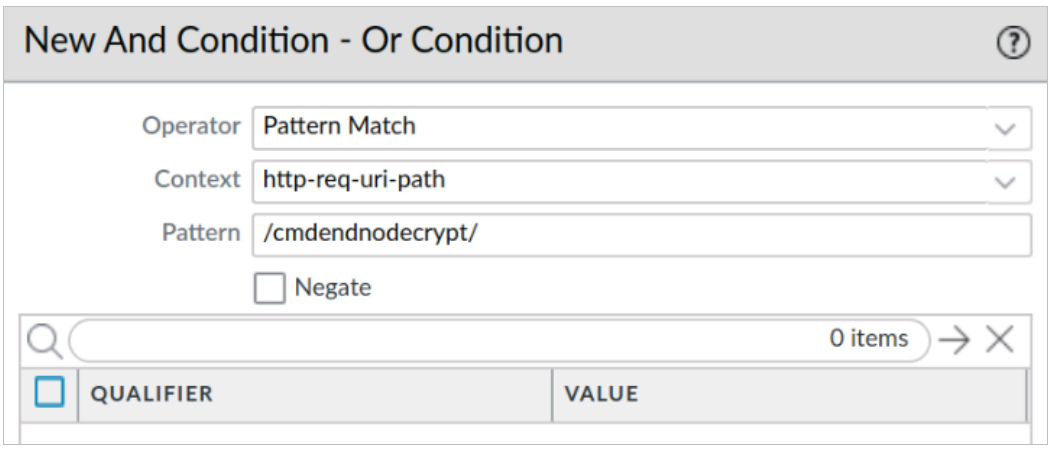


45. Click **Add And Condition** at the bottom of the content area of the **Standard** window:



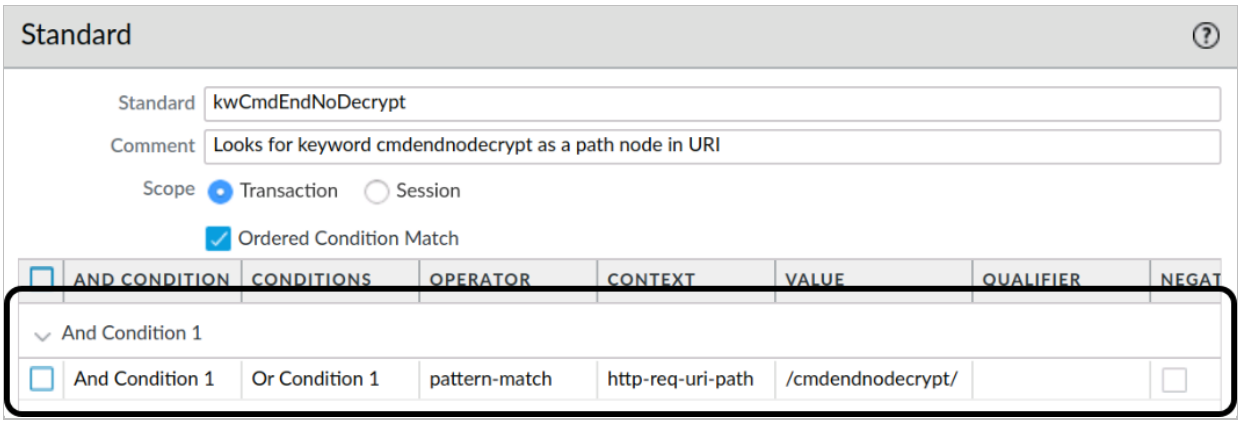
46. Configure the **New and Condition – Or Condition** using the following specifications:

Parameter	Value
Operator	**Pattern Match**
Context	**http-req-uri-path**
Pattern	**`/cmdendnodecrypt/`**
Negate	not selected
Qualifier list	[blank]



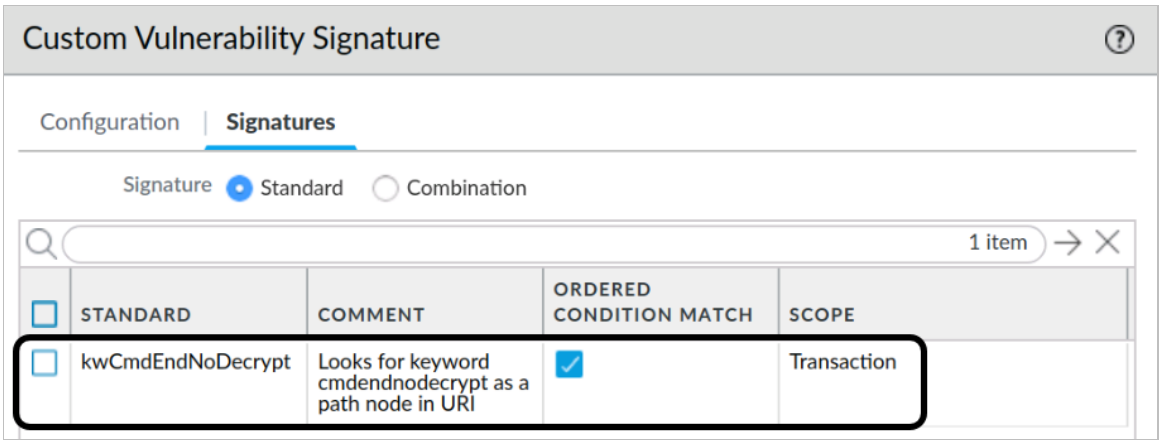
47. Click **OK**.

The **Standard** page should be populated with the new pattern-match condition:



48. Click **OK**.

    The **Signatures** tab on the **Custom Vulnerability Signature** page should be populated with the new standard signature:



49. Click **OK**.

    The **Objects > Custom Objects > Vulnerability** page should be populated with the new threat ID and custom Vulnerability signature:

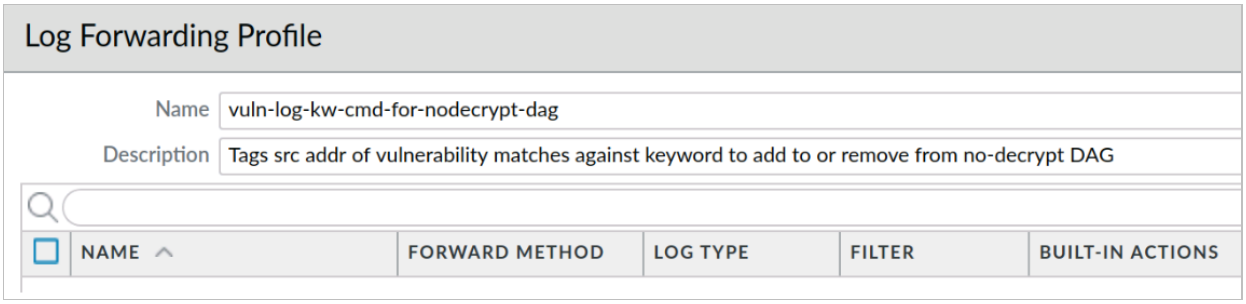
NAME	THREAT ID	SEVERITY	DIRECTION	DEFAULT ACTION	AFFECTED SYSTEM	COMMENT
cmdNoDecrypt	41100	informational	client2server	alert	client	Keyword trigger in URI to add source to no-decrypt DAG
cmdEndNoDecrypt	41101	informational	client2server	alert	client	Keyword trigger in URI to remove source from no-decrypt DAG

8.6 Configure a Log Forwarding Profile

With the use of custom Vulnerability signatures, you can generate Threat logs that contain unique threat ID numbers and names. You now can configure a Log Forwarding profile to watch for Threat logs that contain references to the threat IDs or the threat

names of the custom signatures and take an action, such as to add or to remove a tag from the source IP address of the matched log entry.

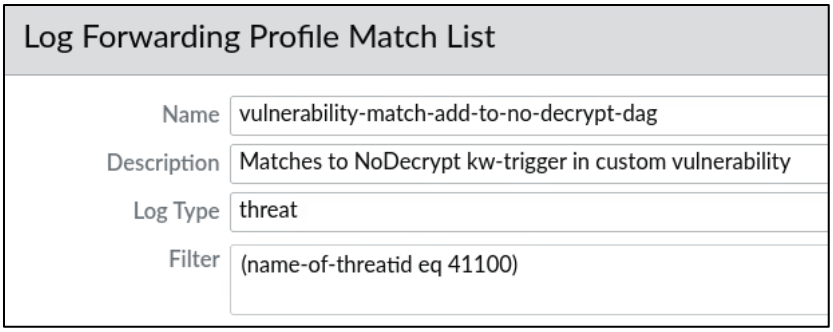
50. Go to **Objects** > **Log Forwarding**.
51. Click **Add** at the bottom of the display area:
52. For the **Name**, type:
    `vuln-log-kw-cmd-for-nodecrypt-dag`
53. For the **Description**, type:
    `Tags src addr of vulnerability matches against keyword to add to or remove from no-decrypt DAG`



54. Click **Add** at the bottom of the display area:

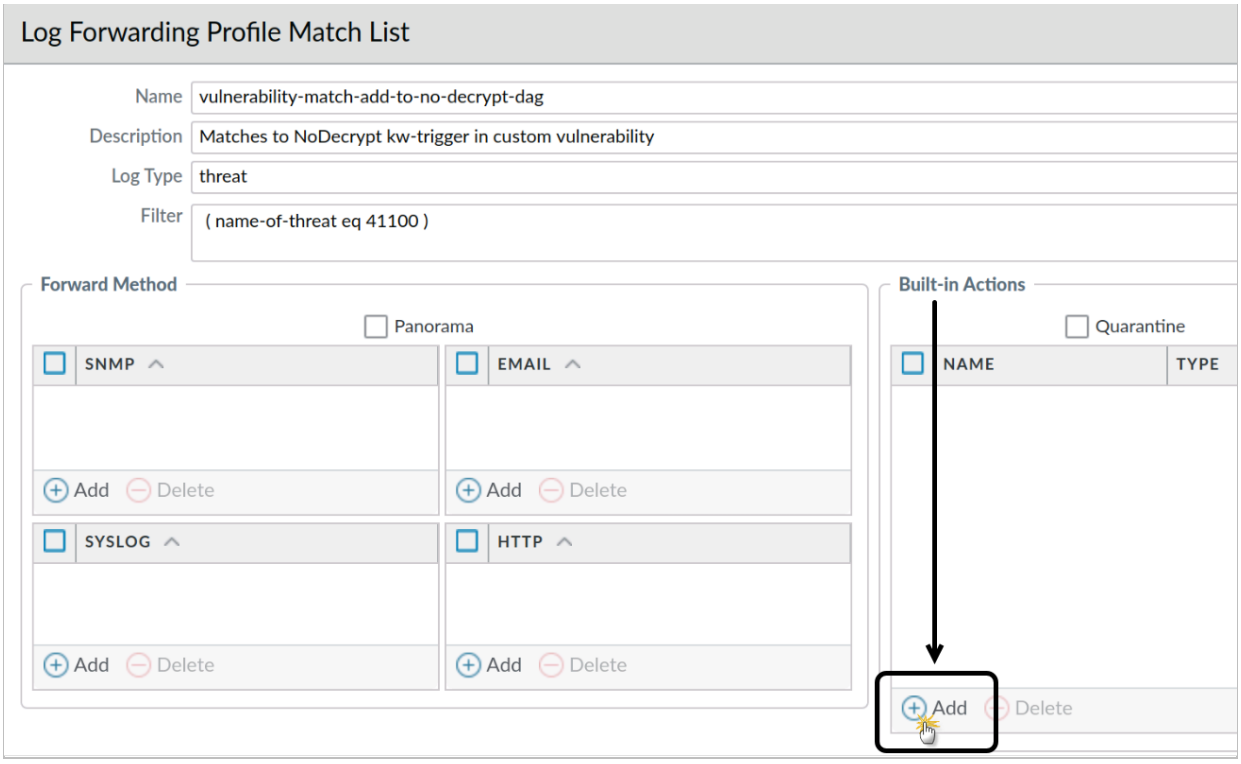
55. Configure the first four parameters of the new **Log Forwarding Profile Match List** as follows:

Parameter	Value
Name	**`vulnerability-match-add-to-no-decrypt-dag`**
Description	**`Matches to NoDecrypt kw-trigger in custom vulnerability`**
Log Type	**threat**
Filter	**`(name-of-threatid eq 41100)`**



Note: For the **Filter** specification, you can type the statement as specified. You also can use the **Filter Builder** option that appears on the drop-down list. You can use either the threat ID number or the alphanumeric name of the threat to match to the "name-of-threatid" attribute. In this activity, the threat ID number is specified. In the Filter Builder, you should select **Threat Name** from the **Attribute** pick list. After you click to **Add** your filter configuration to the filter statement, **Threat Name** will be rendered as **name-of-threatid**.

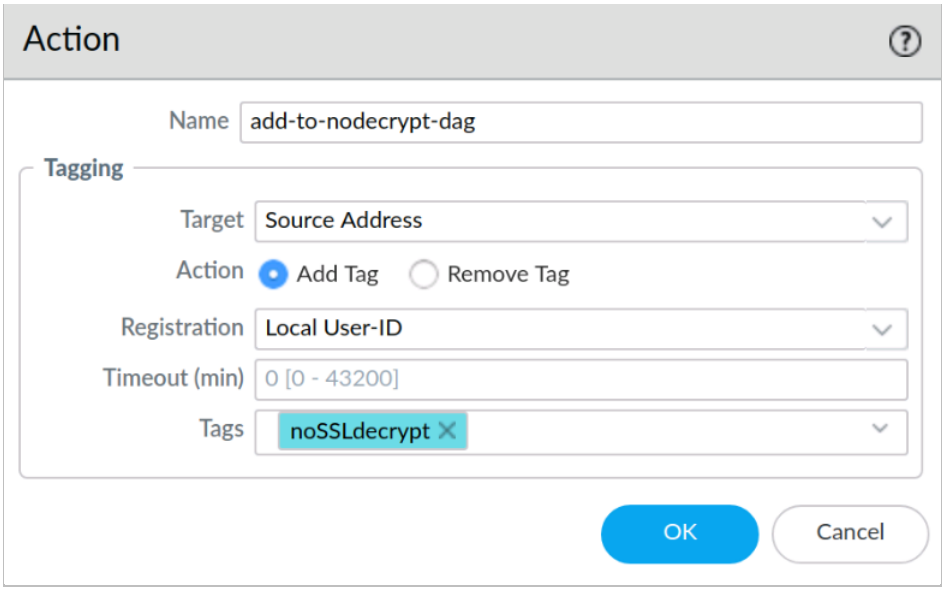
56. Click **Add** at the bottom of the **Built-in Actions** section of the **Log Forwarding Profile Match List** window:



57. Configure the **Action** using the following specifications:

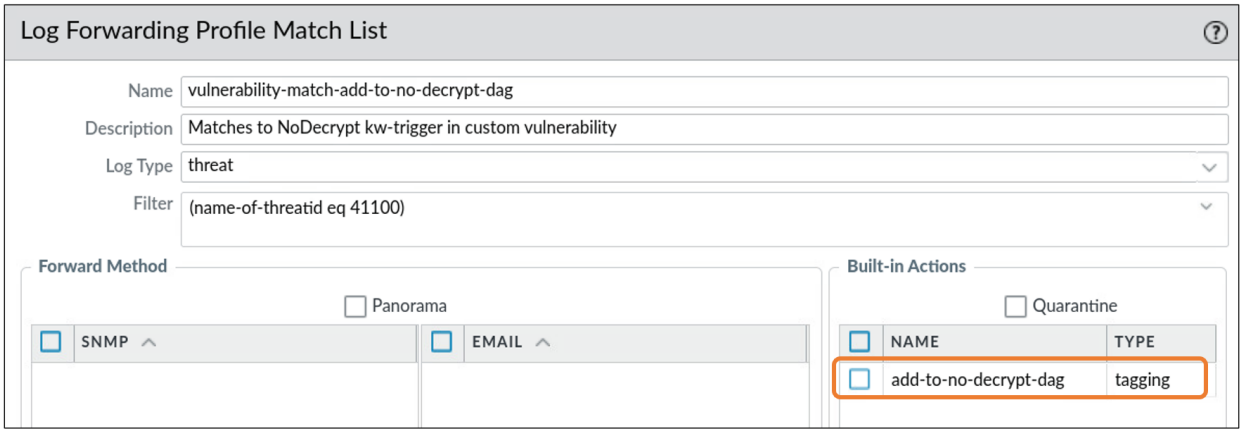
Parameter	Value
Name	**add-to-nodecrypt-dag**
Target	**Source Address**

Parameter	Value
Action	**Add Tag**
Registration	**Local User-ID**
Timeout (min)	**0**
Tags	**noSSLdecrypt**



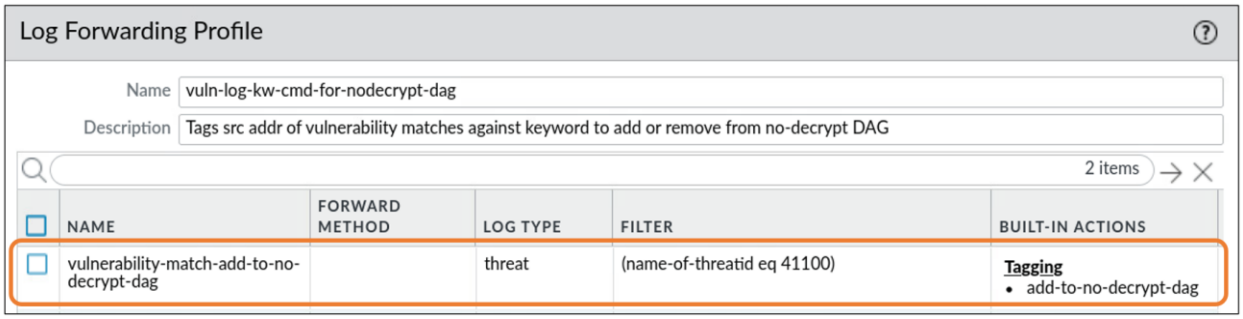
58. Click **OK**.

    The **Built-in Actions** list should be populated with the new action:



59. Click **OK**.

    The **Log Forwarding Profile** should be populated with the new filter and action:



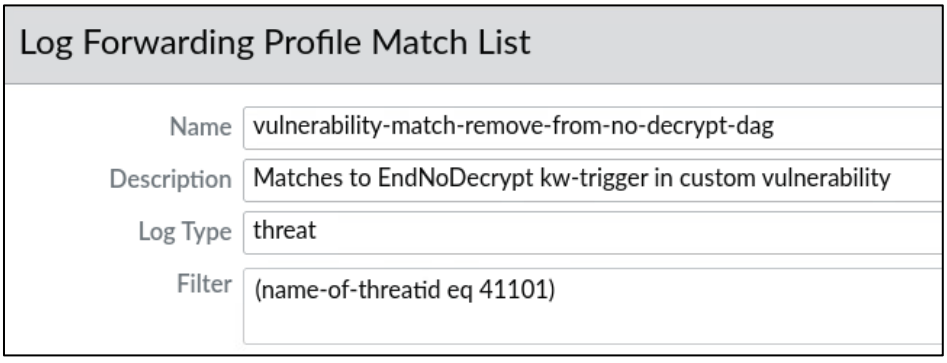
You now have created a log-forwarding filter and action to add the "noSSLdecrypt" tag to the source IP address of matching log entries. Next, you need to create a second filter for the threat ID that corresponds to the command to *remove* the tag.

60. Click **Add** at the bottom of the **Log Forwarding Profile** display area:

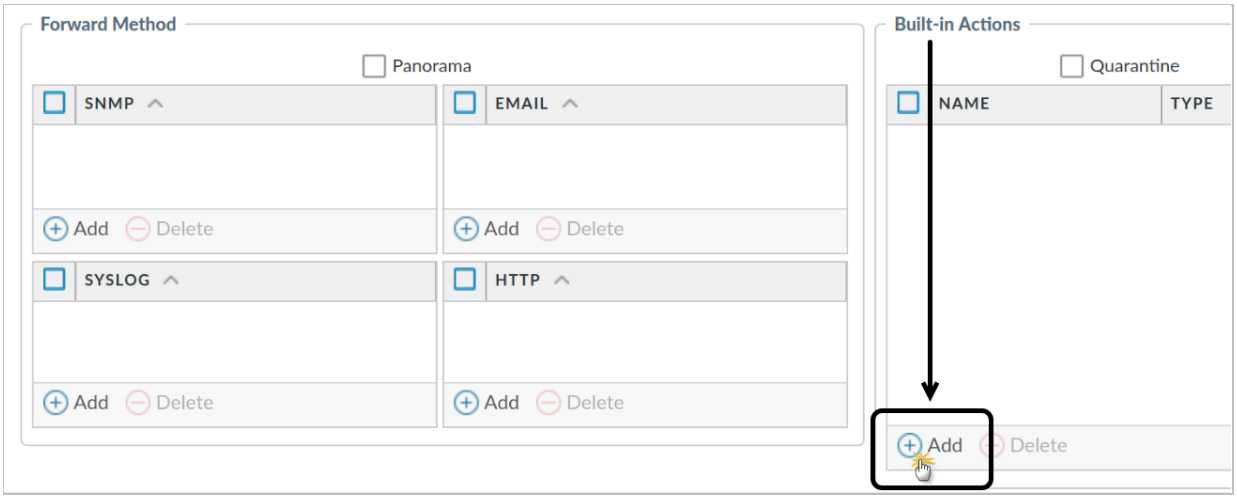
If you clicked **OK** too many times, and the **Log Forwarding Profile** window closed, click the name of the profile on the **Objects > Log Forwarding** page to re-open it, and then click **Add**.

61. Configure the first four parameters of the new **Log Forwarding Profile Match List** as follows:

Parameter	Value
Name	**vulnerability-match-remove-from-no-decrypt-dag**
Description	**Matches to EndNoDecrypt kw-trigger in custom vulnerability**
Log Type	**threat**
Filter	**(name-of-threatid eq 41101)**

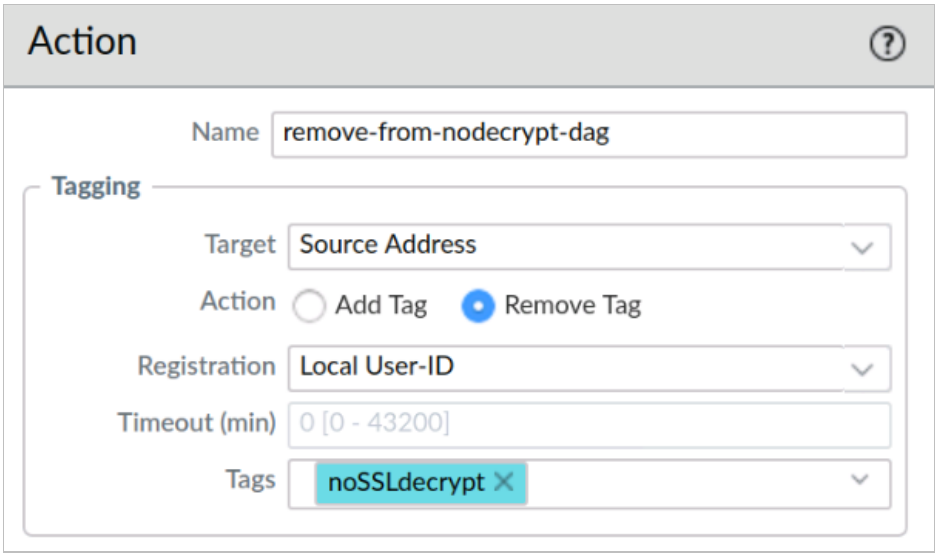


62. Click **Add** at the bottom of the **Built-in Actions** section of the **Log Forwarding Profile Match List** window:



63. Configure the **Action** using the following specifications:

Parameter	Value
Name	**`remove-from-nodecrypt-dag`**
Target	**Source Address**
Action	**Remove Tag**
Registration	**Local User-ID**
Timeout (min)	**0**
Tags	**noSSLdecrypt**

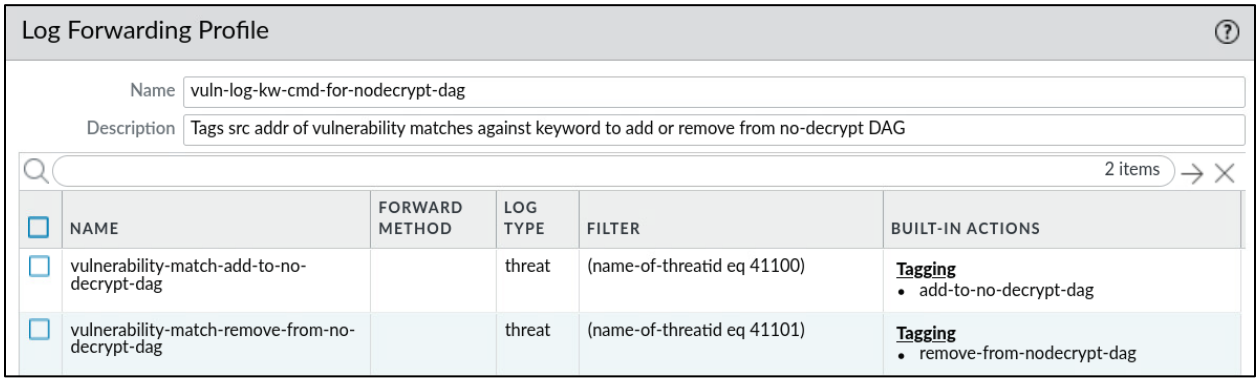


64. Click **OK**.

The **Built-in Actions** list should be populated with the new action.

65. Click **OK**.

    The **Log Forwarding Profile** should be populated with the new filter and action:



66. Click **OK**.

    The **Objects > Log Forwarding** page should be populated with the new profile:

NAME	DESCRIPTION	LOG TYPE	FILTER	BUILT-IN ACTIONS
vuln-log-kw-cmd-for-nodecrypt-dag	Tags src addr of vulnerability matches against keyword to add or remove from no-decrypt DAG	threat	(name-of-threatid eq 41100)	add-to-no-decrypt-dag
		threat	(name-of-threatid eq 41101)	remove-from-nodecrypt-dag

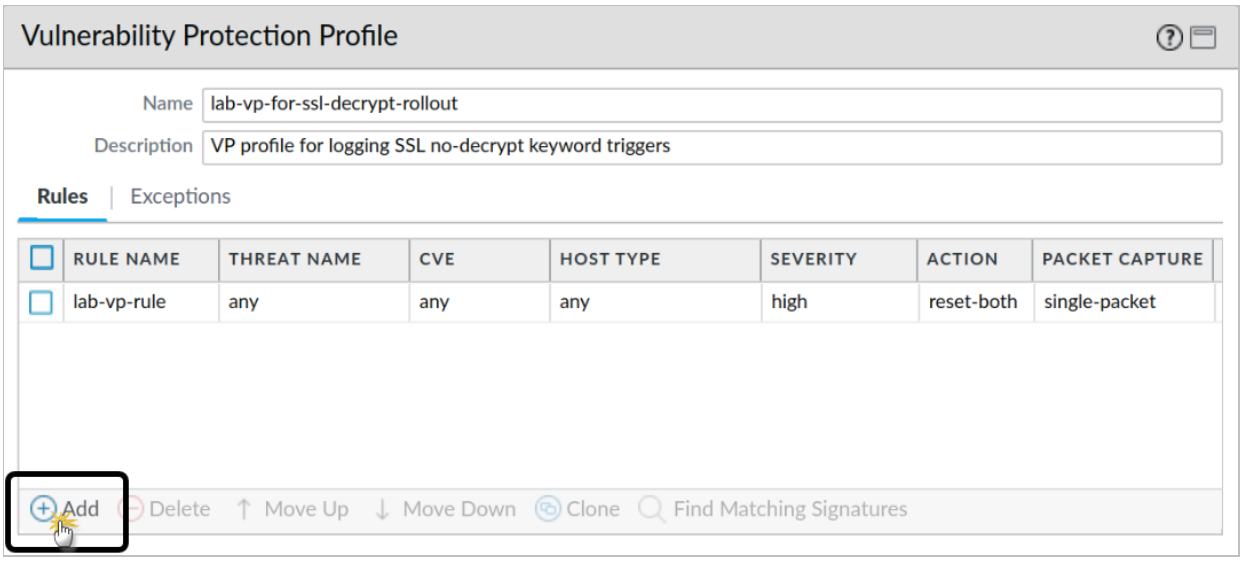
Note: Your web interface in the lab may show several additional columns that are removed in the example to display the relevant parameters in print more clearly.

8.7 Configure a Vulnerability Protection Profile to Generate Alerts

Before you can generate Threat logs based on matches against your custom Vulnerability signatures, you must configure a Security profile for Vulnerability Protection. You then must apply that Vulnerability Protection Profile and the Log Forwarding profile to the Security policy rule(s) that will allow the URL requests that contain your keyword commands.

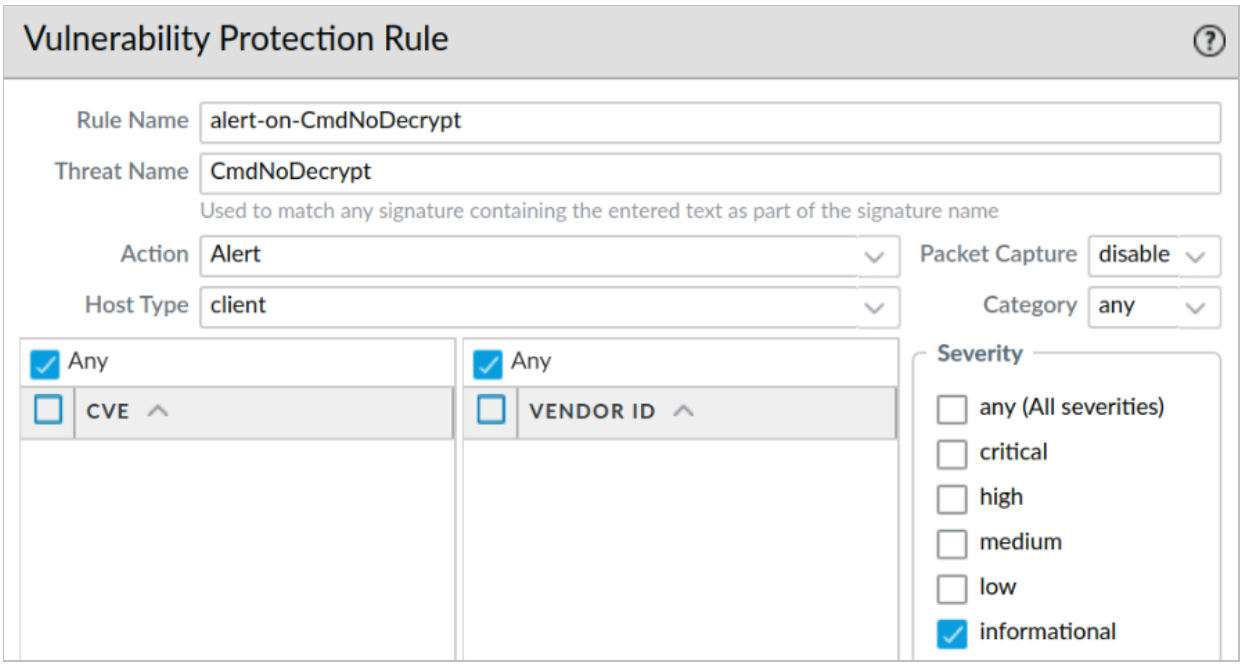
In this lab activity, you will modify a simple pre-existing Vulnerability Protection profile that already is attached to the required Security policy rule.

67. Go to **Objects > Security Profiles > Vulnerability Protection**:
68. Click **lab-vp**:
    - Change the **Name** to:
      `lab-vp-for-ssl-decrypt-rollout`
    - For the **Description**, type:
      `VP profile for logging SSL no-decrypt keyword triggers`
69. Click **Add** at the bottom of the content area of the **Rules** tab:



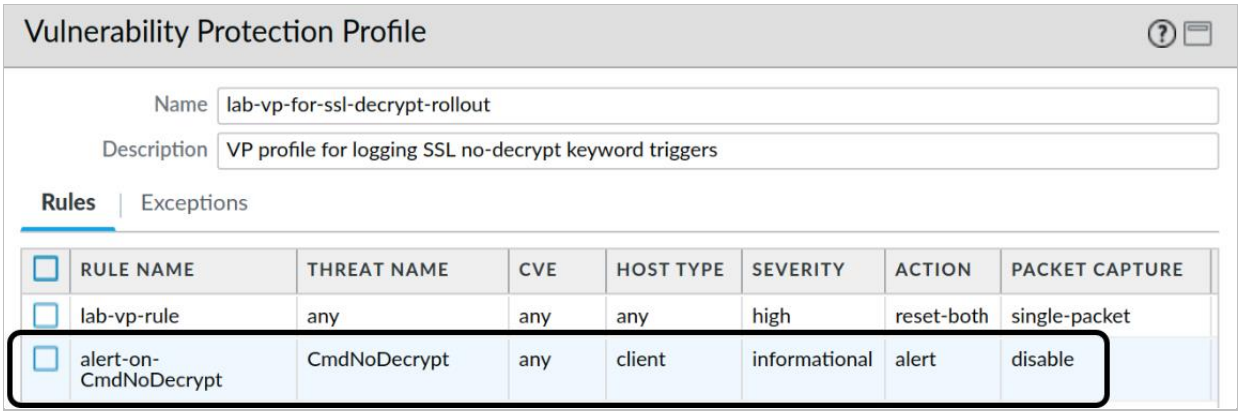
70. Configure the **Vulnerability Protection Rule** using the following specifications:

Parameter	Value
Rule Name	**`alert-on-CmdNoDecrypt`**
Threat Name	**`CmdNoDecrypt`**
Action	**Alert**
Packet Capture	**disable**
Host Type	**client**
Category	**any**
CVE	**Any (Selected)**
Vendor ID	**Any (Selected)**
Severity	**informational (Selected)** [All other options not selected]

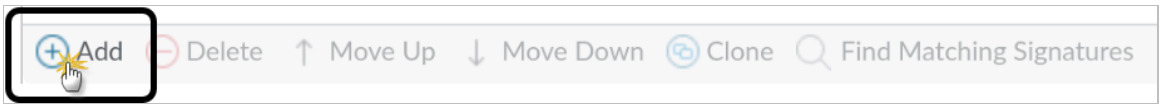


71. Click **OK**.

    The **Vulnerability Protection Profile** should be populated with the new rule:



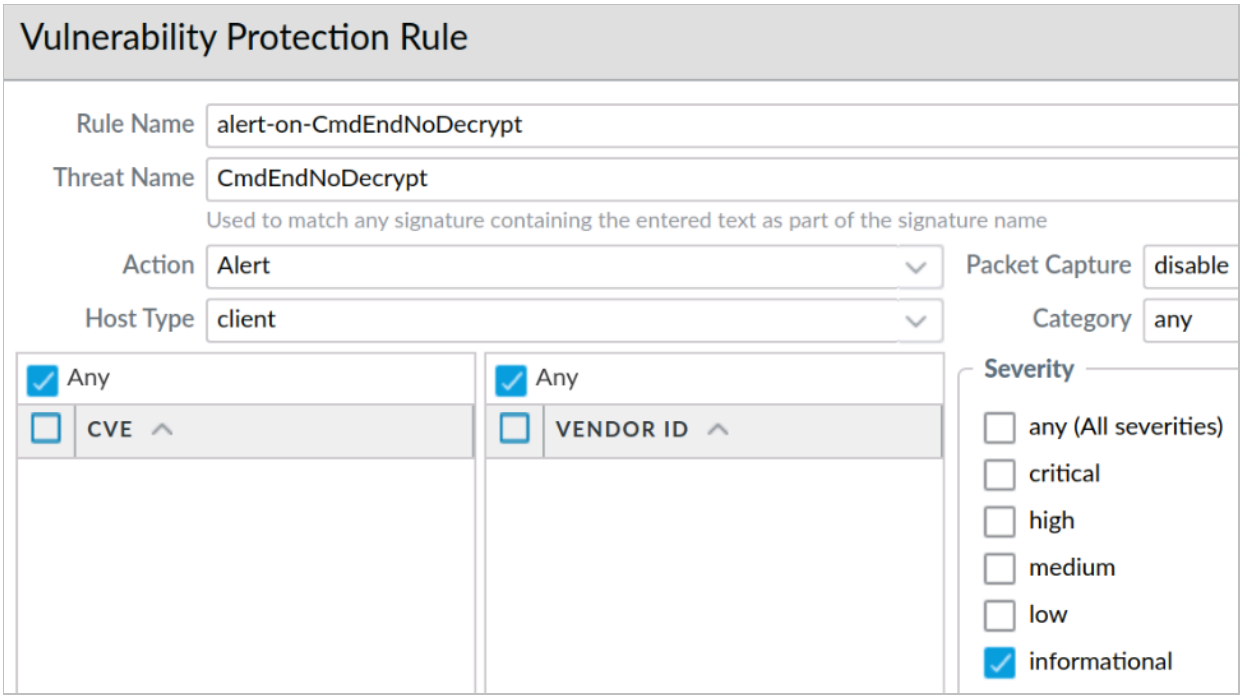
72. Click **Add** at the bottom of the content area of the **Rules** tab:



73. Configure the **Vulnerability Protection Rule** using the following specifications:

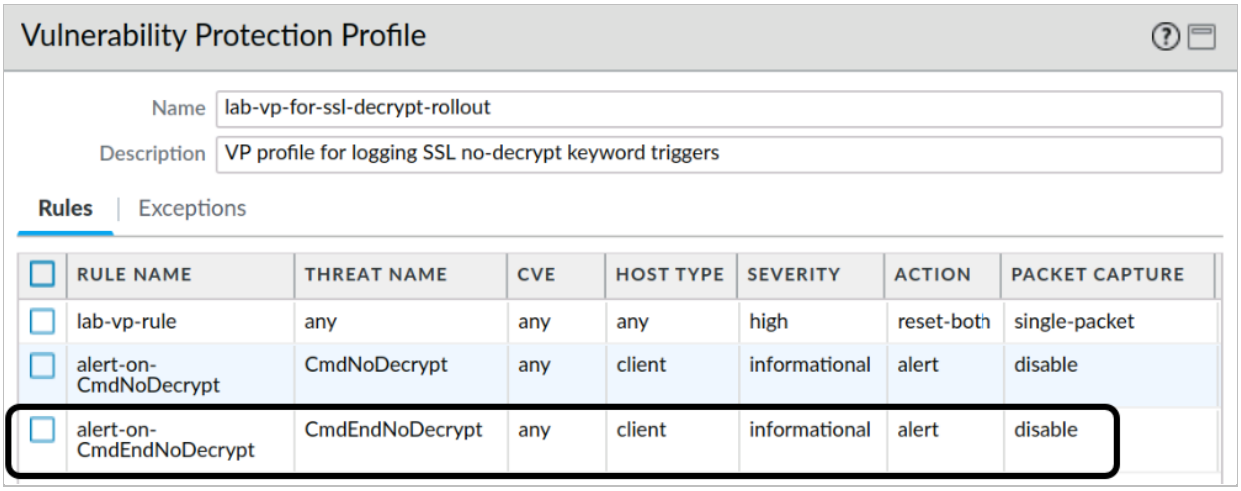
Parameter	Value
Rule Name	**`alert-on-CmdEndNoDecrypt`**
Threat Name	**`CmdEndNoDecrypt`**
Action	**Alert**

Parameter	Value
Packet Capture	**disable**
Host Type	**client**
Category	**any**
CVE	**Any (Selected)**
Vendor ID	**Any (Selected)**
Severity	**informational (Selected)** [All other options not selected]



74. Click **OK**.

The **Vulnerability Protection Profile** should be populated with the new rule:



Vulnerability Protection Profile

Name: lab-vp-for-ssl-decrypt-rollout

Description: VP profile for logging SSL no-decrypt keyword triggers

Rules | Exceptions

RULE NAME	THREAT NAME	CVE	HOST TYPE	SEVERITY	ACTION	PACKET CAPTURE
lab-vp-rule	any	any	any	high	reset-both	single-packet
alert-on-CmdNoDecrypt	CmdNoDecrypt	any	client	informational	alert	disable
alert-on-CmdEndNoDecrypt	CmdEndNoDecrypt	any	client	informational	alert	disable

75. Click **OK**.

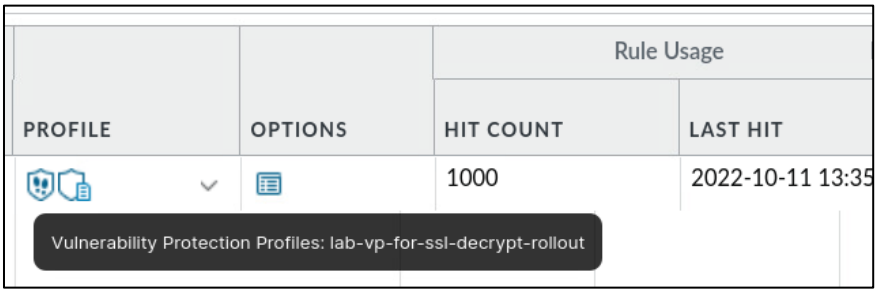
The **Objects > Security Profiles > Vulnerability Protection** page should be populated with the new rules:

NAME	COUNT	RULE NAME	THREAT NAME	HOST TYPE	SEVERITY	ACTION	PACKET CAPTURE
lab-vp-for-ssl-decrypt-rollout	Rules: 3	lab-vp-rule	any	any	high	reset-both	single-packet
		alert-on-CmdNoDecrypt	CmdNoDecrypt	client	informational	alert	disable
		alert-on-CmdEndNoDecrypt	CmdEndNoDecrypt	client	informational	alert	disable

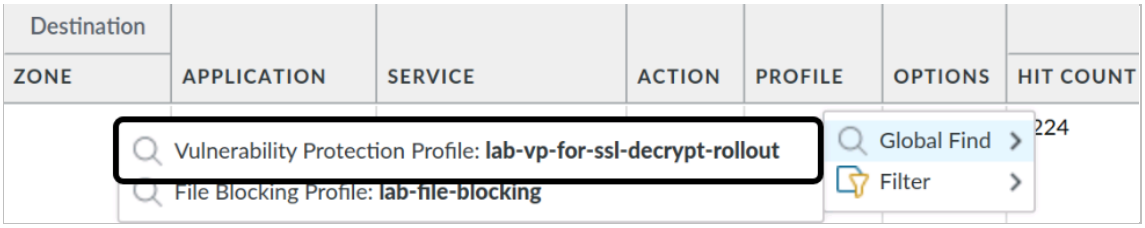
76. Go to **Policies > Security**:

	NAME	TAGS	Source ZONE	Destination ZONE	APPLICATION	SERVICE	ACTION	PROFILE
1	internal-inside-dmz	internal	inside	dmz	ftp ssh ssl web-browsing	application-default	Allow	
2	egress-outside	egress	inside	outside	any	application-default	Allow	none
3	egress-outside-threat-scan	egress	inside	outside	any	application-default	Allow	
4	danger-simulated-traffic	none	danger	danger	any	application-default	Allow	
5	intrazone-default	none	any	(intrazone)	any	any	Allow	none
6	interzone-default	none	any	any	any	any	Deny	none

For the **internal-inside-dmz** rule, hover your pointer over the Vulnerability Protection icon in the **Profile** column, then click the drop-down menu icon:



77. Hover your pointer over **Global Find** and verify that the **lab-vp-for-ssl-decrypt-rollout** Vulnerability Protection profile is applied:



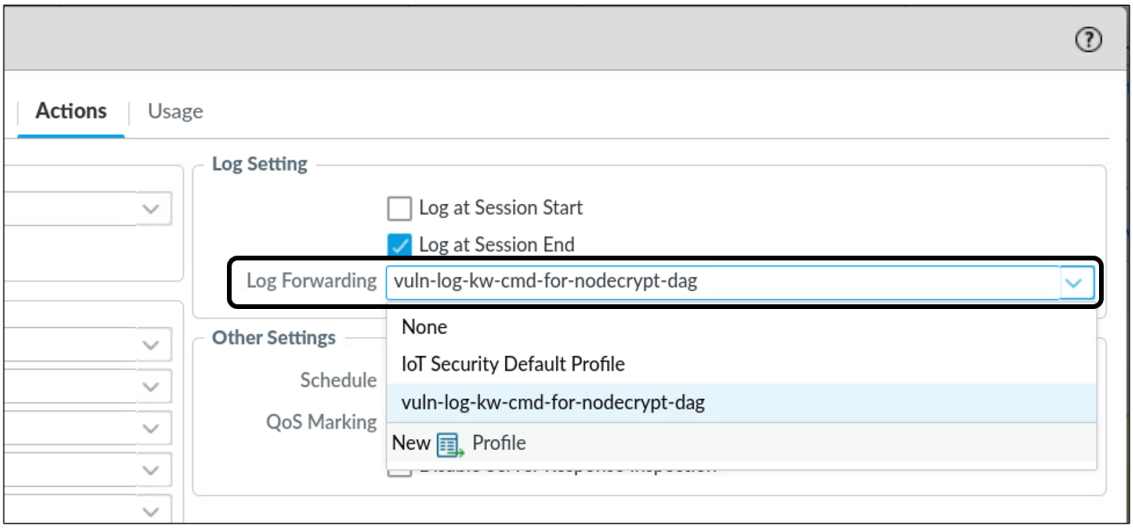
78. Click in an open area of the page to close the **Global Find** popup menu.

8.8 Add the Log Forwarding Profile to a Security Policy Rule

The automation scheme that you have built will enable you to use the standard web browser of a client to request a URL such as http://www.anywhere.any/cmdnodecrypt/. The firewall will detect "/cmdnodecrypt/" as a keyword "vulnerability" and generate a Threat log.

In this lab activity, you will apply the Log Forwarding profile to the Security policy rule that allows requests that you send to the application server in the DMZ zone. The effect of the firewall's detection of the request will apply to any Decryption policy rule that uses the "no-decrypt" DAG, regardless of the destination zone or IP address of the original trigger.

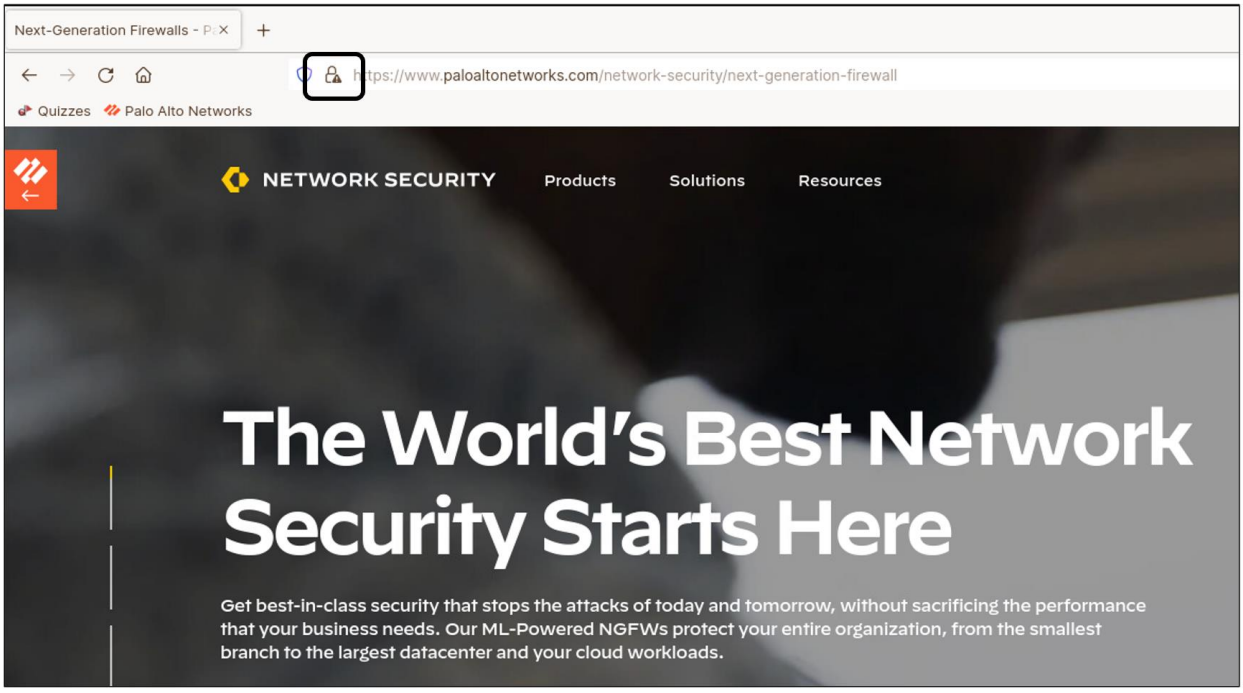
79. On the **Policies > Security** page, click **internal-inside-dmz**.
80. Click the **Actions** tab:
81. Click the **Log Forwarding** drop-down list and select: **vuln-log-kw-cmd-for-nodecrypt-dag**



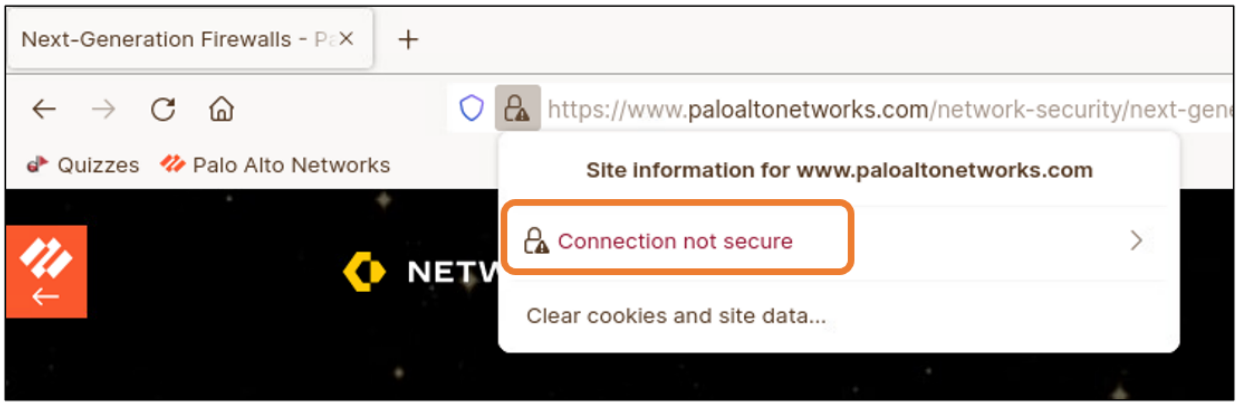
82. Click **OK**.
83. **Commit** the configuration.

8.9 Test the Configuration and Confirm Results

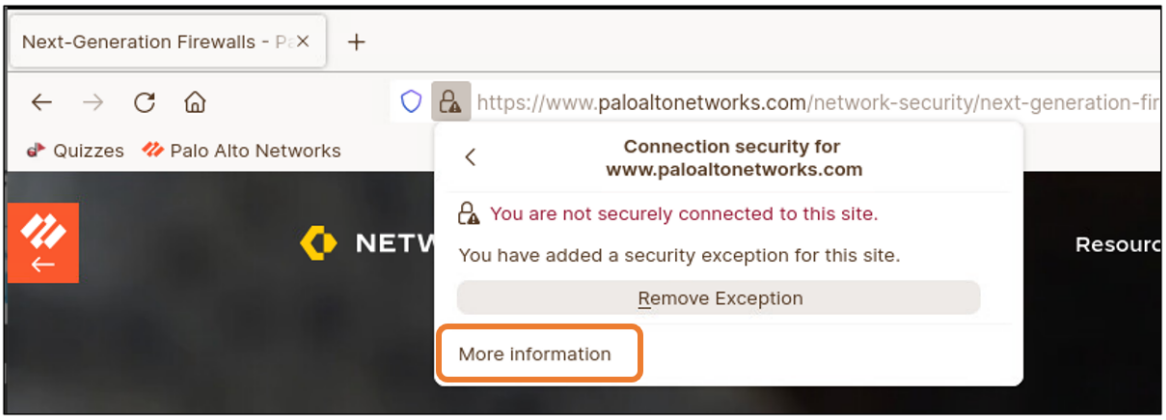
84. Launch Firefox and go to:
    `www.paloaltonetworks.com`
85. Click **Advanced**, then click **Accept the Risk and Continue**.
    The browser should connect to the site, render the target page in the content area, and display a certificate error notification in the address bar:



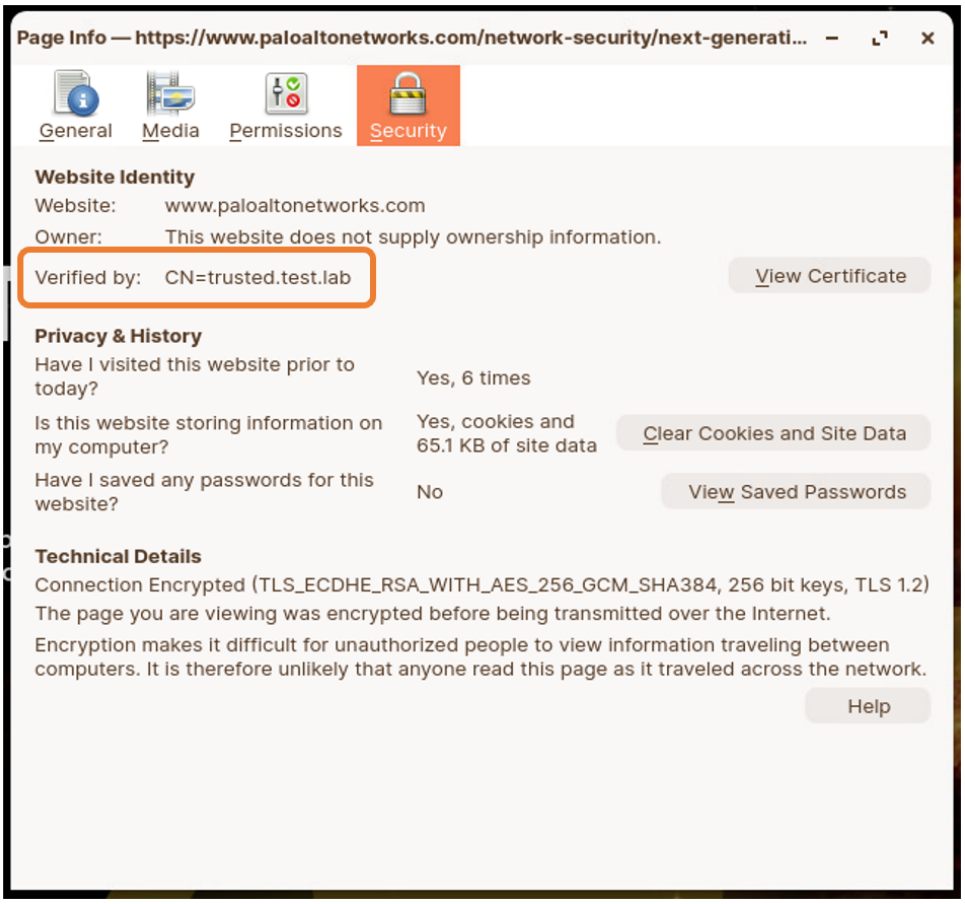
86. Click **Connection not secure** in the address bar:



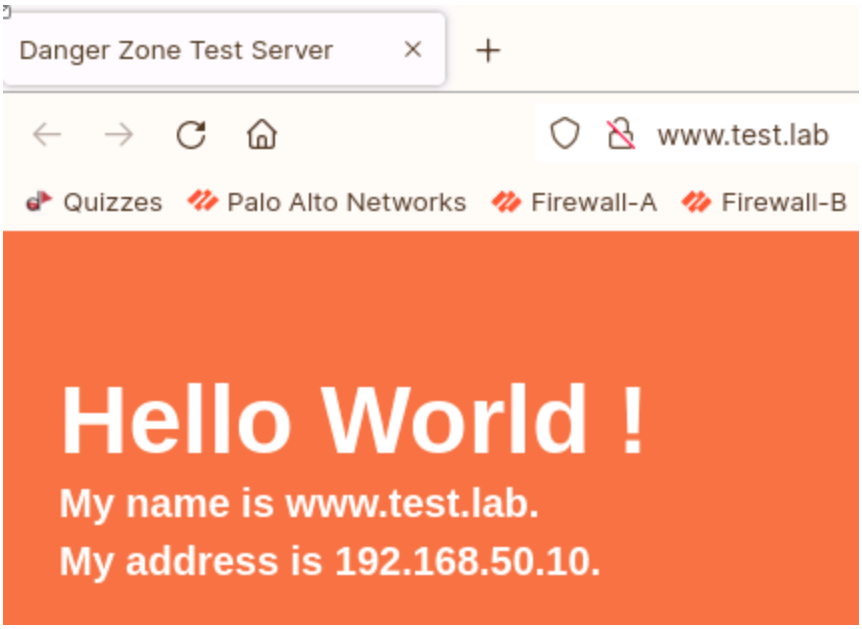
87. Click **More Information**.



Verify that the certificate is issued by “trusted.test.lab.” This procedure confirms that the baseline Decryption policy still is functional after the commit of your configuration changes.



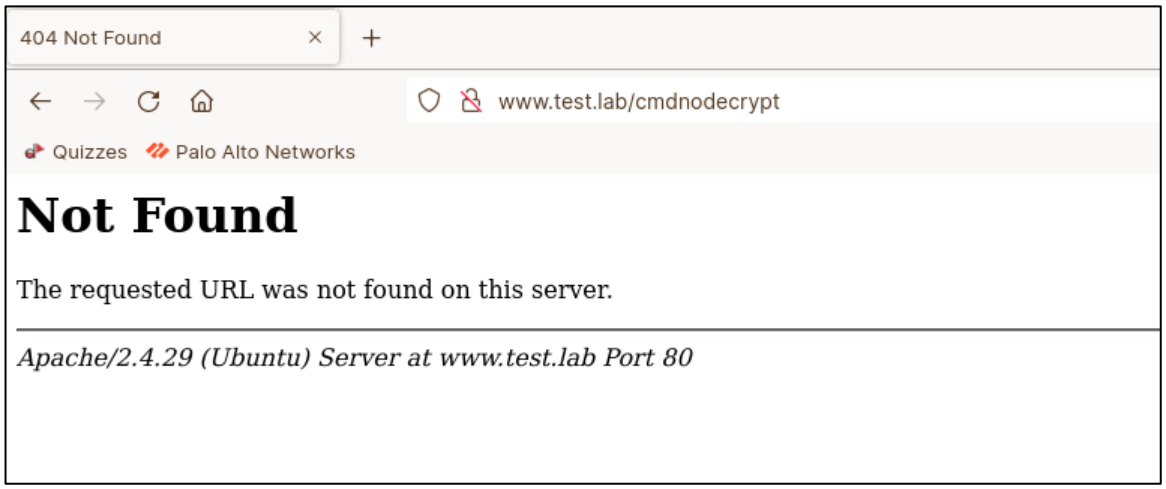
88. Close the **Certificate** window.
89. Close **Firefox**.
90. Launch a new instance of **Firefox** and go to:
    `http://www.test.lab`



This procedure verifies access to the DMZ server. It also tests basic functionality of a Security rule that:

- allows access,
- applies the required Vulnerability profile, and
- applies the required Log Forwarding profile that tags and untags the source IP address based on the log results.

91. Clear the text in the address bar of **Firefox** and type: **`http://www.test.lab/cmdnodecrypt/`** and press **Enter**:

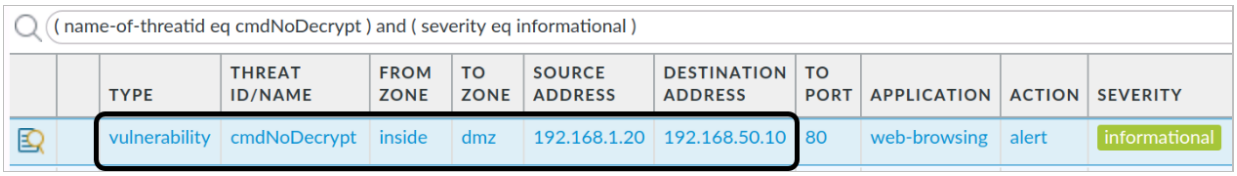


An HTTP 404 Not Found error is the expected result, because the destination "/cmdnodecrypt/" does not exist. However, the firewall still should detect this text string as a match to the custom Vulnerability signature.

92. Close **Firefox**.

Closing the browser ends the session and thereby ensures that entries in the Traffic and Threat logs are created. Threat logs entries typically are generated after the initial detection is processed, even if the corresponding Security policy rule is configured to generate Traffic logs only at the end of the session.

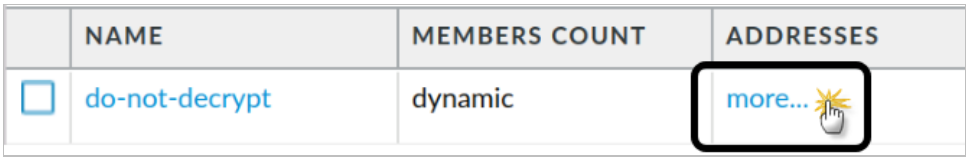
93. Go to **Monitor** > **Logs** > **Threat** in the web interface of the firewall.
    Look for the Threat log that corresponds to the detection of the custom **cmdNoDecrypt** Vulnerability signature in the URL request that you just sent to www.test.lab.



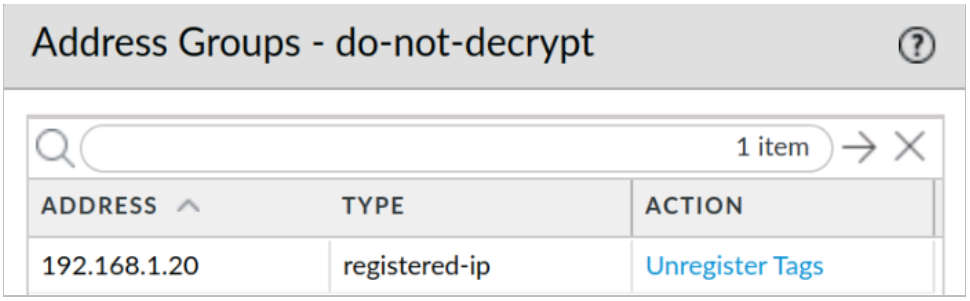
    If there are many log entries, filter for subtype “vulnerability” and severity “informational,” as shown in the example.

94. Go to **Monitor** > **Logs** > **IP-Tag**.
    Look for a registration event for the source IP address of the client (192.168.1.20) to which the **noSSLdecrypt** tag was applied. This log event confirms that the Threat log entry for the custom Vulnerability signature detection has been processed by the Log Forwarding profile.

95. Go to the **Objects** > **Address Groups**.
96. On the **Address Groups** page, you can verify the expected change in DAG membership based on the tag applied to the client IP address.



97. Click **more…** in the **Addresses** column:

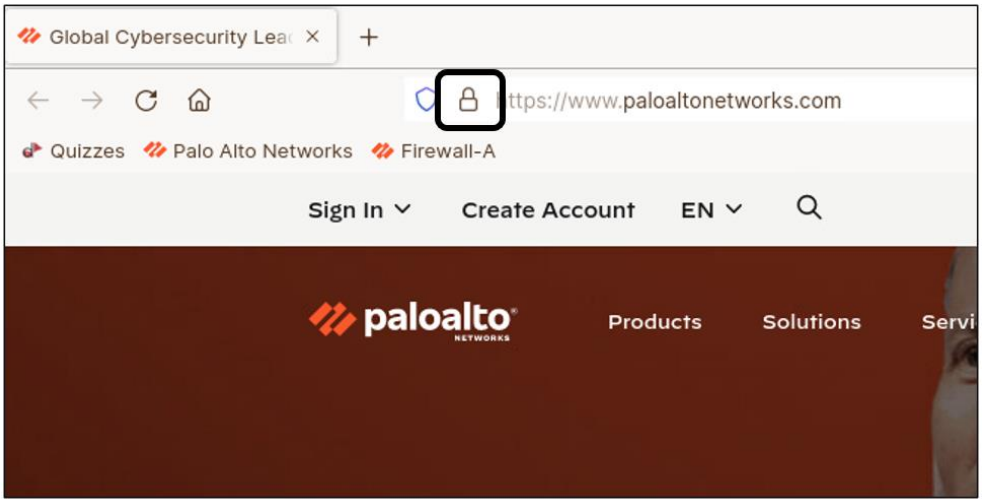


    Confirm that the IP address of the Student desktop (192.168.1.20) appears in the list.

98. Click **Close**.

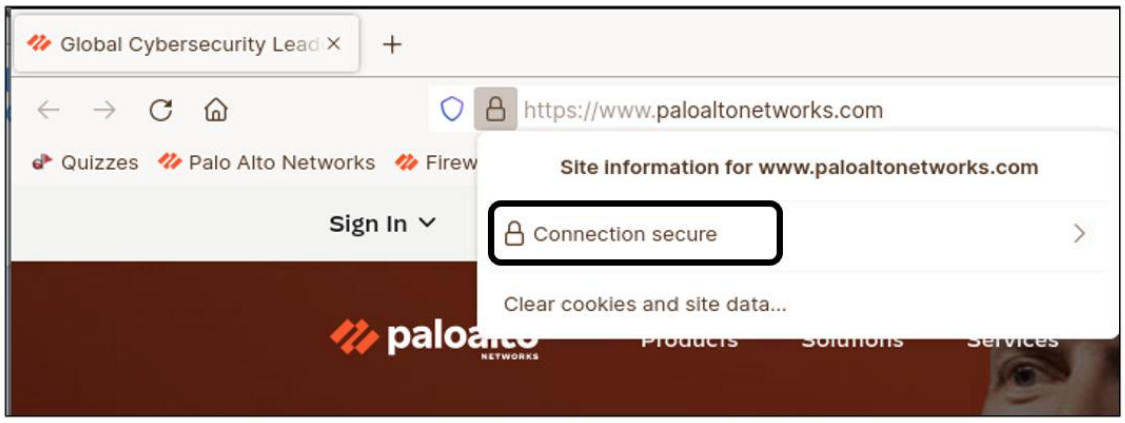
Next, you will use Firefox to confirm the expected change in the Decryption policy rule that matches to the client. For the website that you previously loaded (www.paloaltonetworks.com) and all other websites that the firewall might previously have intercepted for SSL decryption, the firewall now should match the source IP address of the client to the **do-not-decrypt-DAG** Decryption policy rule and apply the "no-decrypt" action.

99. Launch **Firefox** and go to:
    `www.paloaltonetworks.com`

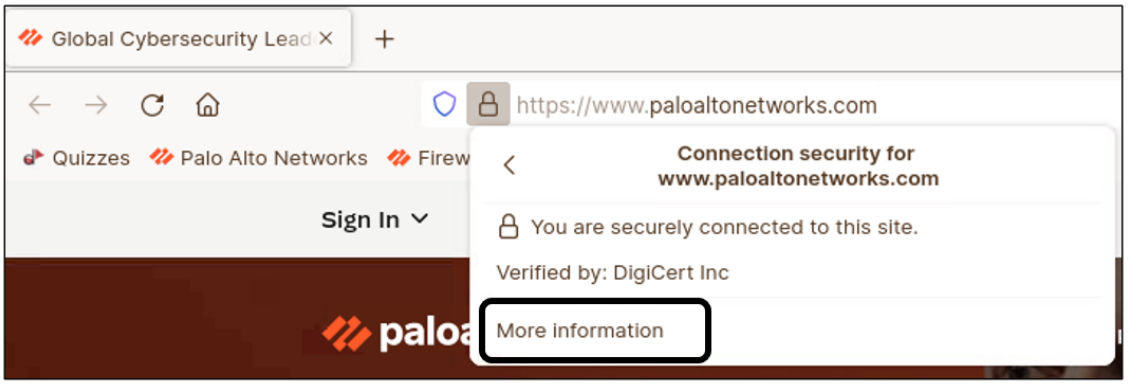


The webpage should load without notification of a problem with the security certificate.

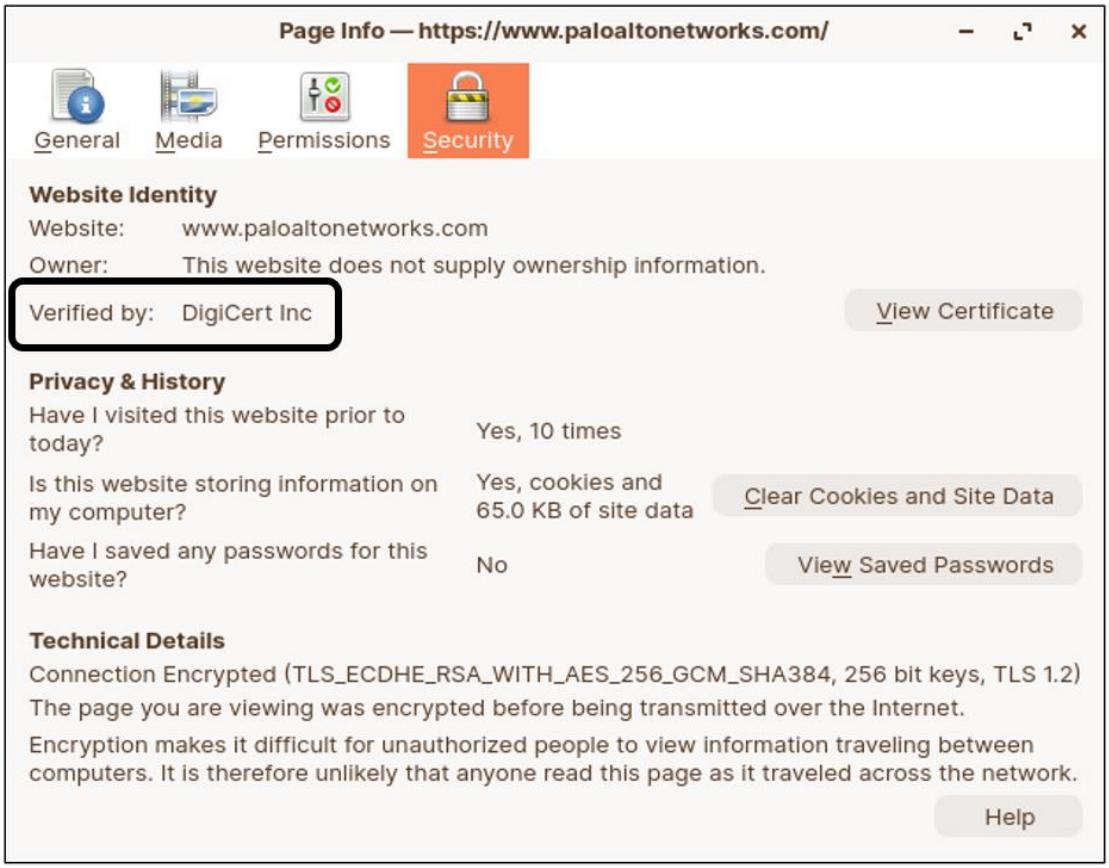
100. Click the **Lock icon** in the address bar:
101. Click the arrow to the right of **Connection secure**.



102. Select **More Information**.



103. Verify that the certificate authority referenced is from DigiCert Inc or from another external CA.

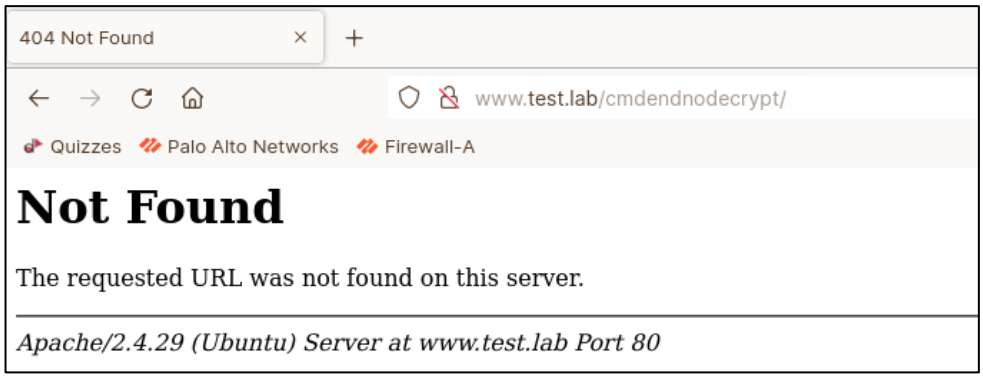


Note that certificate authority may not be Digicert at the time you perform this lab, but what you should not see is an entry for Verified by CN=trusted.test.lab.

104. Close the certificate**.**
105. Close the **Certificate** browser window.

Next, you will test the second “command” to remove the client IP address from the “no-decrypt” DAG.

106. Clear the text in the address bar of **Firefox** and type:
`http://www.test.lab/cmdendnodecrypt/` and press **Enter**.

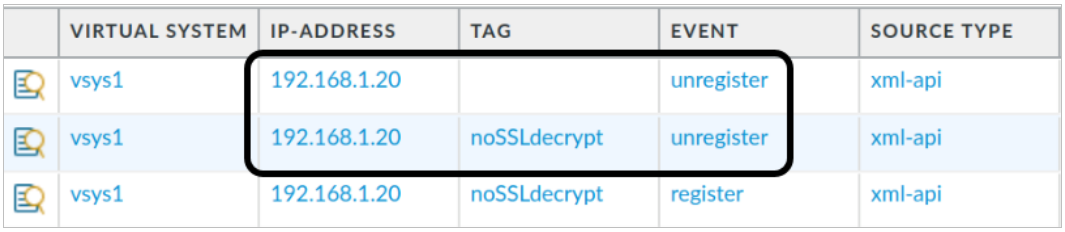


A Not Found error is the expected result; the destination "/cmdendnodecrypt/" does not exist. However, the firewall still should match this text string to the custom signature.

107. Close **Firefox**.

108. Go to **Monitor** > **Logs** > **IP-Tag**.

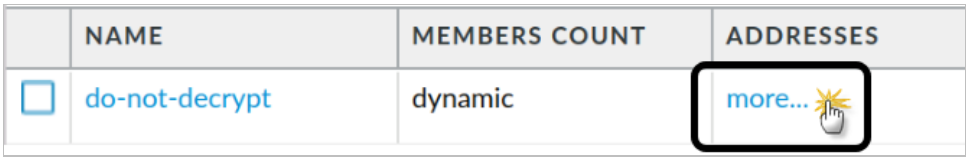
Look for two **unregister** events for the source IP address of the client (192.168.1.20) to which the **noSSLdecrypt** tag was applied. One event removes the tag; the other removes the IP address itself from the registration list that the firewall uses to track tagged IP addresses.



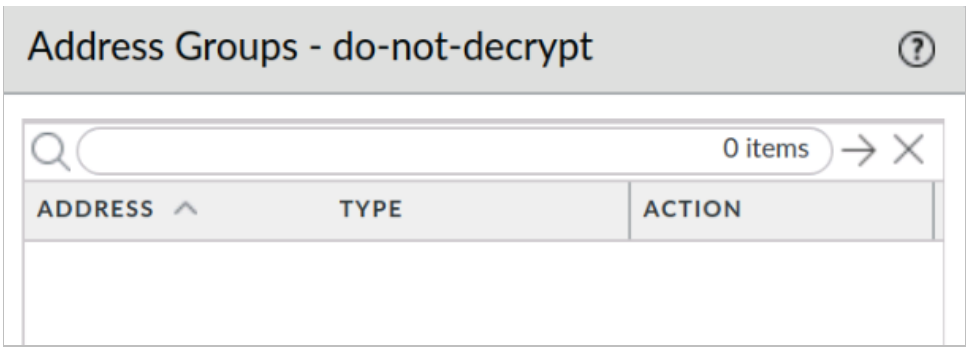
	VIRTUAL SYSTEM	IP-ADDRESS	TAG	EVENT	SOURCE TYPE
	vsys1	192.168.1.20		unregister	xml-api
	vsys1	192.168.1.20	noSSLdecrypt	unregister	xml-api
	vsys1	192.168.1.20	noSSLdecrypt	register	xml-api

109. Go to **Objects** > **Address Groups**.

110. Click **more…** in the **Addresses** column:



	NAME	MEMBERS COUNT	ADDRESSES
☐	do-not-decrypt	dynamic	more...



The address list now should be empty.

111. Click **Close**.
112. Launch **Firefox** and go to:

     `www.paloaltonetworks.com`

     Confirm that the Decryption policy applied to the client once again causes the firewall to intercept and decrypt the connection. You will know this because the firewall shows the certificate warning error thus proving that the firewall is again decrypting the traffic.

     Note: There are many use cases for dynamic management of address groups based on standard and custom-generated log data in combination with Log Forwarding profiles. This example has proved useful in troubleshooting SSL decryption by automating a change in applied Decryption policy for a specified host without being required to change and commit a new firewall configuration each time. You may have noticed that the conditions that are made available for the configuration of Log Forwarding objects include almost everything that is or can be logged.

8.10 Clean Up Your Lab Environment

113. Close all open **Firefox** tabs and windows. Clear any filters in logs.

     Leave open the **Firefox** browser and the connection to the web interface of the firewall.

114. If you have an open CLI connection to the firewall, leave the connection open for the next lab.

Stop. This is the end of the lab.

9. Lab: No User-ID Names in Logs

Lab Objectives

- Diagnose a common problem with User-ID
- Implement a solution

Lab Scenario

One of your firewall administrators recently configured the integrated User-ID agent. You perform a routine check of the Traffic log and notice that the **Source User** column is blank. Multiple configuration errors could be preventing User-ID from displaying the proper usernames:

FROM ZONE	TO ZONE	SOURCE	DESTINATION	SOURCE USER	TO PORT
inside	dmz	192.168.1.25	192.168.50.53		53
inside	dmz	192.168.1.25	192.168.50.53		53
inside	dmz	192.168.1.25	192.168.50.53		53

9.1 Apply a Baseline Configuration to the Firewall

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-09.xml**
   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.
2. After the load task is complete, **Commit** the configuration.
3. Open the Terminal on the student desktop and type:
   `cd Common` <**Enter**>; then type:
   `./url-traffic.sh` and press **Enter**.
   This script generates Traffic and URL log entries and allows you to verify the problem.
4. In the web interface, go to **Monitor** > **Logs** > **Traffic**.
5. In the filter box, type:
   `(addr.src in 192.168.1.20)` and press **Enter**.
6. Verify that usernames are not being mapped to new **Traffic** log entries by looking in the Source User column.

9.2 Diagnose and Fix the Problem

Use the information that is provided in the subsequent “Reference Information” section to help you diagnose and fix the problem.

You may consider this lab activity complete when:

- The firewall is writing a User-ID value in the Traffic log for any new web traffic generated from the Client (IP address 192.168.1.20).

- You can provide complete and accurate answers to the following questions:
  - What were the configuration issues?

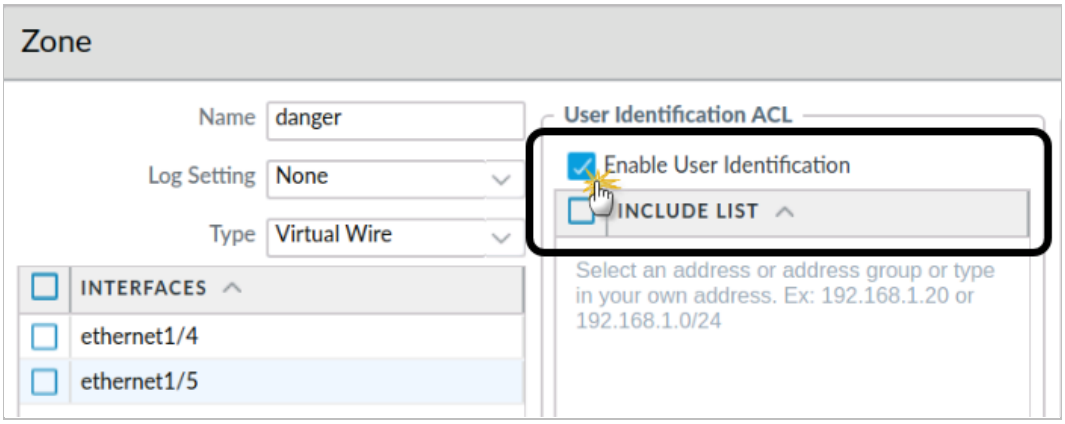
  - Which configuration tasks did you perform to resolve the issues?

9.3 Reference Information

This section identifies tasks that should be completed for User-ID to function properly. It also includes tasks that can help you test functionality. You can use this information to help you diagnose the problem and implement a solution. The steps to complete the tasks will be in the following section, 9.4.

- **User Identification** must be enabled on all **Zones** from which users will pass through the firewall using a valid username and password.

  These zones typically will be the internal zones of your network (the **inside** zone in the lab):



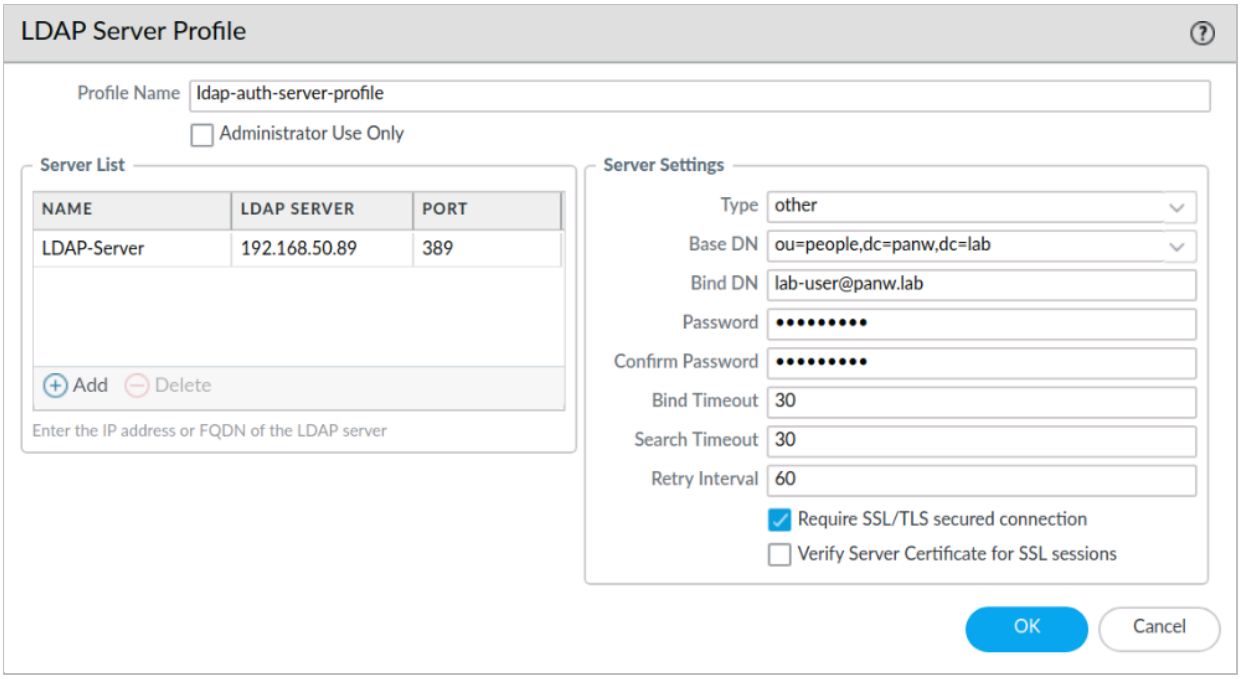
- A **Server Profile** will need to be configured to provide access to an authentication server, such as an LDAP or Kerberos Server.

  The Server profile will specify which server to connect to, the server type, and additional information such as a valid service account that will be used for the firewall to authenticate with

the authentication server. When you configure the authentication server, you can specify either the IP address of the server or the FQDN. A valid communication port also is required.

Hint: For Active Directory, the default LDAP port is 389, and for Kerberos the default port is 88.

Important: The password is: **`Pal0Alt0!`**



- To simulate a login event within the lab environment for the firewall to parse, you can run the **Appgenerator.sh** file from the **Server-Extranet**. This file is located in the **/home/paloalto42/pcaps92019/app.pcaps** folder:

```
Last login: Fri Apr  3 14:38:15 2020 from 192.168.1.20
paloalto42@extranet1:~$ cd /home/paloalto42/pcaps92019/app.pcaps
paloalto42@extranet1:~/pcaps92019/app.pcaps$ ./Appgenerator.sh
```

Note: You need to use the script to generate a security event in the lab environment.

- To verify that User-ID is functioning properly, use the **Traffic** log.
- To display the IP-to-user mappings in the CLI of the firewall, use the **`show user ip-user-mapping all`** command.

Stop. First, try to solve the lab problem on your own.
The steps in the following section provide an example diagnosis and solution.
Do not proceed without permission from your instructor.

9.4 Lab Solution: Enable User-ID on the Correct Zone

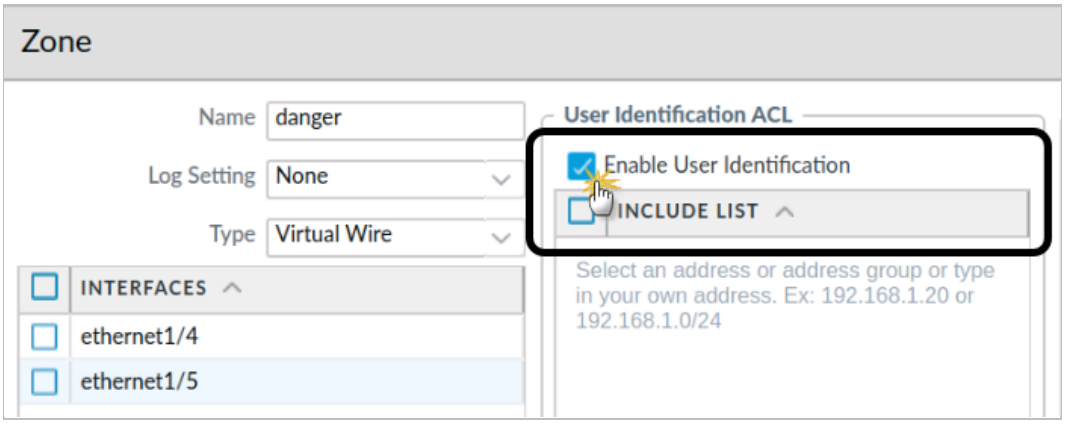
Diagnosis: One problem with the current configuration is that the **Zones** configuration has the **Enable User Identification** option enabled for the **outside** zone instead of the **danger** zone (the location of the source users).

> In this network configuration, source users originate connections from the **danger** zone. To ensure that User-ID functions properly, the **danger** zone must have the **Enable User Identification** option selected. In the current configuration, user identification enabled on the **outside** zone will not help to identify any users, and so user identification should be disabled:

	NAME	TYPE	INTERFACES / VIRTUAL SYSTEMS	ZONE PROTECTION PROFILE	PACKET BUFFER PROTECTION	LOG SETTING	User-ID ENABLED	User-ID INCLUDED NETWORKS	User-ID EXCLUDED NETWORKS
☐	danger	virtual-wire	ethernet1/4 ethernet1/5		☑		☐	any	none
☐	dmz	layer3	ethernet1/3		☑		☐	any	none
☐	inside	layer3	ethernet1/2		☑		☐	any	none
☐	outside	layer3	ethernet1/1		☑		☑	any	none

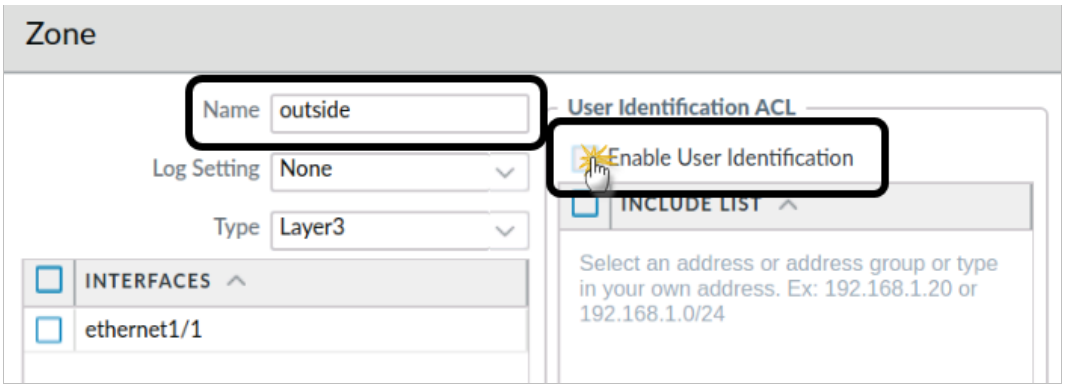
Resolution: Select **Enable User Identification** for the **danger** zone configuration.

1. In the web interface of the firewall, go to **Network** > **Zones**.
2. Click the name of the **danger** zone configuration.
3. Select the **Enable User Identification** check box:



4. Click **OK**.
5. Select the **Enable User Identification** check box on the **inside** zone as well.
6. Click **OK**.
7. Click the name of the **outside** zone configuration.
8. Deselect the **Enable User Identification** check box.

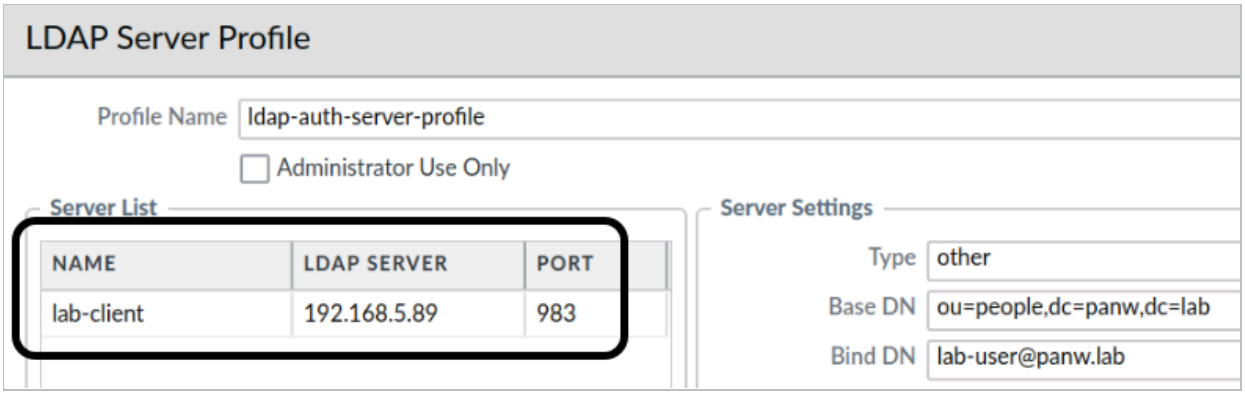
9. Click **OK**:



9.5 Lab Solution: Fix the LDAP Server Profile

Diagnosis: The IP address and port number for the LDAP server are incorrect.

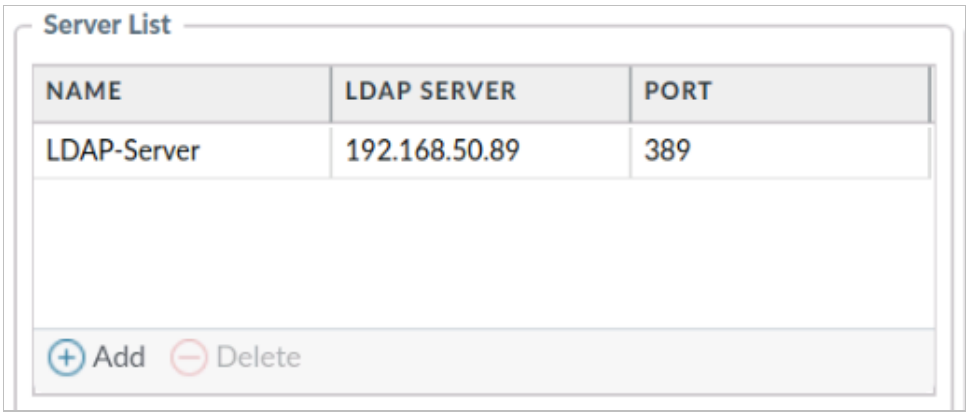
The IP address is set to **192.168.5.89**, but the correct IP address for the LDAP server is **192.168.50.89**. Also, the LDAP port is set to **983**, instead of **389**:



Resolution: Reconfigure the LDAP Server profile.

10. In the web interface, go to **Device > Server Profiles > LDAP**.
11. Click **ldap-auth-server-profile** to edit the LDAP Server profile.
12. In the **Server List**, update the server configuration to match the following parameters:

Parameter	Value
Name	**LDAP-Server**
LDAP Server	**192.168.50.89**
Port	**389**



13. Password and Confirm Password: **Pal0Alt0!**
14. Click **OK**.
15. After the tasks are complete, use the web interface to **Commit** the configuration.

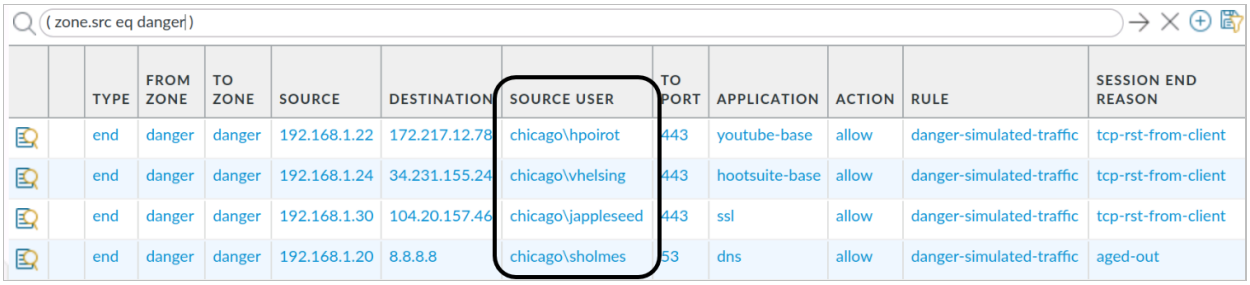
9.6 Lab Solution: Verify the Solution with Traffic Logs

You can use the following steps to generate web traffic and review the Traffic log to verify that User-ID is functioning properly.

16. The following script will generate User-ID and Group membership information. Open **Remmina** and double-click **Server-Extranet** and type the following:
    a. **`cd /home/paloalto42/pcaps92019/app.pcaps`** and press **Enter.**
    b. **`./Appgenerator.sh`** and press **Enter.**

    Note: If you want to clear the log files from the firewall, you can use the following command on the CLIENT workstation (not the server):
    c. **`cd /home/lab-user/Desktop/Lab-Files/Common`** and press **Enter.**
    d. **`./clearlogs-all.sh`** and press **Enter.**

    You can move to the next step before the file finishes.
17. In the web interface, go to **Monitor > Logs > Traffic**.
18. Notice that the **Source User** column (add it if is not visible) now shows various user accounts who source from the **danger zone**:



Note: The User-ID name could require up to three minutes to be populated in the log file. Click the **refresh** icon to update the log entries.

IMPORTANT: In an Active Directory environment you would need to configure the following on the firewall to capture User-ID traffic:

1. Enable User-ID on the source zone of the users: **Network > Zones**
2. Configure an LDAP Server profile that points to an Active Directory domain controller: **Device > Server Profiles > LDAP**
3. Configure an Authentication profile that is linked to the LDAP Server profile you just created: **Device > Authentication Profile**
4. Configure Server Monitoring of the Active Directory domain controllers you will have the User-ID agent query for User-ID-to-IP mappings: **Device > UserID > Server Monitoring**

9.7 Clean Up Your Lab Environment

19. Close all open testing browser tabs, windows, and applications. Clear any filters in logs.

    Leave open the connection to the web interface of the firewall.

20. Make sure the url-traffic.sh script is stopped.
21. If you have an open CLI connection to the firewall, leave the connection open for the next lab.

Stop. This is the end of the lab.

10. Lab: Troubleshooting GlobalProtect

Lab Objectives

- Provide user access through an external gateway
- Validate the functionality of a GlobalProtect Portal and external gateway that authenticates users using LDAP or a local user group configured on the firewall

Lab Scenario

As with other complex features, an examination of how GlobalProtect operates in a functional state will help you to understand how to interpret symptoms and diagnostic output when later troubleshooting a dysfunctional system. The following configuration requirements already have been completed:

- Configure the external gateway to provide IP addresses in the range of 192.168.100.200 to 192.168.100.250.
- Configure the tunnel interface to a new and separate security zone.
- Set up a Security policy rule to allow internet access for hosts that use the external gateway IP pool.
- Enable the external gateway to use the IPsec protocol.
- Create certificates for the portal and external gateway authentication.
- Create a Security policy rule to allow the internal host access to the portal and external gateway. This access will require the use of a no-NAT rule.

Subsequent steps will guide you through a validation of the system as it has been configured thus far. The discovery, diagnosis, and remediation of any problems with the configuration also will be addressed to help you establish a working GlobalProtect configuration.

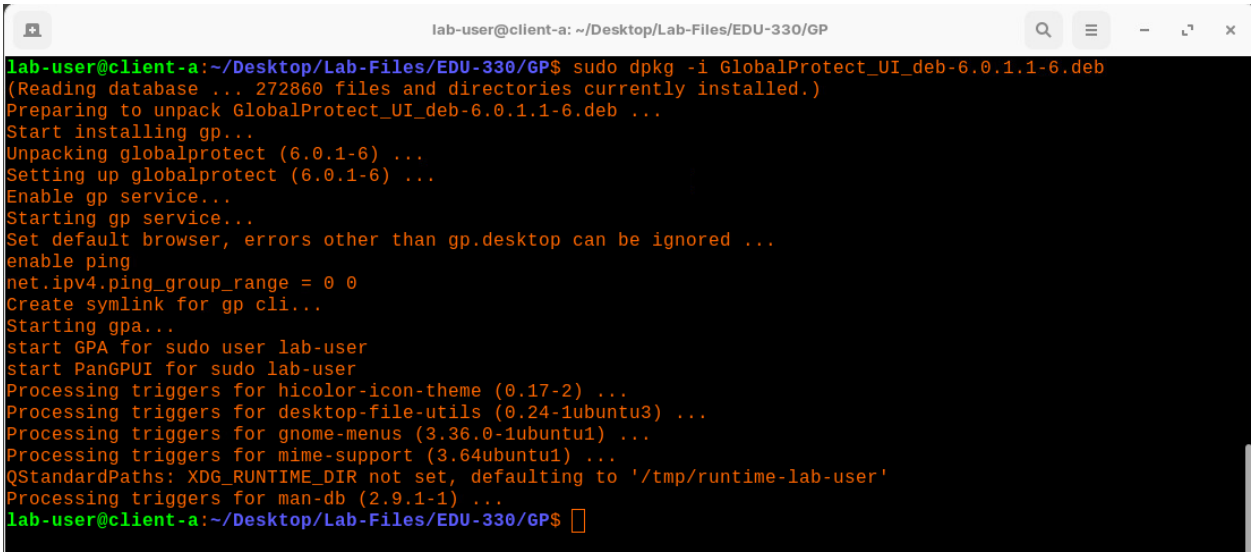
10.1 Apply a Baseline Configuration to the Firewall

1. Use the web interface to **import** and **load** the following configuration file: **330-FWA-11.1a-Start-Lab-10.xml**
   Import the file from the **/home/lab-user/Desktop/Lab-Files/EDU-330/firewall-config-files** folder. To load the file, leave all options in the bottom half of the **Load Named Configuration** window unselected.
2. After the load task is complete, use the web interface to **Commit** the configuration.
   You safely can ignore the warning about the use of IPv6 on tunnel.11.

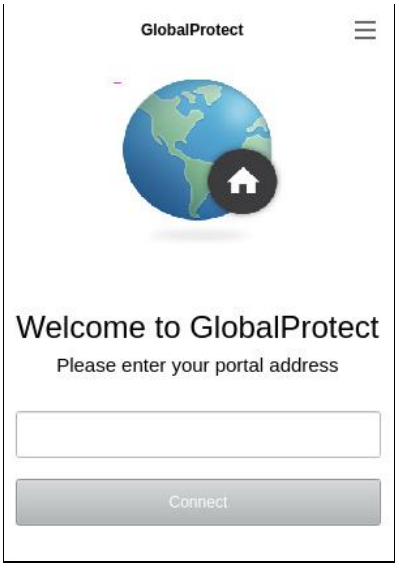
10.2 Install the GlobalProtect Agent

The installation files for the Linux GlobalProtect agent are already available on the client host so that you can easily install them.

3. Open the **Terminal** on your desktop.
4. Change the working directory by typing in the following command:
   `cd /home/lab-user/Desktop/Lab-Files/EDU-330/GP` and press **Enter**.
5. Install the GlobalProtect package by running the following command:
   `sudo dpkg -i GlobalProtect_UI_deb-6.0.1.1-6.deb <ENTER>`



6. When the process is complete, you should see a GlobalProtect login window appear onscreen:



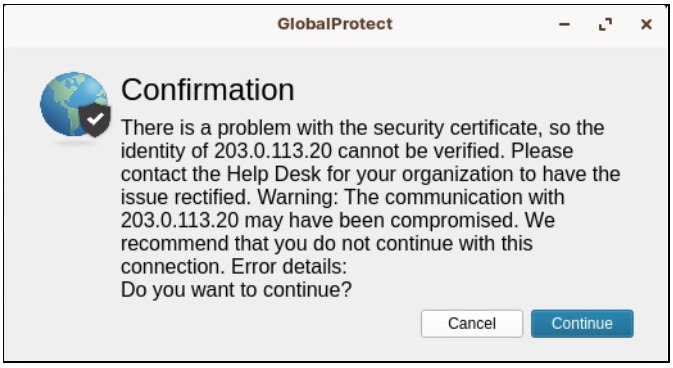
7. Leave the terminal window open.

10.3 Connect to the External Gateway

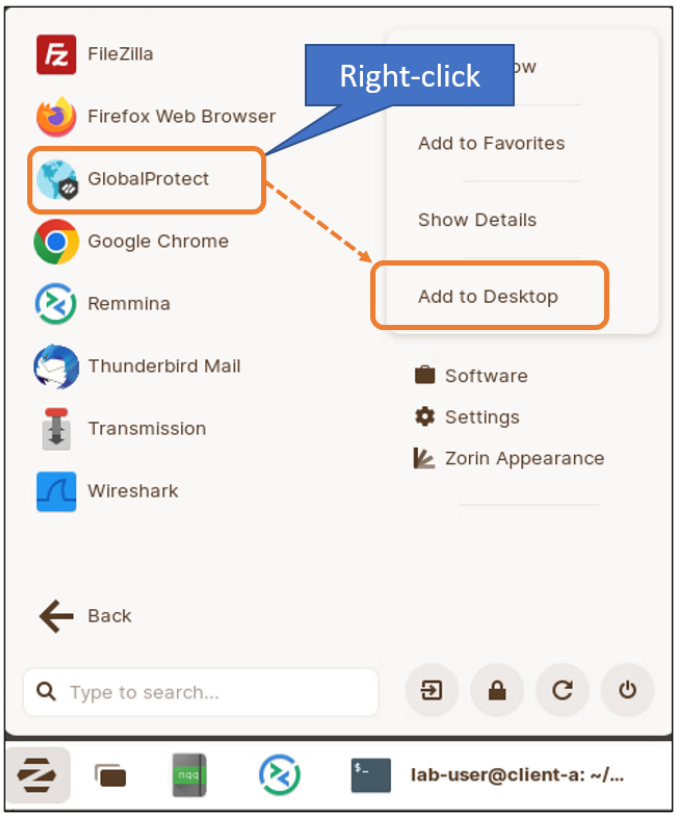
In the following steps, you will connect to the GlobalProtect Gateway.

8. Enter the IP address of the GlobalProtect Gateway – 203.0.113.20.

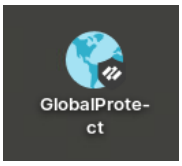
9. You will receive a notification about the certificate from the Gateway:



10. Click **Cancel**.
11. Create a shortcut for GlobalProtect on the desktop of the client.
12. Click the Application button in the bottom left corner of the desktop.
13. Choose Internet.
14. Right-click on the GlobalProtect icon and choose Add to Desktop:



15. You should now have an icon for GlobalProtect on the Desktop:

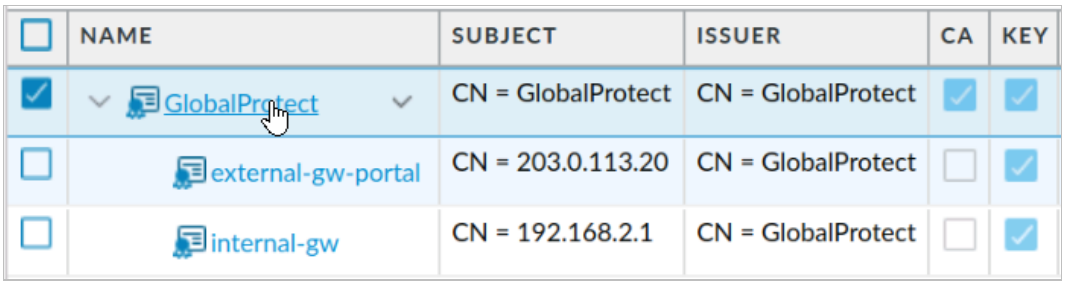


Note that if the GlobalProtect application window disappears while you are working through the remainder of this lab, use the icon to bring the window back.

10.4 Export GlobalProtect Certificate

Next, you will export the certificate from the firewall and import it into the student desktop so that GlobalProtect will accept it.

16. In the web interface of the firewall, go to **Device > Certificate Management > Certificates** and select the **GlobalProtect** CA certificate:

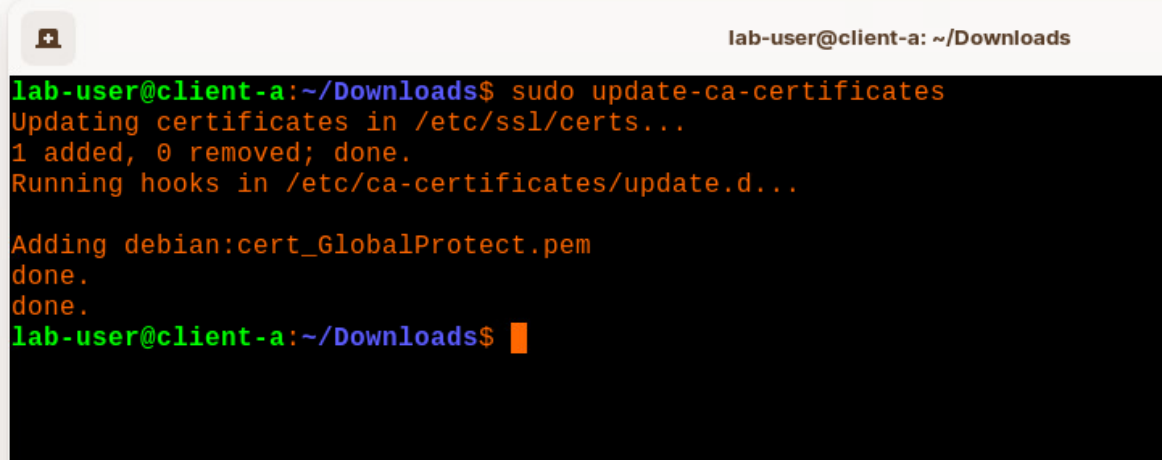


17. At the bottom of the content-display area, click **Export Certificate**.
18. Change the File Format to **Base64 Encoded Certificate (PEM)**.
19. Uncheck the option to Export Private Key.
20. Leave all other settings unconfigured.

21. Click **OK** to export the certificate.
22. If prompted, select the Downloads folder and click Save:

23. Select the open terminal window on the client desktop.
24. Create a new directory to store the GlobalProtect certificate:
    `sudo mkdir /usr/local/share/ca-certificates/extra` <**Enter**>

25. Change to the Downloads folder:
    `cd /home/lab-user/Downloads <Enter>`
26. Examine the contents of the directory using **`ls`** and press **Enter**.
27. Copy the cert_GlobalProtect.crt file to the ca-certificates/extra folder you created:
    `sudo cp cert_GlobalProtect.crt /usr/local/share/ca-certificates/extra/` and press **Enter**.
28. Update the list of certificates used by the host by issuing the following command:
    `sudo update-ca-certificates <Enter>`



29. Leave the terminal window open to use later in this lab.
30. Locate the GlobalProtect connection window and click Connect:

31. For **Username**, enter **`lab-user`**
32. For **Password**, enter **`Pal0Alt0!`**

    The connection will still fail, but for a different reason. For connectivity issues, the firewall Traffic logs might provide useful information about the nature of the failure.

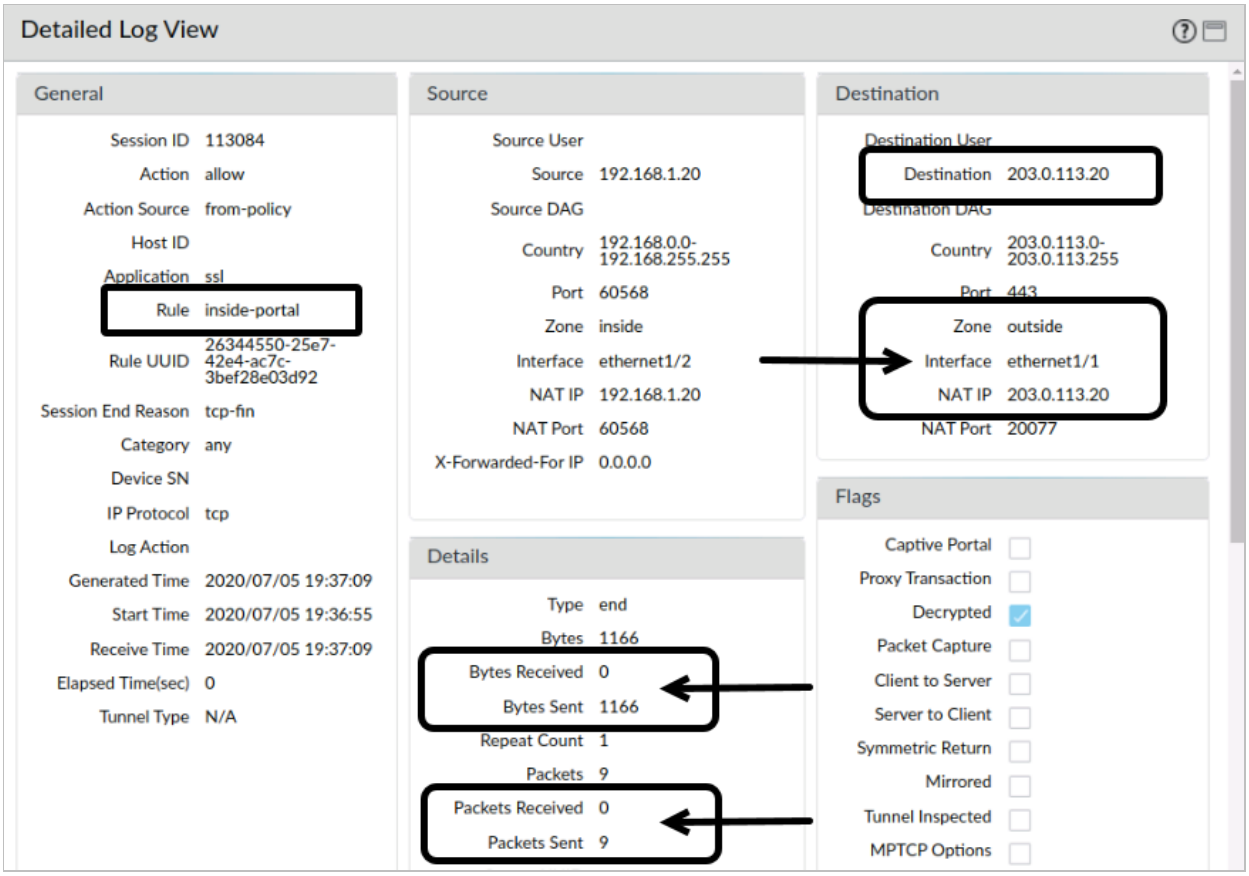
33. Go to **Monitor > Logs > Traffic** and filter for the IP address of the client (source) and the IP address of the GlobalProtect gateway (destination):
`( addr.src in 192.168.1.20 ) and ( addr.dst in 203.0.113.20 )`

TYPE	FROM ZONE	TO ZONE	SOURCE	DESTINATION	TO PORT	APPLICATION	ACTION	RULE	SESSION END REASON
end	inside	outside	192.168.1.20	203.0.113.20	443	ssl	allow	inside-portal	tcp-fin
end	inside	outside	192.168.1.20	203.0.113.20	443	panos-global-protect	allow	inside-portal	tcp-fin
end	inside	outside	192.168.1.20	203.0.113.20	443	panos-global-protect	allow	inside-portal	tcp-fin
end	inside	outside	192.168.1.20	203.0.113.20	443	ssl	allow	inside-portal	tcp-fin
end	inside	outside	192.168.1.20	203.0.113.20	443	ssl	allow	inside-portal	tcp-fin
end	inside	outside	192.168.1.20	203.0.113.20	443	panos-global-protect	allow	inside-portal	tcp-fin

In the details of the Traffic logs (magnifying glass icon), you will find sessions related to GlobalProtect portal communication that provide evidence of successful connections and completed application identification. Notice that the byte count for sessions identified as application type “panos-global-protect” ranges from about 2 to 12 kilobytes. However, sessions identified as “ssl” have a consistent byte count around 1 kilobytes.

Note: The Session End Reason recorded in Traffic logs can be misleading. An end reason of “tcp-fin” does not necessarily mean that an initial TCP handshake was completed. Various clients may end sessions in various ways. For example, an end reason of “tcp-rst-from-client” does not necessarily mean that the session was not also closed in conjunction with a “FIN–FIN-ACK” packet sequence.

34. Click the **magnifying glass** icon to review the session details for one of the sessions identified as "**ssl**":

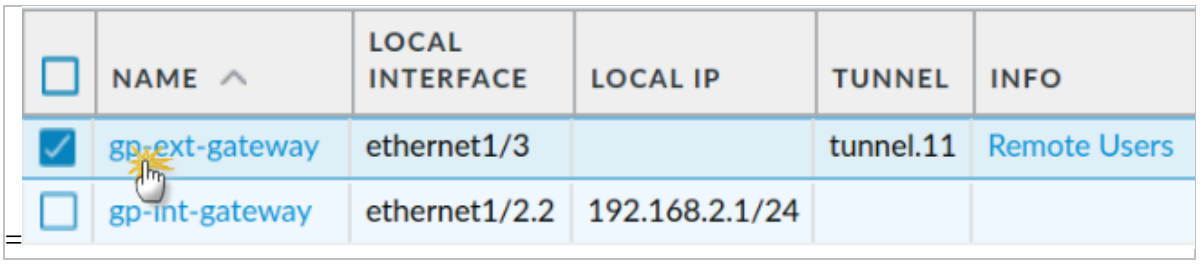


The session details appear to validate the connectivity problem reported in the last error displayed by the user interface of the GlobalProtect client. The client cannot establish a connection with the gateway because neither the client nor the firewall has received any response packets from the gateway.

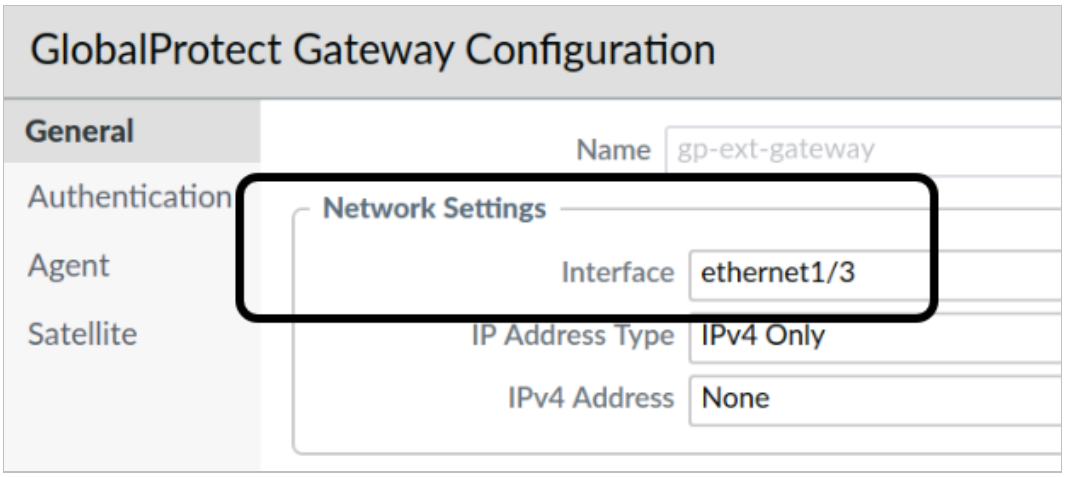
If any traffic was shown to be received by the firewall and transmitted back to the client, it would be a good idea to generate and inspect the client logs to try to discover the content of the return traffic and how the client might have interpreted it. However, with evidence of zero bytes of information being returned to the client from the gateway, a decision to put the focus of your investigation on the existence and health of the gateway itself is reasonable.

Note the destination IP address (**203.113.0.20**), destination zone (**outside**), and destination interface (**ethernet1/1**) to which the firewall forwards the sender (client-to-gateway) traffic.

35. Close Detailed Log View.
36. Go to **Network > GlobalProtect > Gateways**.
37. Click **gp-ext-gateway**:



38. Note which interface is configured for the Gateway:



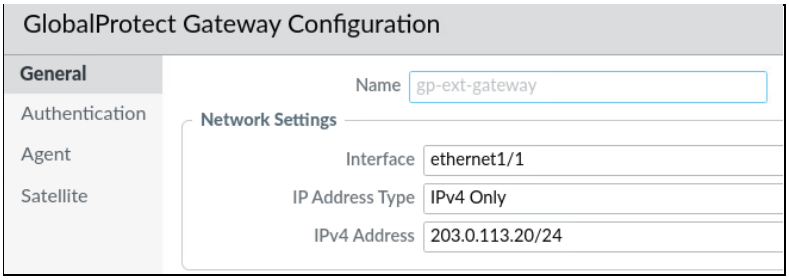
Notice that the gateway is configured to listen on **`ethernet1/3`** but the Traffic logs indicate that the firewall expects IP address 203.0.113.20 to be located on **`ethernet1/1`**.

This is an example of a case in which an accurate and comprehensive knowledge of the network topology and how that topology should match to the core network and zone configurations of the firewall provides an invaluable personal resource for being able to quickly recognize basic misconfigurations.

39. Go to **Network > Zones** to confirm that interface **ethernet1/3** is *not* a member of the **outside** zone.

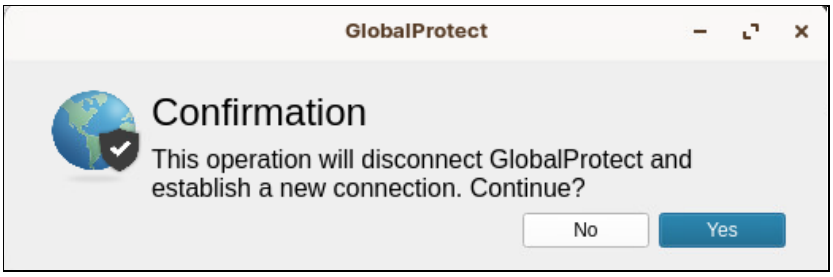
Note: Interface **ethernet1/3** is properly configured as a member of the **dmz** zone.

40. Change the interface assignment for **gp-ext-gateway** to **ethernet1/1** and select the proper interface IP address (203.0.113.20) from the IPv4 Address drop-down list:



41. **Commit** the configuration.
42. In the GlobalProtect window, select the menu button and choose Refresh Connection.

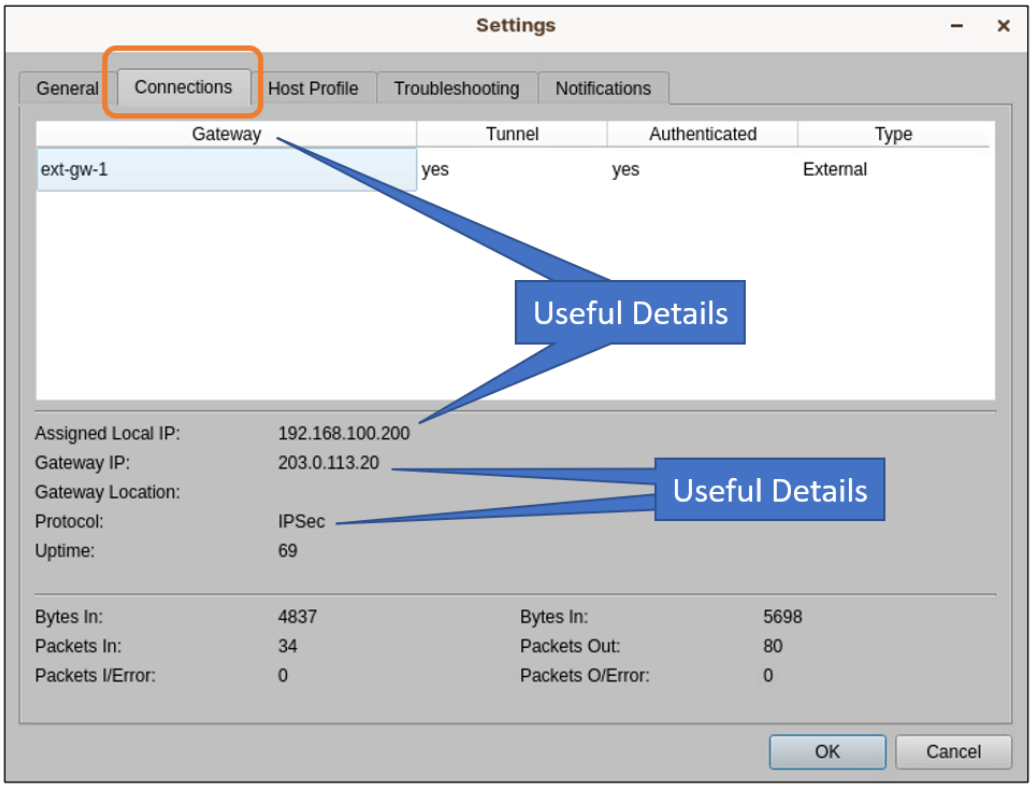
43. In the Confirmation message box, click Yes to establish a new connection:



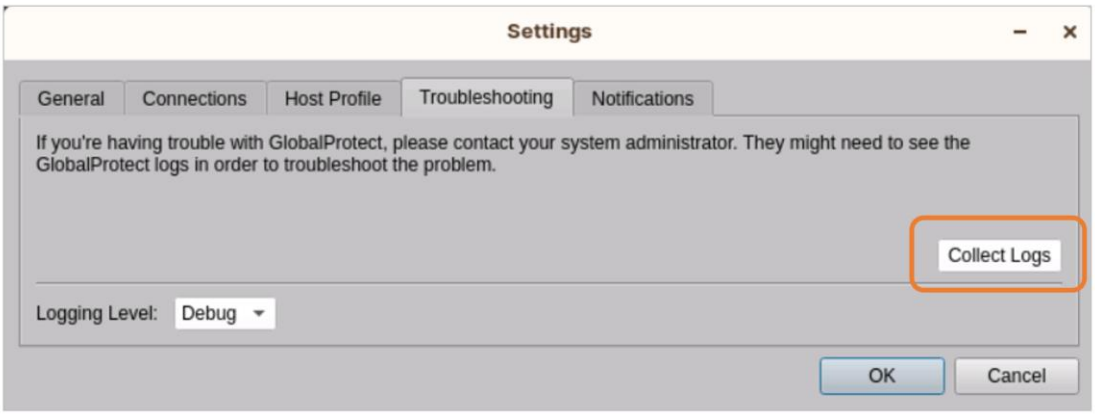
44. The client should now connect.

Note: If you are prompted for credentials, use **lab-user** for username and **Pal0Alt0!** for password.

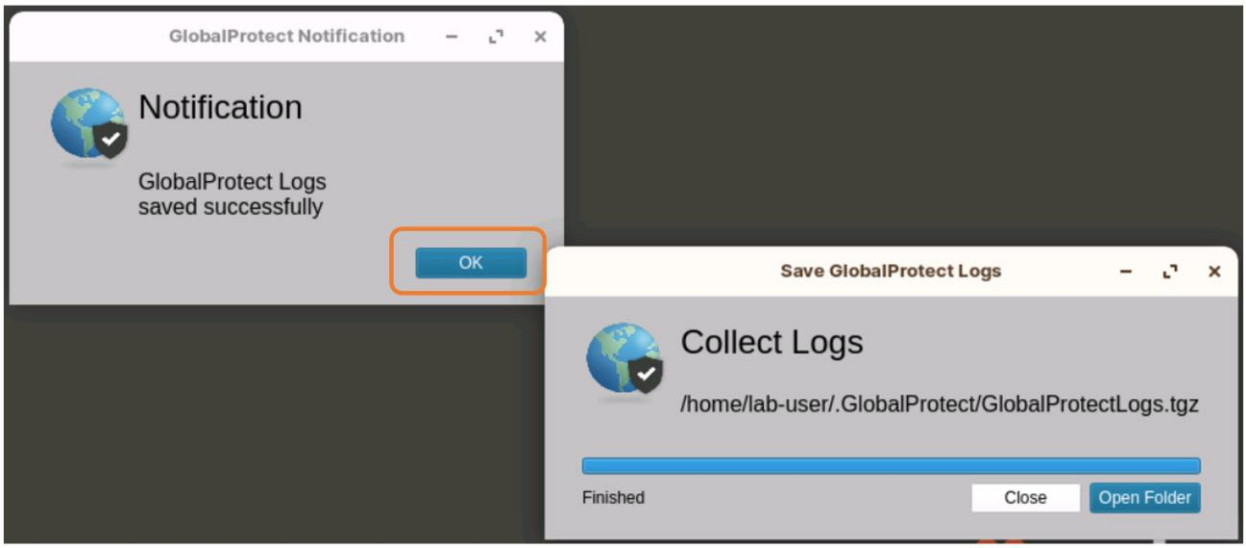
45. In the GlobalProtect application window, click the menu button and choose Settings.
46. From the Settings window that appears, choose the Connections tab.
47. Note the details available in this window:
    - Gateway IP Address
    - Assigned Local IP
    - Protocol



48. In the Settings window, click the tab for Troubleshooting.
49. Click the button for Collect Logs:

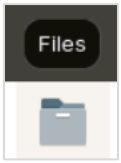


50. You will see two notifications:

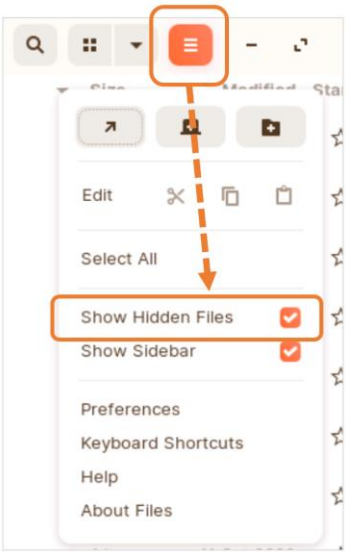


51. Click **OK** on the Notification window but leave the Collect Logs window open.
52. In the window for Collect Logs, click the **Close** button.

53. Open the **Files** application from the menu bar (or from the Application button under **Accessories > Files**).



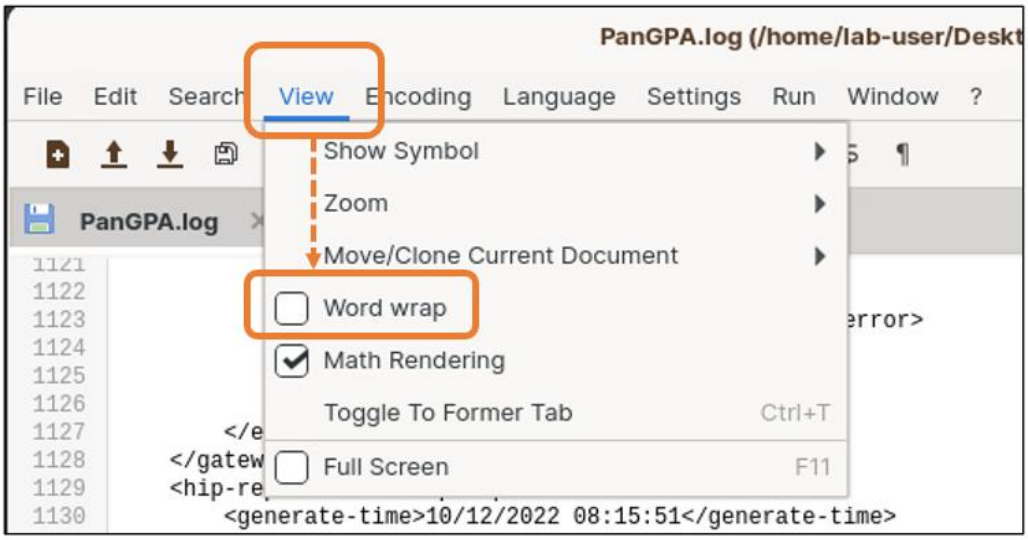
54. In the upper right corner of the Files window, click the ‘hamburger’ button and place a check in the box for **Show Hidden Files**.



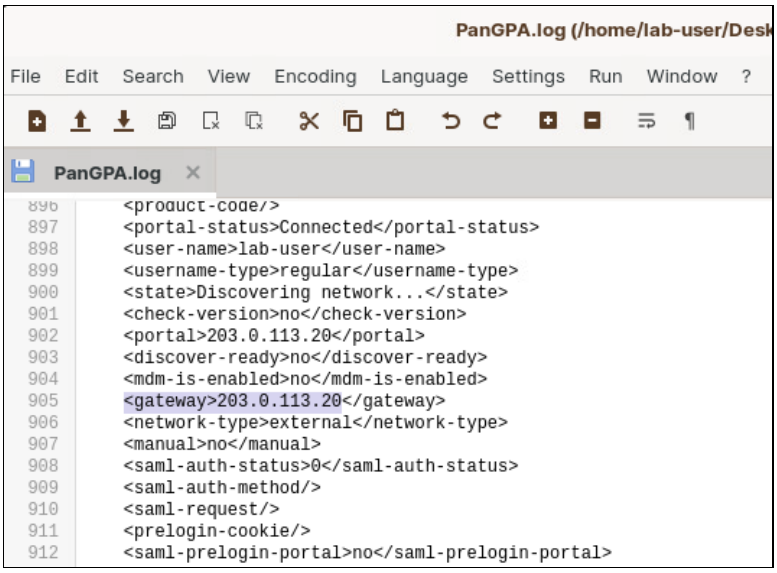
55. In the list of folders, double click the entry for **.GlobalProtect**.
56. Double-click **PanGPA.log** to open it in Notepadqq.
57. Survey some of the information that is available.

    What information might have proved useful in troubleshooting these connectivity issues in your current environment?

58. In Notepadqq, select **View** and deselect **Word wrap** (it may already be unchecked).



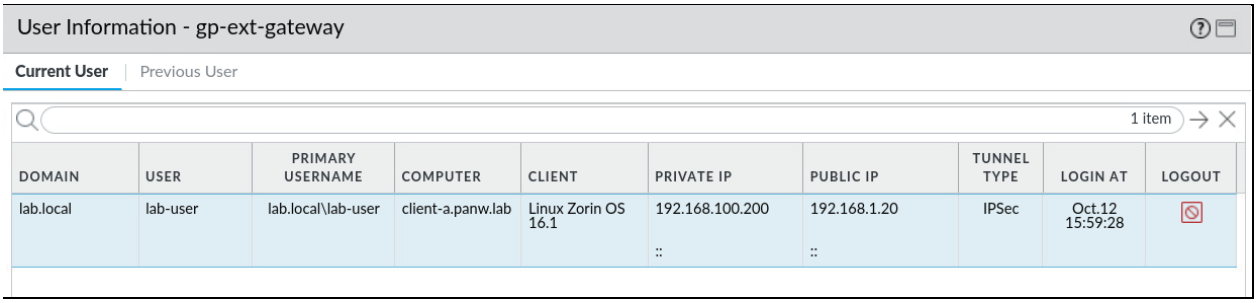
59. Use **CTRL+F** to open the search dialog box and look for **`<gateway>203.0.113.20`**



60. Close Notepadqq.
61. Close the .GlobalProtect folder.
62. Click **OK** to close the Settings window of GlobalProtect.

10.5 Disconnect the Connected User

63. In the firewall web interface, go to **Network > GlobalProtect > Gateways**.
64. Click **Remote Users** to the far right of **gp-ext-gateway**:



65. Under the Logout column, click  to disconnect **lab-user**.
66. Click **Close**.

Locate Information about the Client

In the following steps, you will use the GlobalProtect logs to find information about the client.

67. In the firewall web interface, click **Monitor > Logs > GlobalProtect**.
68. Click the **Stage** type of the first log displayed in the table, and then in the search bar replace the type selected with "**`login`**" and press **Enter**:

(stage eq login)

RECEIVE TIME	PORTAL/GATE...	STATUS	STAGE	EVENT	SOURCE USER	HOST NAME	PUBLIC IPV4	AUTH METHOD
10/12 17:31:06	gp-ext-gateway	success	login	gateway-auth	lab-user	client-a.panw.lab	192.168.1.20	ldap
10/12 17:29:49	gp-ext-gateway	success	login	gateway-register	lab.local\lab-user	client-a.panw.lab	192.168.1.20	
10/12 17:29:49	gp-ext-gateway	success	login	gateway-auth	lab-user	client-a.panw.lab	192.168.1.20	ldap
10/12 17:29:48	gp-portal	success	login	portal-auth	lab-user	client-a.panw.lab	192.168.1.20	ldap

Review the various event types. Notice that GlobalProtect users get authenticated by *both* the portal and the gateway.

Stop. This is the end of the lab.

Bonus Lab

In this lab, you will create a new API certificate on the firewall. This certificate can then be used to eliminate the API KeyGen warning message you receive when committing a configuration.

Lab Objectives

- Modify the firewall Authentication Settings to use a new API Key Certificate

Detailed Lab Steps

Apply a Baseline configuration to the Firewall

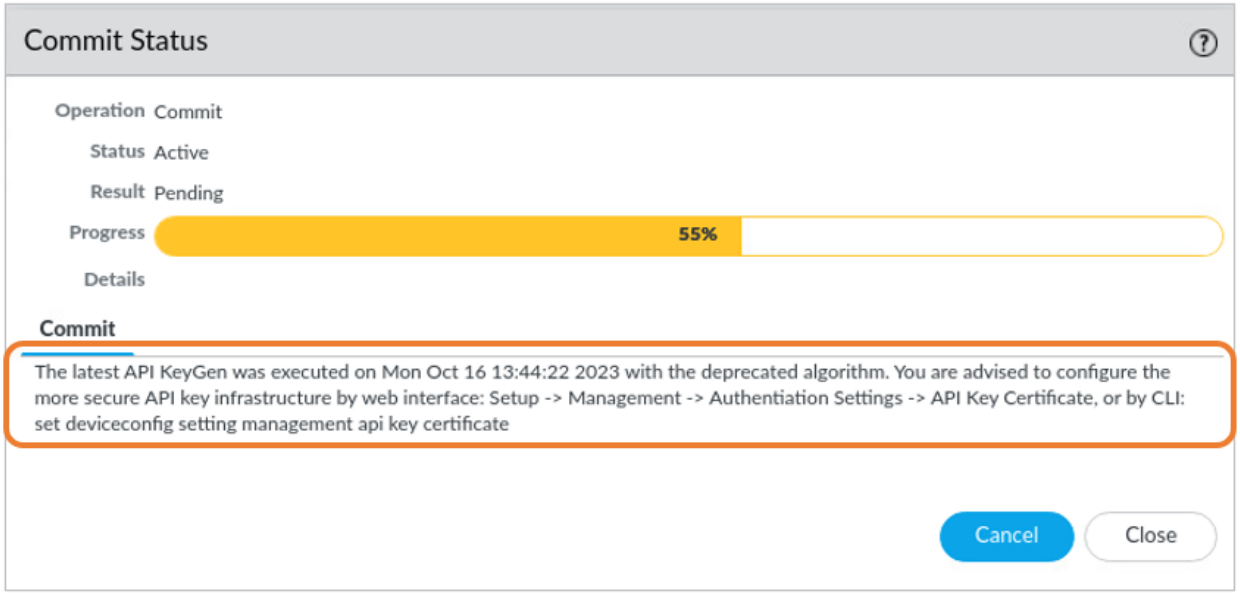
To start this lab exercise, load a preconfigured firewall configuration file.

1. Open the configuration browser and connect to firewall-a.
2. In the Palo Alto Networks firewall web interface, select **Device** > **Setup** > **Operations**.
3. Click **Load named configuration snapshot**.
4. Click the drop-down list next to the **Name** text box and select **`edu-210-11.1a-Capstone-end.xml`**.

This file is used as part of the EDU-210 class on the firewall. This configuration will let you work on the firewall to fix the API KeyGen warning message.

5. Click **OK**.

   A window should open that confirms that the configuration is being loaded.
6. Click **Close**.
7. Click the **Commit** link at the upper right of the web interface:
8. Click **Commit** again and wait until the commit process is complete.
9. Note the error message you receive regarding the API KeyGen algorithm:

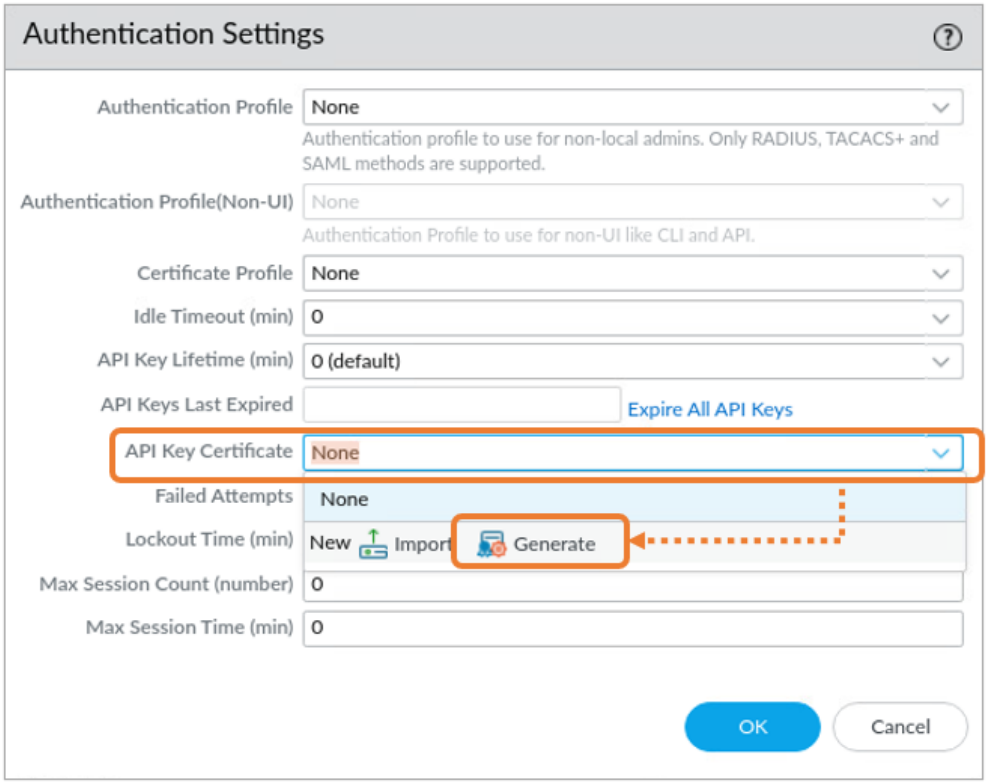


10. Click **Close** to continue.

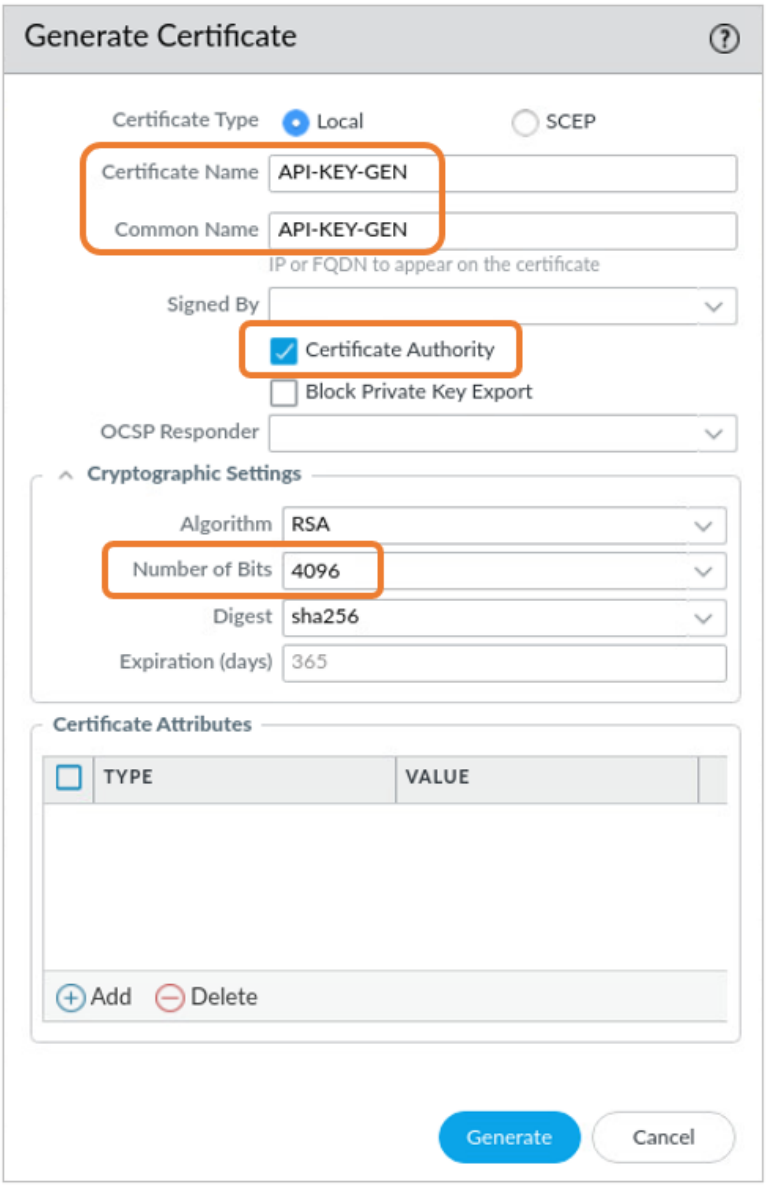
Modify Authentication Settings

In this section, you will create a certificate that the firewall will use to generate API Keys. Doing so will remove the error message you see when you commit a configuration. With this certificate in place, you will not see the error message when committing any configuration files you save from this point forward. If you load an older configuration file (one you created before applying the API Key certificate), you will receive the error message.

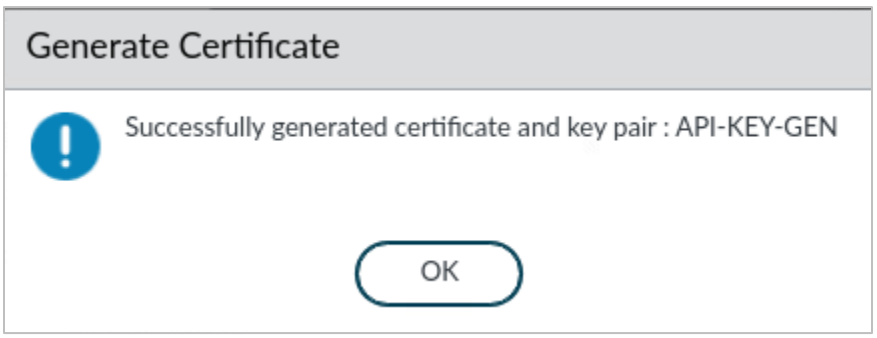
11. Go to **Device > Setup > Management**.
12. Scroll down and locate the section for **Authentication Settings**.
13. Click the gear icon to edit this section.
14. In the field labeled **API Key Certificate**, use the dropdown box to select **Generate**.



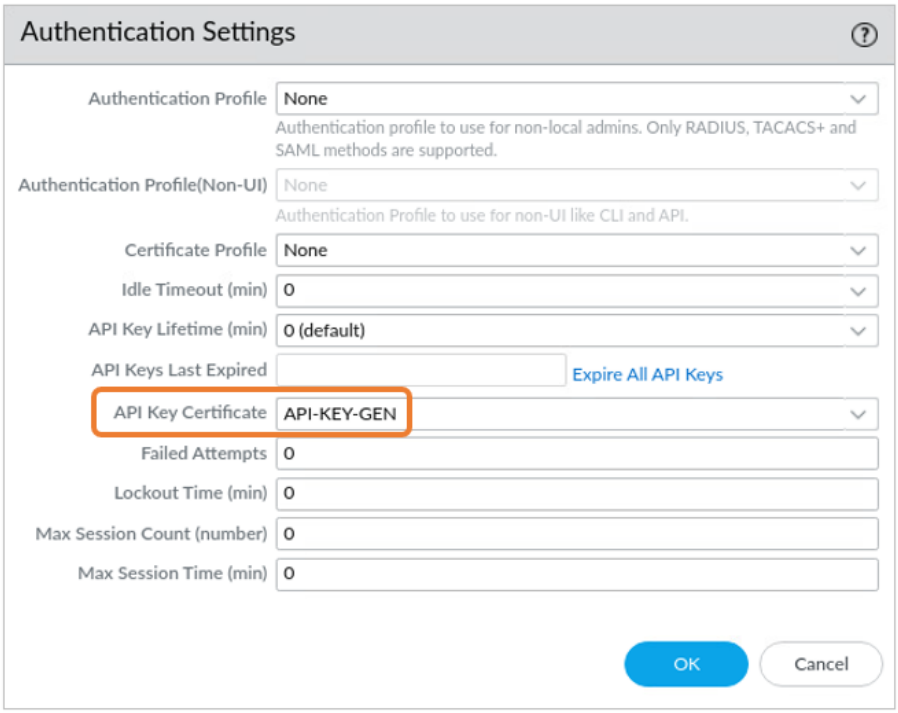
15. In the **Generate Certificate** window, enter **`API-KEY-GEN`** for **Certificate Name**.
16. For **Common name**, also enter **`API-KEY-GEN`**.
17. Check the box for **Certificate Authority**.
18. Under **Cryptographic Settings**, change the **Number of Bits** to **`4096`**.
19. Leave the remaining settings unchanged.



20. Click **Generate**.
21. Click **OK** on the **Generate Certificate** message box.



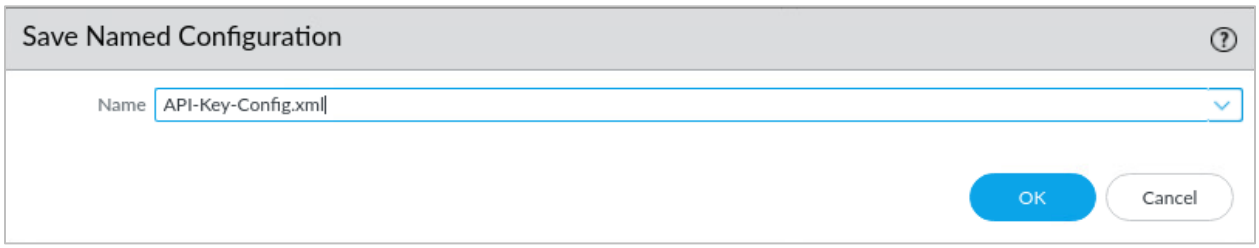
22. Your Authentication Settings window should now display the API-KEY-GEN certificate in the API Key Certificate field.



23. Leave the remaining settings unchanged.
24. Click **OK** to close the **Authentication Settings** window.
25. Click **Yes** to close the **API Key Certificate Change** window
26. Click the **Commit** link at the upper right of the web interface:
27. Click **Commit** again and wait until the commit process is complete.
28. Click **Close** to continue.

Save the Configuration

29. Under **Device > Setup > Operations > Configuration Management**, click **Save named configuration snapshot**.
30. In the **Save Named Configuration** window, enter **`API-Key-Config.xml`** for **Name**.

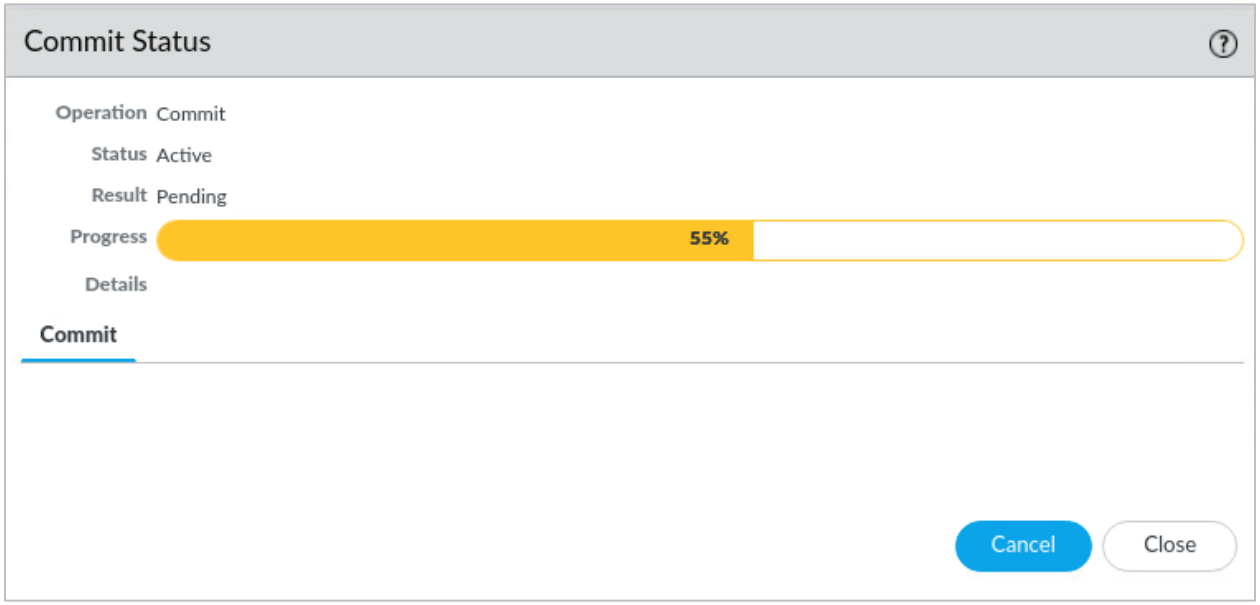


31. Click **OK**.
32. Click **Close** on the **Save Named Configuration** message window.

33. Under **Device** > **Setup** > **Operations**, click **Load named configuration snapshot**.
34. For **Name**, use the drop-down list to select the **`API-Key-Config.xml`** file.
35. Click **OK** to close the **Load Named Configuration** window.
36. Click **Close** to close the **Loading Configuration** message box.

Commit Your Changes and Verify Fix

37. Click the **Commit** link at the upper right of the web interface:
38. Click **Commit** again and wait until the commit process is complete.
39. Note that you no longer receive any error messages regarding the API KeyGen algorithm.



40. Click **Close** when the Commit Status is complete.