



# ClearPass Policy Manager 6.x

## Tech Note: ClearPass

## Palo Alto Networks Integration with CPPM

---

<b><u>Version</u></b>	<b><u>Date</u></b>	<b><u>Modified By</u></b>	<b><u>Comments</u></b>
1.0	May 2013	Danny Jump	Initial Integration Guide V1
2.0	June 2013	Danny Jump	Minor updates for CPPM 6.1
3.0	Sept 2013	Danny Jump	Updates to support CPPM 6.2, changes to post_auth and Troubleshooting section
4.0	Feb 2014	Danny Jump	Updates to support CPPM 6.3, changes to post_auth and details on our HIP support
5.0	May 2015	Danny Jump	Updates to support changes in CPPM 6.5

Overview .....	6
Why is this Integration Important? .....	6
The Challenge .....	6
Background .....	7
Next-Generation Solution .....	7
Software Requirements .....	8
ClearPass Configuration .....	8
What's new in CPPM 6.5? .....	8
CPPM Basic Configuration - All CPPM Versions .....	9
CPPM Basic Configuration - Insight .....	9
CPPM Basic Configuration - Interim Accounting .....	10
NAS/NAD Basic Configuration – Interim Accounting .....	10
CPPM Basic Configuration – Post_authentication processing .....	12
post_authentication configuration for CPPM 6.5/6.4/6.3 .....	13
Real-Time Framework introduced post CPPM 6.3 .....	13
post_authentication configuration for CPPM 6.1 and 6.2 .....	13
Adding PANW Firewall & Panorama context servers endpoints .....	15
Adding PANW Context Servers in CPPM 6.5 & 6.4 .....	15
Adding Palo Alto Networks Panorama Context Server Endpoint .....	17
Summary of Shared Context Attributes .....	18
Triggering Updates from ClearPass to a PANW endpoint post 6.5 .....	19
Triggering Updates from ClearPass to a PANW endpoint pre 6.5 .....	20
Adding an Enforcement Profile to a Enforcement Policy .....	21
Adding an Enforcement Policy to a Service Policy .....	21
Sending Health/Posture status to PANW from CPPM .....	22

Configuring OnGuard on ClearPass .....	22
Configuring PANW to use Health/Posture context.....	23
Configuring TAGS for Health/Posture .....	23
Setting the CPPM Posture/Health Delay Timer.....	25
Configuring Palo Alto Networks Next-Generation Firewall.....	26
Configuring CPPM to communicate just using the XMLAPI .....	26
Configuring a Policy on PANW to use CPPM context data – generic info.....	27
Creating Device Profile Categories.....	28
Configuring Palo Alto Networks PAN-OS 6.x - Tags and HIP Objects .....	29
Other Attributes from HIP Object/General explained.....	31
Configuring Palo Alto Networks PAN-OS 5.x - Dynamic-Objects .....	32
PAN-OS 6.x Changes to DAO Limits .....	34
Faultfinding Tips (PANOS cli cmds/CPPM Logs).....	35
UserID <-> IP Address Mapping (PAN-OS 5.x & 6.x cmd) .....	35
Dynamic Device (Tag) <-> IP Address Mapping (PAN-OS 5.x & 6.x cmd).....	36
Show HIP Reports.....	37
Show XMLAPI statistics.....	37
Active real-time debug monitoring of the UserID process .....	38
Check Logs files in CPPM.....	39
Sending login UserID + Source IP@, as user logs in .....	41
Adding IP@ to Category, as CPPM profiles the IP@ .....	41
Sending logoff UserID + IP@, as user logouts .....	41
Removing IP@ from Category as device logout .....	41
XML example of HIP Object.....	42
Conclusion .....	42



Figure 1 - ClearPass and Palo Alto Networks Integration Overview .....	7
Figure 2 - Checking Insight DB is enabled .....	9
Figure 3 - Checking RADIUS Interim-Accounting is enabled on CPPM .....	10
Figure 4 - Enable RADIUS Interim accounting on Aruba Controller .....	11
Figure 5 - Configuring RADIUS authentication on Cisco WLC .....	11
Figure 6 - Configuring RADIUS accounting on Cisco WLC .....	11
Figure 7 - Post Authentication run-times across different CPPM versions .....	12
Figure 8 - Modifying the Post-authentication daemon sleep-time under CPPM 6.5/6.4/6.3 .....	13
Figure 9 - Modifying the post_authentication daemon sleep-time under CPPM 6.1.x .....	13
Figure 10 - Modifying the post_authentication daemon sleep-timer under CPPM 6.2.x .....	14
Figure 11 - Adding a Palo Alto Networks Firewall in CPPM 6.5 & 6.4 .....	15
Figure 12 - Appending DOMAIN/Full-username .....	16
Figure 13 - Adding Palo Alto Networks Panorama endpoint .....	17
Figure 14 - Attributes we can share with Palo Alto Networks endpoints .....	18
Figure 15 - Adding an enforcement-profile for PANW in CPPM 6.5 .....	19
Figure 16 - Adding a Session Restriction Enforcement profile .....	20
Figure 17 - Adding a Session-Check .... one endpoint per profile .....	20
Figure 18 - Trigger PANW update on AD memberOf .....	21
Figure 19 - PANW enforcement profile added to a service policy .....	21
Figure 20 - Configuring TAGS on PANW .....	23
Figure 21 - Copy of "Not_Healthy" TAGS .....	23
Figure 22 - Creating an Address-Group to match on ANYTHING unhealthy .....	24
Figure 23 - Creating different Address-groups to check on individual failures .....	24
Figure 24 - Adding an Address-group to and firewall policy .....	25
Figure 25 - Setting Eager handler to 120 seconds when sending posture/health .....	25
Figure 26 - Creating an restricted Admin-Role .....	26
Figure 27 - Adding a User to Palo Alto Networks Firewall .....	27
Figure 28 - CPPM Fingerprints .....	28
Figure 29 - Adding a TAG under PAN-OS 6.x .....	29
Figure 30 - Grouping Tags into a Dynamic Address Group .....	30
Figure 31 - Creating HIP Objects .....	30
Figure 32 - CPPM Fingerprints - Client Version .....	31
Figure 33 - Utilizing Tags in a Firewall Rule .....	31
Figure 34 - Configuring Dynamic Objects under PAN-OS 5.x .....	32
Figure 35 - Palo Alto Networks 'dynamic' objects .....	32
Figure 36 - Basic Firewall Rules .....	33
Figure 37 - Firewall Rule Based Upon a Source-Device-Type of an endpoint .....	33
Figure 38 - Firewall Rule Based Upon a Source of a User Name .....	34
Figure 39 - Signed in User's to their IP Mapping .....	35
Figure 40 - Signed in Users to their IP Mapping and also matched policy hits .....	35
Figure 41 - Dynamic Object Category - IP Address Mapping .....	36
Figure 42 - Logged in user in PAN-OS 6.x .....	36
Figure 43 - HIP Report for a user .....	37
Figure 44 - XMLAPI Stats .....	37
Figure 45 - List of ALL users registered through ID Manager .....	38



<i>Figure 46 - How to collect CPPM Logs – limited data, but includes postautctrl.log .....</i>	<i>39</i>
<i>Figure 47 - Collection of CPPM Logs complete .....</i>	<i>40</i>
<i>Figure 48 - Where to locate postauthctrl.log .....</i>	<i>40</i>

## Overview

This document is intended to help field engineering, customers, and channel partners integrate Aruba Networks ClearPass 6.X with Palo Alto Networks next-generation firewall and its central management system, Panorama. Customers can now leverage the Identity tracking features provided by ClearPass for known enterprise users using Active Directory and LDAP server, and for unknown guest/public user credentials that are used by Guest and HotSpot networks.

 **Note:** Where you see a red-chili  this is to signify a ‘hot’ important point and highlights that this point is to be taken as a best-practice recommendation.

## Why is this Integration Important?

---

Palo Alto Networks next-generation firewall offers contextual security for all users for a number of reasons – especially for safe enablement of applications. Simple firewalling beyond basic IP address or TCP port numbers only provides a subset of the enhanced security required for enterprises to secure their networks.

As an example, it’s no longer acceptable to just ‘deny Twitter’ or ‘deny Facebook’ access. Many organizations use social networking Web sites to advertise their products, solutions, and activities. Social networking has become an accepted marketing tool and many companies now opt to use this as a mainstream part of their marketing efforts. As such, legacy firewalls are not able to differentiate valid authorized users from casual social networking users. So today’s challenge to allow Facebook based upon contextual data such as username makes it almost impossible for legacy firewalls to implement granularity in security policy.

## The Challenge

---

Historically, traditional firewalls make decisions based on Layer3/4 and some Layer7 information. For Web-based traffic, a decision would typically be based upon a domain or a URL string. Today, enterprises want to make decisions based upon the user and associated permissions, and, for this to happen, the firewall needs to correlate between the user and the assigned IP address. The challenge is finding meaningful sources of user information covering the full spectrum of network activity, including known users, guests, and non-enterprise configured users.

## Background

One of the core features of the Palo Alto Networks next-generation firewall is User-ID, which provides many methods for connecting to sources of identity information and associating them with firewall policy rules. For example, it has an option to gather user information from an Active Directory or LDAP server. In the past, this functionality required the use of a Palo Alto Networks User-ID agent running on a Windows workstation.

Similarly, an agent can be used to allow integration with a legacy Amigopod deployment to gather user information for the guest users. This integration allowed Amigopod to send user information to a Palo Alto Networks firewall via the User-ID agent running on a Windows workstation. In both scenarios above, the past approaches required an agent, which created dependencies that might not be easy to resolve in certain deployment scenarios. Now you can take advantage of the Palo Alto Networks PAN-OS and Aruba Networks ClearPass Policy Manager, making a more seamless integration possible.

## Next-Generation Solution

Starting with the release of ClearPass Policy Manager 6.1, Aruba re-architected the integration between ClearPass and the Palo Alto Networks next-generation firewall to take advantage of the new XML APIs that were available in the PAN OS 5.x code release. This simplified the solution significantly by making it more efficient and streamlined. The requirement to download and configure a separate plug-in was eliminated and instead the solution was fully integrated into the ClearPass core product.

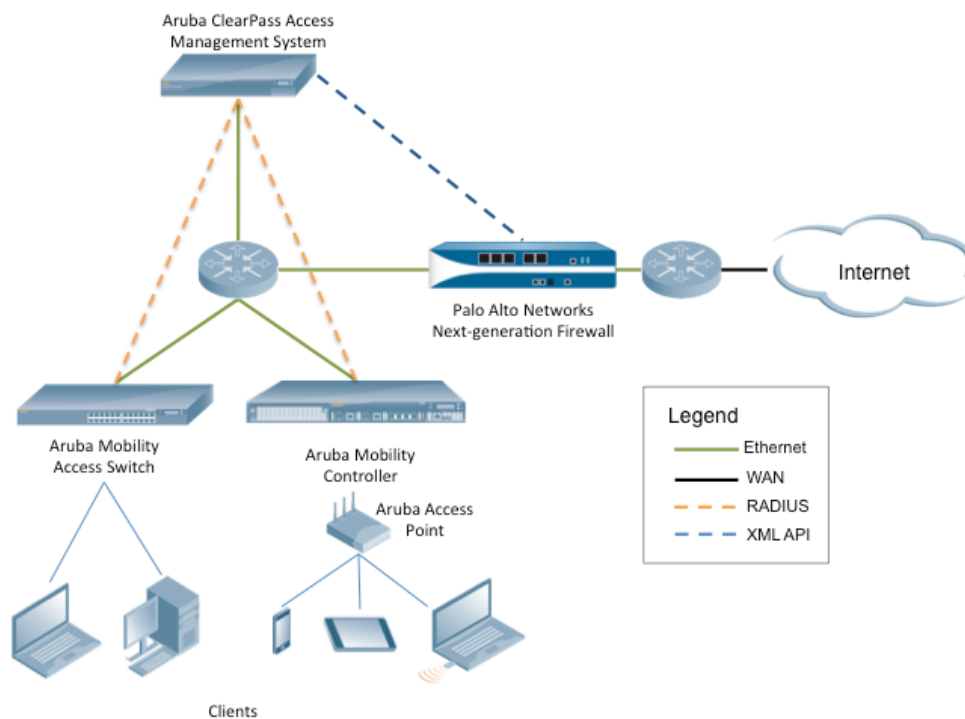


Figure 1 - ClearPass and Palo Alto Networks Integration Overview

## Software Requirements

The minimum software version required on ClearPass Policy Manager is 6.1.0, released in April 2013. The minimum software version on the Palo Alto Networks firewall is PAN-OS 5.0.0, released in November 2012.

However, it is recommended that you regularly review software updates to utilize the benefits from the latest fixes and feature updates from Aruba and Palo Alto Networks.

## ClearPass Configuration

---

Configuring ClearPass Policy Manager for Palo Alto Networks firewall integration is a fairly simple straightforward process. Step-by-step instructions are outlined in the following sections. The configuration has been separated into several sections. The first being to highlight the new functionality in CPPM 6.5, then several sections covering the basics for multiple releases and then some of the nuances for the older ClearPass releases.

The enhanced ClearPass Policy Manager Exchange framework added in 6.5 allows us to enhance the integration with additional 3<sup>rd</sup> party vendors. This allows us to push the endpoint source IP address, username and other attributes to other 3<sup>rd</sup> party firewalls (e.g. Checkpoint, Fortinet, and iBoss).

This information can be located in the TechNote section on the [Aruba Support Site](#).



### What's new in CPPM 6.5?

Within the latest release of ClearPass, version 6.5 released in March 2015, we added some new features and tweaked a couple of older features to improve their function.

Firstly, we added the ability for Policy Manager, when it's aware of posture/health for an endpoint, to now share this context. CPPM gathers different health class information from our OnGuard client, context such as the state of the endpoint firewall (enabled/disabled, engine version), derives a posture state, and then return a healthy/un-healthy state per class back to the PANW firewall. There are 10 classes we can report against and they are covered later in the document.

The other item of notable reference is the reduction of the post\_authentication daemon sleep-timer to 3-seconds. To aids in making the update between CPPM and the PANW more real-time.

## CPPM Basic Configuration - All CPPM Versions

### CPPM Basic Configuration - Insight

Before we commence the configuration of the Palo Alto Networks services/profiles, etc., we need to ensure that some of the basic configuration items are covered. From CPPM 6.1.x and all later releases the Insight Database is disabled by default. This must be enabled on CPPM on a single box or somewhere in the cluster for the Palo Alto Networks integration to function.

Check under **Administration > Server Manager > Server Configuration > System** and if the **'Enable Insight'** is not enabled, enable as appropriate.

Administration > Server Manager > Server Configuration - cppm-prod-vm1

### Server Configuration - cppm-prod-vm1 (10.2.100.225)

System			Services Control	Service Parameters	System Monitoring	Network
Hostname:	cppm-prod-vm1					
Policy Manager Zone:	default					
Enable Profile:	<input checked="" type="checkbox"/> Enable to allow this server to perform endpoint classification					
Enable Insight:	<input checked="" type="checkbox"/> Enable Insight on this server					
			Management Port:		Data/External Port:	
IP Address:	10.2.100.225					
Subnet Mask:	255.255.255.0					
Default Gateway:	10.2.100.1					
DNS Settings:			Primary		Secondary	
IP Address:	10.2.100.120					

Figure 2 - Checking Insight DB is enabled

**Why we need INSIGHT** - The Insight Application must be running. It is used to collate the records that make up the XML API we send to the Palo Alto Networks Endpoint. The RADIUS Authentication triggers a NetEvent from which we write data into the Insight DB. When Insight receives the RADIUS Accounting data (again from a NetEvent) we match this data with the MAC address to update the SRC IP address in Insight. Having the IP address we are then able to fingerprint the device and obtain the endpoint device-type etc.

## CPPM Basic Configuration - Interim Accounting

Next, we have to ensure that CPPM is logging the RADIUS Interim-Accounting Updates. This can be checked at the following **Administration > Server Manager > Server Configuration > Service Parameters** as shown below. Note that the default is **FALSE**. Ensure you have it configured as **TRUE** as shown below. Scroll to the bottom to see the 'Accounting' drop-down-box.

Administration » Server Manager » Server Configuration - cppm-prod-vm1

Server Configuration - cppm-prod-vm1 (10.2.100.225)

System	Services Control	Service Parameters	System Monitoring	Network
Process Server-Status Request				
		FALSE	FALSE	
<b>Main</b>				
Authentication Port	1812 , 1645	1812, 1645		
Accounting Port	1813 , 1646	1813, 1646		
Maximum Request Time	30 seconds	30	5-120	
Cleanup Time	5 seconds	5	2-10	
Local DB Authentication Source Connection Count	32	32	5-150	
AD/LDAP Authentication Source Connection Count	64	64	5-300	
SQL DB Authentication Source Connection Count	32	32	5-100	
EAP-TLS Fragment Size	1024 bytes	1024	512-1500	
Use Inner Identity in Access-Accept Reply	FALSE	FALSE		
Reject if OCSP response does not have Nonce	TRUE	TRUE		
Check the validity of all certificates in the chain against CRLs	TRUE	TRUE		
TLS Session Cache Limit	3750 sessions	3750	1000-100000	
<b>Thread Pool</b>				
Maximum Number of Threads	20 threads	20	10-300	
Number of Initial Threads	10 threads	10	10-300	
<b>AD Errors</b>				
Window Size	5 minutes	5	1-60	
Number of Errors	150	150	10-1000	
Recovery Action	None	None		
<b>EAP-FAST</b>				
Master Key Expire Time	1 weeks	1 weeks		
Master Key Grace Time	3 weeks	3 weeks		
PACs are valid across cluster	true	true		
<b>Accounting</b>				
Log Accounting Interim-Update Packets	TRUE	FALSE		

Figure 3 - Checking RADIUS Interim-Accounting is enabled on CPPM

## NAS/NAD Basic Configuration – Interim Accounting

Ensure RADIUS interim accounting is also enabled on the NAS device. Also important to ensure that the **calling-station-ID** is set to use the MAC address of the client (this is the default for Aruba controller). If the NAS device is configured to use the system IP address, we will not be able to collate the data correctly within Insight, and thus will be unable to send the correct data to the Palo Alto Networks Firewall or Panorama system. **Note:** Cisco uses the system IP address by default, ensure this is changed as shown below.

For **Aruba controllers**, enable RADIUS Interim accounting as shown below....

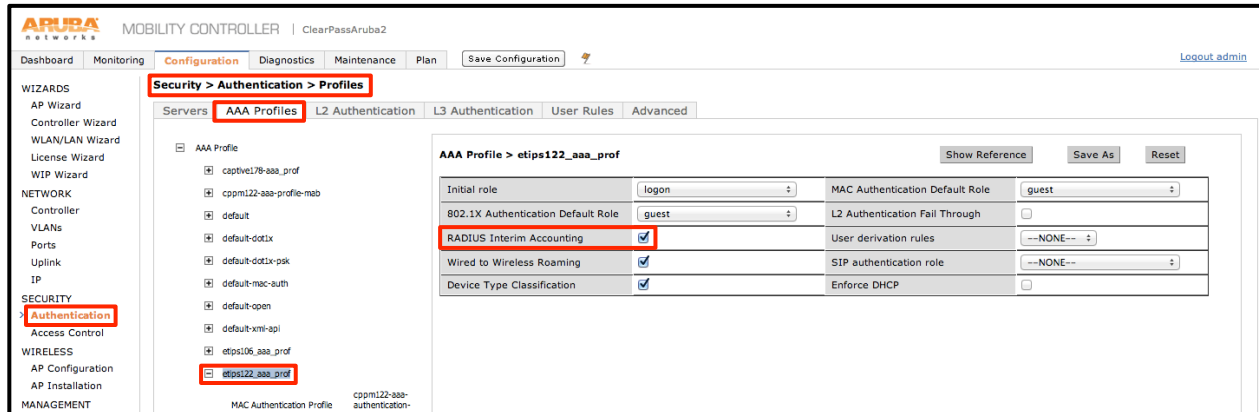


Figure 4 - Enable RADIUS Interim accounting on Aruba Controller

For **Cisco controllers**, ensure RADIUS Authentication and RADIUS Accounting are configured as shown below.... taking special notice that the **Call Station ID Type** is set to **System MAC Address**.

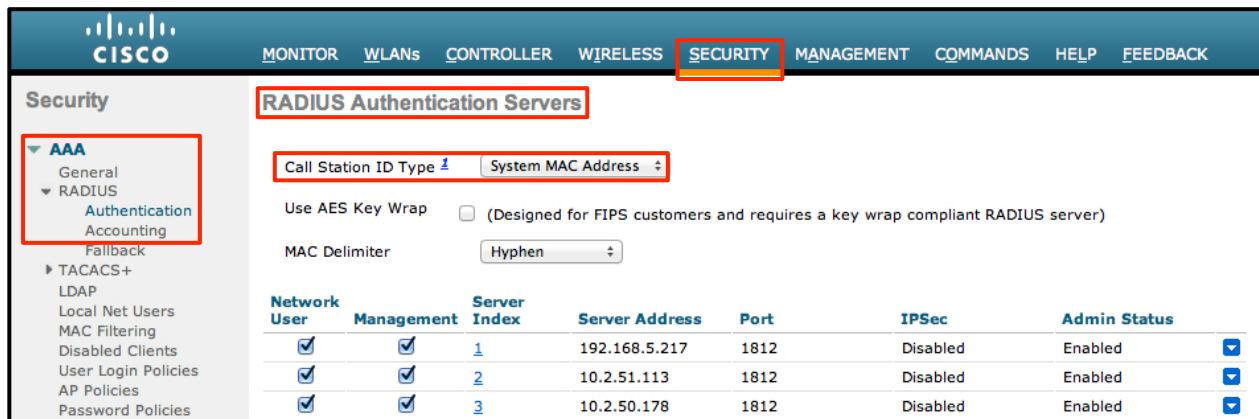


Figure 5 - Configuring RADIUS authentication on Cisco WLC



Figure 6 - Configuring RADIUS accounting on Cisco WLC

## CPPM Basic Configuration – Post\_authentication processing

The data that CPPM collates and writes to the Insight DB is extracted and written to the Palo Alto Networks Firewall or Panorama endpoint by the post\_authentication daemon. This daemon runs at periodic times as shown below.

The version of CPPM deployed will dictate the frequency this daemon runs and how this can be adjusted. See below to understand the run-time/sleep frequency values of this daemon.

Note that there are multiple stages from a client/endpoint associating to an AP or bringing up the carrier on an Ethernet port to the information being posted to the Palo Alto Firewall.

Several seconds can elapse before the client has authenticated and obtained its IP address and the NAS has sent RADIUS Accounting packets to ClearPass (we need this for the client IP address). Assuming that Profiling is enabled ClearPass then will profile the endpoint. Following these steps ClearPass has all of the attributes it needs to be able to update the Palo Alto Networks endpoint. The process to gather all the contextual data into a format that we send has been streamlined over several releases. Once the data is gathered there is a batch process which POST's this data to the PANW. This batch process is called the post\_auth daemon that is discussed below and on some of the following pages.

The below table shows the settings of the post\_auth daemon. Note that in CPPM 6.2 we split the post\_auth daemon into the lazy and eager handler.

CPPM Version	Max / Min / Default Values	Recommended Value	Expected delay in endpoint appearing in PANW
6.1.x	10 mins / 3 mins / 5 mins	3 mins	2-3 mins
6.2.x	300 sec / 10 sec / 30 sec	10 sec **	10-15 seconds
6.3.x	300 sec / 10 sec / 30 sec	10 sec **	10-15 seconds
6.4.x	300 sec / 10 sec / 30 sec	10 sec **	10-15 seconds
6.5	300 sec / <b>3 sec</b> / 30 sec	10 sec **	10-15 seconds

Figure 7 - Post Authentication run-times across different CPPM versions



\*\* Lowering the Eager handler must be done with care to not effect other system functions.

The entire process of a device associating, authenticating, and getting an IP address, to CPPM profiling the data and then the daemon sending this information to PANW can take a number of seconds. The process to gather all of the data was significantly streamlined in CPPM 6.3 under a process called the real-time-framework. Read more later on the Real-Time Framework on the next page under “Real-Time Framework introduced post CPPM 6.3”.



## post\_authentication configuration for CPPM 6.5/6.4/6.3



CPPM 6.5 introduces some new features related to Palo Alto Networks Integration. Specific to the post\_authentication eager\_handler daemon, we lowered the minimum value to 3 seconds. What this 3-seconds refers to is the interval sleep-timer the daemon will wait until the next processing cycle. It looks for new session details that have been gathered that are complete that ClearPass can post to the PANW firewall (the process to harvest the endpoint info is performed by the Real-Time Framework); it runs, then goes to sleep for this interval, then runs again, etc.

Administration » Server Manager » Server Configuration - cppm161

Server Configuration - cppm161 (10.2.100.161)

System Services Control **Service Parameters** System Monitoring Network FIPS

Select Service: Async network services

Parameter Name	Parameter Value	Default Value	Allowed Values
<b>Post Auth</b>			
Number of request processing threads	20 threads	20	20-100
Lazy handler polling frequency	5 minutes	5	3-10
Eager handler polling frequency	30 seconds	30	3-300
Send Posture Data	TRUE	FALSE	

Figure 8 - Modifying the Post-authentication daemon sleep-time under CPPM 6.5/6.4/6.3

## Real-Time Framework introduced post CPPM 6.3

Post the CPPM 6.3 release we introduced an improved framework that provides **near-real-time** processing for certain functions within CPPM. One of the functions able to take advantage of this is the Palo Alto Networks integration. Previously minor delays could have been experienced between CPPM receiving the initial RADIUS Auth Request and CPPM updating the Palo Alto Networks endpoint with the relevant meta-data: Username, IP address, Device-Type, etc. With the Real-Time Framework and the lowering of the Eager-Handler the updates to the Palo Alto Networks endpoint should arrive in a few seconds.

**Note:** No configuration changes are necessary to benefit from this new framework. All the improvements are transparent to the CPPM administrator and happen 'under-the-covers'.

## post\_authentication configuration for CPPM 6.1 and 6.2

Figure 9 below shows the configuration options for the post\_authentication daemon under **CPPM 6.1.x**. Notice that only a single value is configurable, this is a system wide setting.

Administration » Server Manager » Server Configuration - cppm-prod-vm1

Server Configuration - cppm-prod-vm1 (10.2.100.225)

System Services Control **Service Parameters** System Monitoring Network

Select Service: Async network services

Parameter Name	Parameter Value	Default Value	Allowed Values
<b>Post Auth</b>			
Number of request processing threads	20 threads	20	20-100
Polling frequency	3 minutes	5	3-10

Figure 9 - Modifying the post\_authentication daemon sleep-time under CPPM 6.1.x

Figure 10 below shows the configuration options for the post\_authentication daemon under **CPPM 6.2.x**.

**Note:** It's important to note that back in the 6.2 release we split the post\_authentication processing into two separate processes. The '**eager handler**' is responsible for the Palo-Alto Networks updates, whilst the remaining post\_authentication processing is handled by the '**lazy handler**'.

Administration » Server Manager » Server Configuration - cppm-6.2.0

Server Configuration - cppm-6.2.0 (10.2.100.155)

System	Services Control	Service Parameters	System Monitoring	Network
Select Service: Async network services				
Parameter Name	Parameter Value	Default Value	Allowed Values	
<b>Post Auth</b>				
Number of request processing threads	20 threads	20	20-100	
Lazy handler polling frequency	5 minutes	5	3-10	
Eager handler polling frequency	30 seconds	30	10-300	

Figure 10 - Modifying the post\_authentication daemon sleep-timer under CPPM 6.2.x

**Note:** CPPM 6.2 lowers the post\_authentication processing delay of the Palo Alto Networks updates. Whilst it can be set as low as 10 seconds, the default of 30 seconds remains the recommendation under CPPM 6.2. Care should be taken when lowering this value as system resources for this daemon may need to be carefully managed.

## Adding PANW Firewall & Panorama context servers endpoints

Some minor differences exist in the GUI Context server configuration depending on the version of ClearPass in use. These relate to Aruba enhancing the integration capabilities between ClearPass Policy Manager and Palo Alto Networks firewalls.

### Adding PANW Context Servers in CPPM 6.5 & 6.4

Under **Administration > External Server > Endpoint Context Servers > Add Context Server > [choose] Palo Alto Networks Firewall**, then enter the required IP address of the Palo Alto Networks Firewall, and a username/password pair that ClearPass will use to send context endpoint data to the Palo Alto Networks Firewall.

Administration » External Servers » Endpoint Context Servers

### Endpoint Context Servers

#### Modify Endpoint Context Server

**Server**

Server Type:	Palo Alto Networks Firewall		
Server Name:	10.2.100.10		
Server Base URL:	https://{server_ip}/api/?type=keygen&user={username}&password={password}		
Username:	cppm-api		
Password:	.....	Verify:	.....
Username Transformation:	Prefix NETBIOS name		
GlobalProtect:	<input checked="" type="checkbox"/> GlobalProtect Enabled on Palo Alto Networks Firewall		
UserID Post URL:	https://{server_ip}/api/?type=user-id&action=set&key={key}&cmd={cmd}		
Validate Server:	<input type="checkbox"/> Enable to validate the server certificate		

Figure 11 - Adding a Palo Alto Networks Firewall in CPPM 6.5 & 6.4

We discuss configuring the username on the Palo Alto used above in a later section, *"Configuring Palo Alto Networks Next-Generation Firewall"*.

**Note:** Do **not** change the Server Base URL or UserID Post URL. Although the fields can be modified, they are specifically formatted to work with a Palo Alto Networks firewall running PAN-OS 5.x or 6.x software.

**Note:** On earlier version of CPPM 6.2/6.1 the option to select GlobalProtect did not exist. Enabling **GlobalProtect** on CPPM signifies that CPPM can send Host Information Profile (HIP) Objects to the Palo Alto Networks endpoint to allow it to apply enforcement policies based upon the HIP context attributes received for a user/endpoint; this is in addition to the more basic UserID XML API attributes. HIP data allows us to send more granular context about the endpoint. As an example, when sending context about the endpoint with

the UserID XMLAPI we might inform the PANW the endpoint is a SmartDevice. With HIP data we could actually inform the PANW the endpoint is a Apple iPad running IOS 7, so a significantly more granular level of data.

Also starting with CPPM 6.4 we improved the ability of CPPM to send/append the domain level data for a user when they authenticate. In the scenario where a user authentication is not in the format DOMAIN\username and we just receive a username, we can now configure CPPM to append the DOMAIN to the username. CPPM 6.3 or less had an option to **send Full Username in the UID updates** to PANW. The Full Username prefix can include the user domain (NETBIOS name as in case of Active Directories) or an LDAP domain (less frequent as in case of OpenLDAP or a non-AD source).

In 6.4 this option is deprecated and replaced with **Prefix NETBIOS name**. This addresses a portion of the problem when a domain prefix is available if the user is authenticated against an AD.

When **None** is selected – UID updates will have entry as username; HIP Report also will have only the username.

When **Prefix NetBIOS Name** is selected – UID updates will have entry as NetBIOS Name\username; HIP Report will also have NetBIOS Name\username, will also include NetBIOS Name as domain field.

When **Use Full Username is selected** – UID updates will have entry as Full Username propagated by policy server, if authentication is against AD it will be similar to above NetBIOS Name\username, else Some Other Domain\username in case of say Guest Captive Portal or Username@somedomain which is not accepted by Palo Alto, HIP Report will have same entry, but the domain field will not be sent.

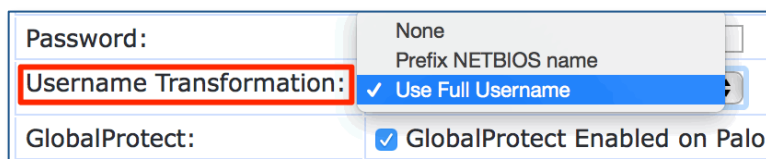


Figure 12 - Appending DOMAIN/Full-username

**Note:** CPPM normalizes the formatting sent to the PANW endpoint as **domain\username**. If we receive an incoming ID that reads “danny@arubanetworks.com”, we’ll send ARUBANETWORKS.COM\danny as the payload. If we receive ARUBANETWORKS.COM\danny, we’ll send ARUBANETWORKS.COM\danny as the payload to PANW endpoint.

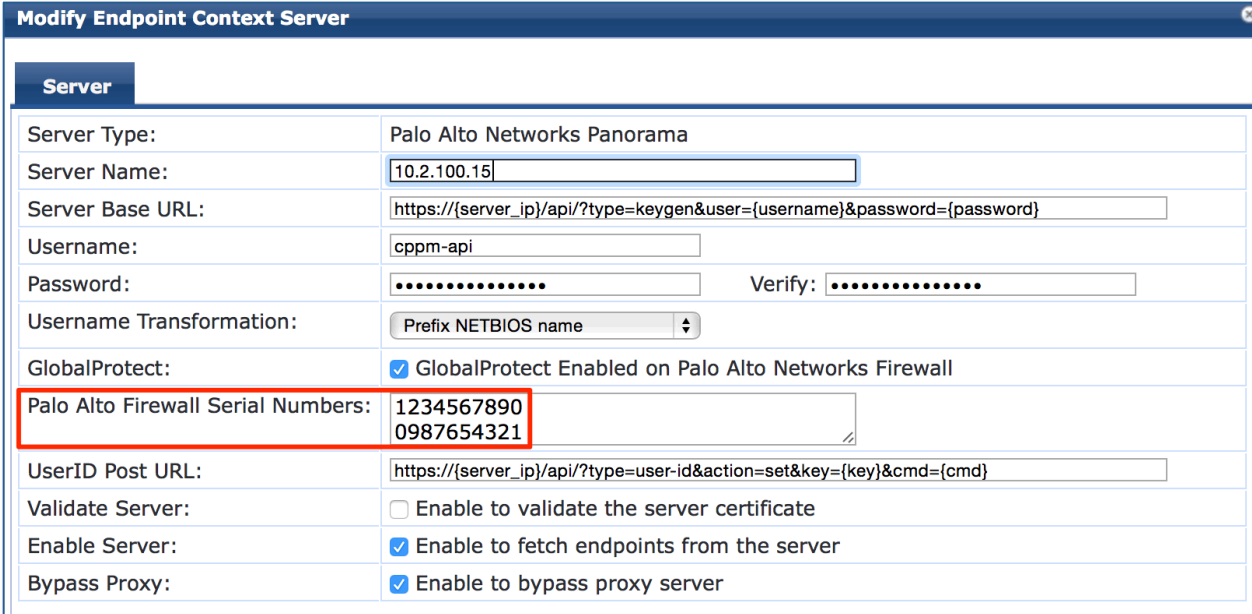


The Palo Alto Networks firewall can only accept the UserID in the following format **domain\username**, policies on the PANW can then be set on the “**domain**” portion of the **domain\username** if required.

## Adding Palo Alto Networks Panorama Context Server Endpoint

Under **Administration > External Server > Endpoint Context Servers > Add Context Server > [choose] Palo Alto Networks Panorama**, enter the required IP address of the Palo Alto Networks Panorama server and a username/password pair that ClearPass will use to send the information. In addition, it's very important that you configure the serial numbers of the Palo Alto Networks firewall that are under management by the Panorama appliance as shown below, e.g. 1234567890 in Figure 13.

We discuss configuring the username used below in a later section *"Configuring Palo Alto Networks Next-Generation Firewall"*.



Modify Endpoint Context Server	
Server	
Server Type:	Palo Alto Networks Panorama
Server Name:	10.2.100.15
Server Base URL:	https://{server_ip}/api/?type=keygen&user={username}&password={password}
Username:	cppm-api
Password:	..... Verify: .....
Username Transformation:	Prefix NETBIOS name
GlobalProtect:	<input checked="" type="checkbox"/> GlobalProtect Enabled on Palo Alto Networks Firewall
Palo Alto Firewall Serial Numbers:	1234567890 0987654321
UserID Post URL:	https://{server_ip}/api/?type=user-id&action=set&key={key}&cmd={cmd}
Validate Server:	<input type="checkbox"/> Enable to validate the server certificate
Enable Server:	<input checked="" type="checkbox"/> Enable to fetch endpoints from the server
Bypass Proxy:	<input checked="" type="checkbox"/> Enable to bypass proxy server

Figure 13 - Adding Palo Alto Networks Panorama endpoint



**Note:** Do not change the Server Base URL or UserID Post URL. Although these fields can be modified, they are formatted to work with Palo Alto Networks Panorama running 5.x or 6.x software.

The same option is discussed in the previous section in relation to the support for GlobalProtect, and appending Domain information to usernames is supported within the Panorama configuration.

## Summary of Shared Context Attributes

The following table details the contextual attributes ClearPass Policy Manager currently shares with the Palo Alto Networks endpoints.

Attribute	CPPM 6.1.x & 6.2.x	CPPM 6.3/6.4	CPPM 6.5.x
UserID	✓	✓	✓
Source IP	✓	✓	✓
Device Type	✓	✓	✓
Domain Name	✗	✓ [1] [2]	✓ [1] [2]
Host Name	✗	✓ [1]	✓ [1]
Per-Class Health/Posture	✗	✗	✓ [3]

**Figure 14 - Attributes we can share with Palo Alto Networks endpoints**

**Note: [1]** These attributes are passed from CPPM to the Palo Alto Networks endpoint via HIP Objects. The Palo Alto Networks endpoint MUST have a Global Protect License installed to be able to utilize the received HIP data and thus use it in its policy enforcement.

**Note: [2]** The Domain Name can be passed starting in CPPM 6.3.0 with the UserID XML API or via the HIP Objects enabled by use of the GlobalProtect License.

**Note: [3]** To capture the Health/Posture context for an endpoint requires that OnGuard be installed on that endpoint. OnGuard is available for Windows/Mac OS X/Ubuntu. We added the ability to send this endpoint context in CPPM 6.5.

After completing the steps in the previous sections, there are a couple of final steps to ensure that as users are authenticated with ClearPass, information is sent to update the Palo Alto Networks endpoint. This is performed using post\_authentication Session Restrictions profiles.



### Triggering Updates from ClearPass to a PANW endpoint post 6.5

In CPPM 6.5, changes were made to expand the Policy Manager Exchange Framework. This resulted in a new post\_authentication profile type “**Session Notification Enforcement**” being created. Note that for CPPM systems migrating from 6.4 or earlier we will migrate the previous CPPM enforcement profile used (Session Restrictions Enforcement profiles) to the new enforcement profile type.

Adding this enforcement profile for PANW is slightly different from previous CPPM versions. An example is below. Note that you have to specify two attributes of type **Session-Notify**, a **Server-Type** and a **Server IP**. If you have not previously defined the PANW context server endpoint, then when trying to configure this step nothing will be available in the drop-downs.

Configuration » Enforcement » Profiles » Edit Enforcement Profile - PAN-update-node

#### Enforcement Profiles - PAN-update-node

Summary	Profile	Attributes
<b>Profile:</b>		
Name:	PAN-update-node	
Description:	PAN-update-node	
Type:	Post_Authentication	
Action:		
Device Group List:	-	
<b>Attributes:</b>		
Type	Name	Value
1. Session-Notify	Server Type	= Palo Alto Networks Firewall
2. Session-Notify	Server IP	= 10.2.100.10

Figure 15 - Adding an enforcement-profile for PANW in CPPM 6.5

## Triggering Updates from ClearPass to a PANW endpoint pre 6.5

Create a new Enforcement Profile as shown. Ensure this profile is created from the **Session Restrictions Enforcement** template. Select Type to be **Session-Check** as shown below, then Name to be **IP-Address-Change-Notification** and finally from the Value field the IP address of the Palo Alto Networks endpoint.

Figure 16 - Adding a Session Restriction Enforcement profile

**Type = Session-Check**

**Name = IP-Address-Change-Notification**

**Value = Palo Alto Networks endpoint**, previously added (this is a drop-down list)

**Note:** If you don't see the IP address of a Palo Alto Networks endpoint, you have likely missed a step in one of the earlier sections, likely you have missed adding the endpoint under the Context Servers.



**Note:** If you have multiple Palo Alto Networks Firewall / Panorama then you **must** create **multiple** Enforcement Profiles, one per endpoint. The option exists as shown below to add multiple Palo Alto Networks endpoints to a single enforcement profile, however this configuration is invalid.

Figure 17 - Adding a Session-Check ..... one endpoint per profile



## Adding an Enforcement Profile to a Enforcement Policy

From this point to complete the configuration it's a very standard CPPM workflow. A enforcement policy needs to be created using what ever criteria you need. Be this based upon a AD group membership match (as in our example below), or an authorization match or a match from within the user's presented certificate. The choices are virtually unlimited.

Following this add the enforcement policy to a service profile. In our below example, CPPM will send an update when the authenticated user is a member of the AD Group ns-tme in our Active-Directory win28k.

Configuration » Enforcement » Policies » Edit - update-pan-firewall

### Enforcement Policies - update-pan-firewall

**Summary** Enforcement Rules

**Enforcement:**

Name:	update-pan-firewall
Description:	
Enforcement Type:	RADIUS
Default Profile:	[Allow Access Profile]

**Rules:**

Rules Evaluation Algorithm: First applicable

Conditions	Actions
1. (Authorization:win28k:memberOf CONTAINS ns-tme)	PAN-update-node (dot10), [Allow Access Profile]

Figure 18 - Trigger PANW update on AD memberOf

## Adding an Enforcement Policy to a Service Policy

Adding the Enforcement policy to a service policy, a very simple example.

Configuration » Services » Edit - PANW Service

### Services - PANW Service

**Summary** Service Authentication Roles **Enforcement**

Use Cached Results: ☐ Use cached Roles and Posture attributes from previous sessions

Enforcement Policy: update-pan-firewall [Modify](#) [Add new E](#)

**Enforcement Policy Details**

Description:	
Default Profile:	[Allow Access Profile]
Rules Evaluation Algorithm:	first-applicable

Conditions	Enforcement Profiles
1. (Authorization:win28k:memberOf CONTAINS ns-tme)	PAN-update-node (dot10) [Allow Access Profile]

Figure 19 - PANW enforcement profile added to a service policy

## Sending Health/Posture status to PANW from CPPM

The ability for CPPM to send Posture/Health context to PANW was a feature introduced in the CPPM 6.5 code release. To take advantage of this feature requires several moving parts. The full configuration of OnGuard is beyond the scope of this document.

By adding this new functionality ClearPass provides additional valuable context about the endpoint and the health/posture of that device. By utilizing the ability for ClearPass Policy Manager OnGuard NAC client to capture valuable endpoint context we enhance the ability of the PANW firewall to make more enhanced granular policy enforcement decisions.

The OnGuard client has the ability to report multiple individual attributes about a health/posture class (listed below). As an example for antivirus: is the AV Product current/back level, is the AV engine current/back level, has the signature data-file been updated in the last X hours, when was the last scan performed, is real-time scanning enabled.

The complete list of classes we check within the CPPM 6.5 release are as follows; note that different checks can happen based upon the Client OS.

- **Client Version Check**
- **File Check**
- **Process's Check**
- **Virtual machine Check**
- **Firewall Check**
- **AntiVirus Check**
- **AntiSpyWare Check**
- **Network Connection Check**
- **Hotfixes Check**
- **Installed Applications**

CPPM then evaluates this information and reports up to the PANW, again at an individual class level with a posture token that can be one of the following as configured by you (in OnGuard): **healthy / quarantined / checkup / transition / infected / unknown** per class.

For the PANW to take advantage of this context requires configuration on CPPM and within the PANW. Below we discuss the configuration required within CPPM and PANW to make this happen.

## Configuring OnGuard on ClearPass

The configuration of the OnGuard client and Policy is beyond the scope of this document. But in brief we recommend that you use the standard wizard supplied with CPPM to build the basic service policy definitions and then create your posture policies as required per platform... Windows/OSX/Linux.

## Configuring PANW to use Health/Posture context



Within the PANW firewall we utilize TAGS and ADDRESS-GROUPS to match the data posture/health context sent by CPPM. These items need to be pre-created within the PANW.

### Configuring TAGS for Health/Posture

Under the **Device Tab->TAGS [Add]** create the following tags. The names and case have to be a 100% match to the list below, else the data sent by CPPM will not match and the policy enforcement will fail.

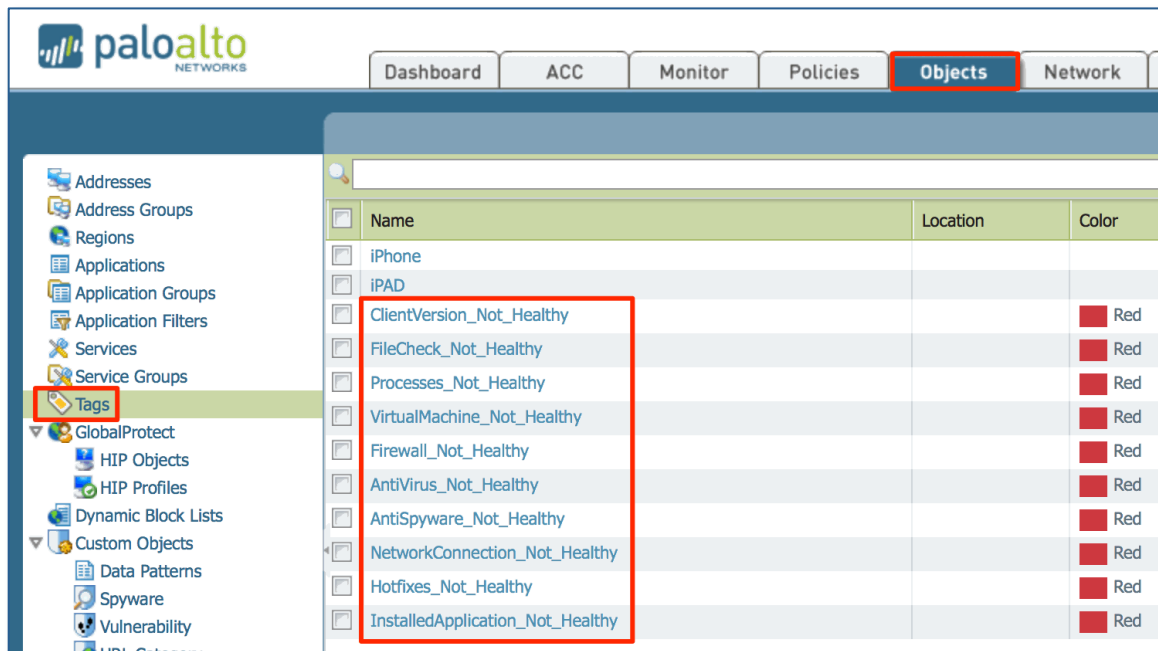


Figure 20 - Configuring TAGS on PANW

Below is a list that can be copied as a reference for the above TAGS when configuring them with in the PANW firewall.

```
ClientVersion_Not_Healthy
FileCheck_Not_Healthy
Processes_Not_Healthy
VirtualMachine_Not_Healthy
Firewall_Not_Healthy
AntiVirus_Not_Healthy
AntiSpyWare_Not_Healthy
NetworkConnection_Not_Healthy
Hotfixes_Not_Healthy
InstalledApplication_Not_Healthy
```

Figure 21 - Copy of "Not\_Healthy" TAGS



As you can see we are creating TAGS with “\_Not\_Healthy” extensions. We want to capture and enforce when people are outside the policy, not when they are compliant. After creating the TAGS we assign them to an Address Group, as below. Address Groups are a collection of TAGS, but the Address-Group match can be built using Boolean AND / OR joins to make for very granular and specific policy rules. See the Address Group below where we are looking for ANYTHING un-healthy to trigger a match.

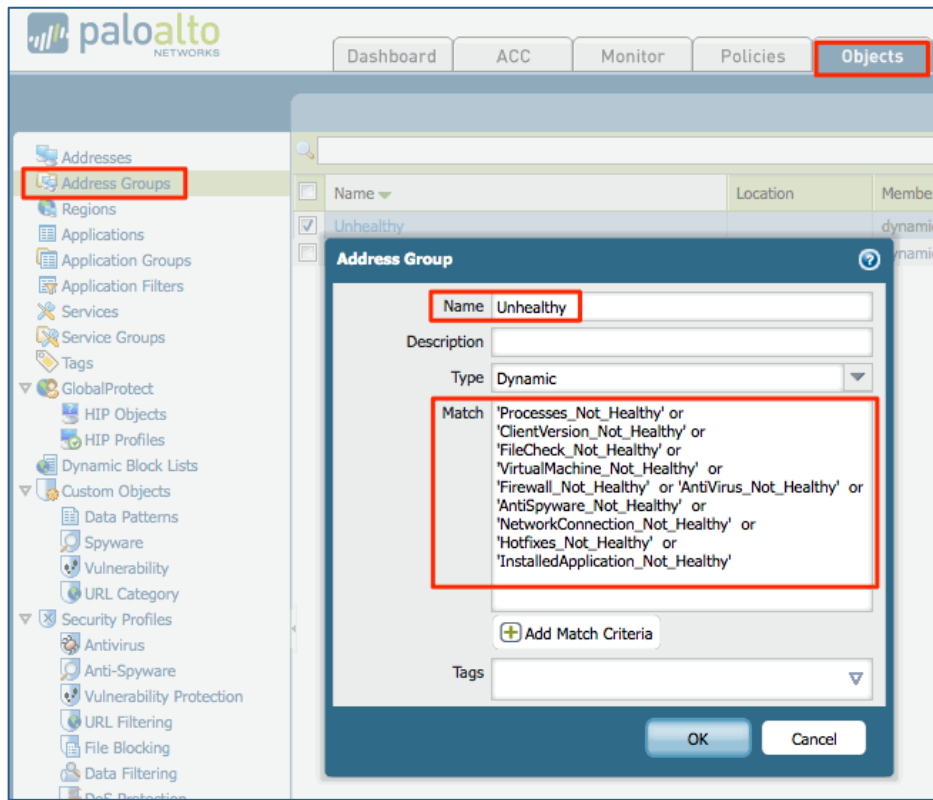


Figure 22 - Creating an Address-Group to match on ANYTHING unhealthy

For our testing we also created some individual checks; they are self-explanatory.

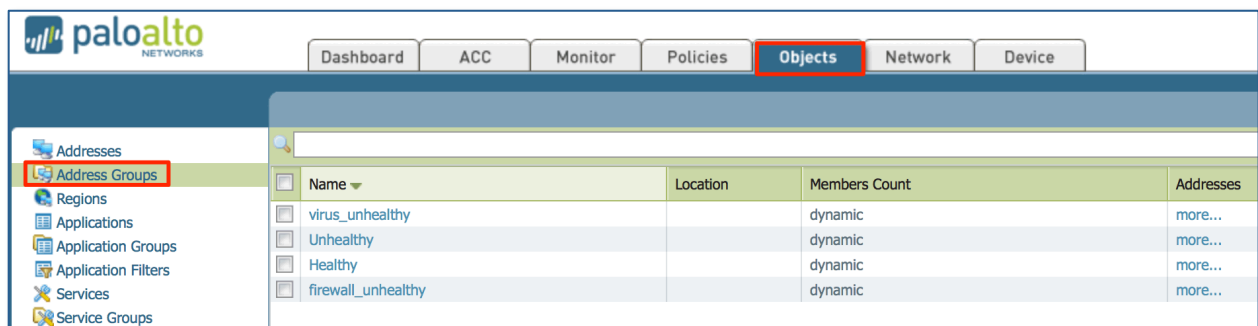


Figure 23 - Creating different Address-groups to check on individual failures

Once the Address Groups have been created, and you could have multiple ones according to how you want to enforce/restrict users based upon their health/posture context, you can apply these to policies within the PANW firewall.

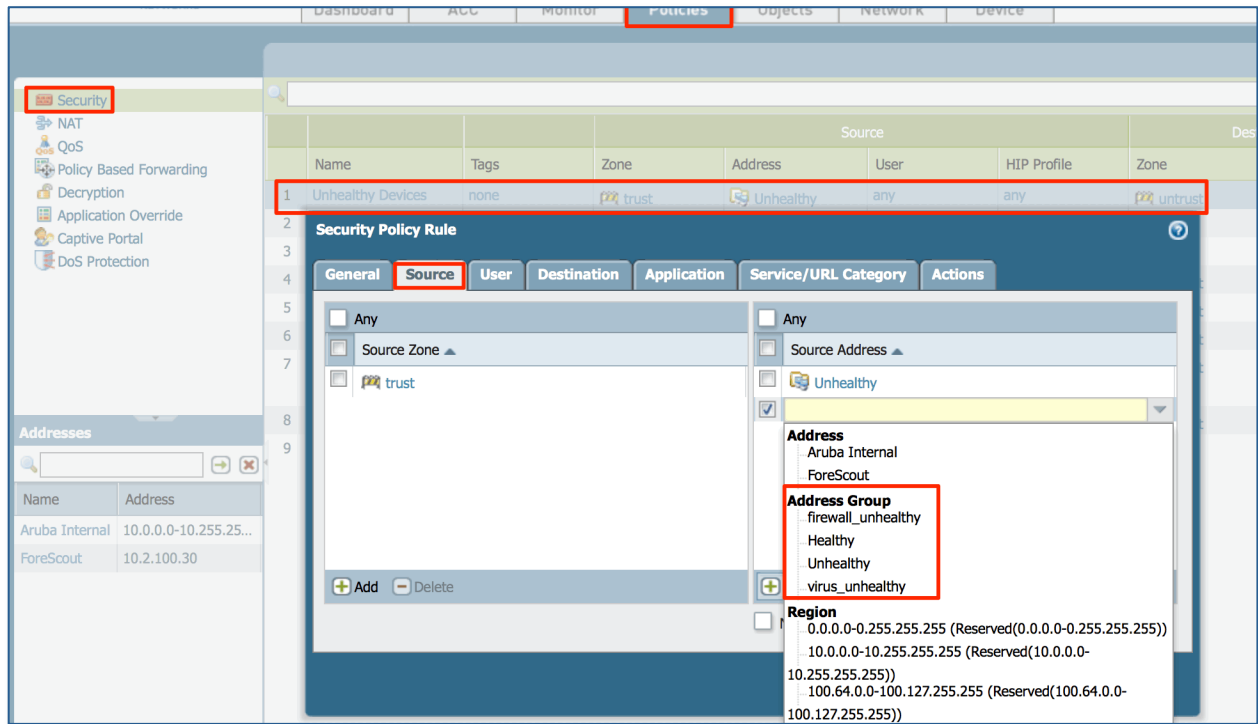


Figure 24 - Adding an Address-group to and firewall policy



## Setting the CPPM Posture/Health Delay Timer

When utilizing CPPM's ability to send OnGuard posture/health status to a Palo Alto Networks endpoint, you must set the post\_authentication eager timer to a MINIMUM of 120 seconds. This is required to allow the OnGuard client time to receive the policy analysis required for the endpoint and then trigger the local processing on the endpoint to analyze and post the results back to CPPM. For this reason we strongly recommend the eager-timer is set to 120 seconds.

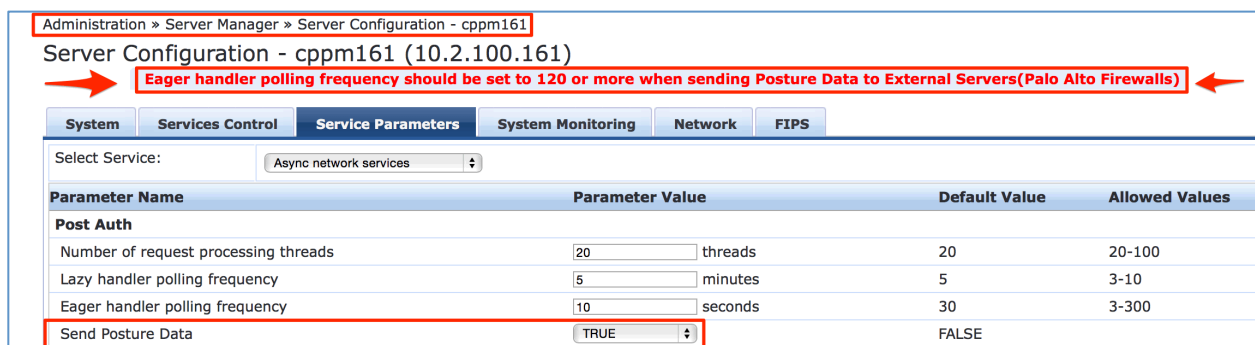


Figure 25 - Setting Eager handler to 120 seconds when sending posture/health

## Configuring Palo Alto Networks Next-Generation Firewall

Several steps must be completed to take advantage of the integration we have developed. Many use cases exist in the scope of this integration to manage and control a user's access to different resources. We have documented the configuration on the firewall to allow ClearPass to send data to the Palo Alto Networks endpoint and then for the Palo Alto Firewall to be able to use this data/context to make enforcement decisions.

### Configuring CPPM to communicate just using the XMLAPI



For ClearPass to send data to a Palo Alto Networks firewall or Panorama, an account needs to be configured within the Palo Alto Networks firewall/Panorama endpoint. You could utilize the built in **admin** account; however we do not recommend this. Please create a new account to be used solely for the purpose of ClearPass communicating with the Palo Alto Networks firewall. We recommended creating a role-based admin account. By utilizing the role-based admin account, the account can be limited to **only** communicating with the Palo Alto Networks firewall via the XML API.

**Note:** The account created here needs to match that configured in the endpoint context server on CPPM when adding the Palo Alto Networks endpoints.

Under the **'Device'** tab and **'Admin Roles'** create an admin-role as below. Ensure that you disable all the options on the Web UI Tab and the XML API **except** the **'User-ID Agent'** as shown below.

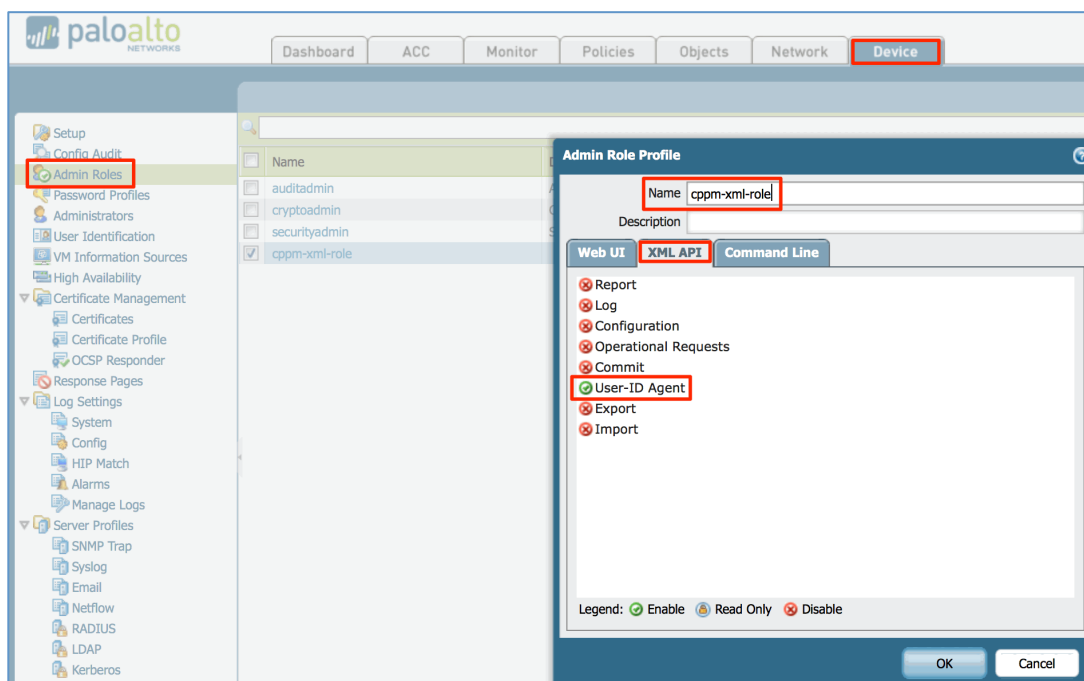


Figure 26 - Creating an restricted Admin-Role

Now we create the actual Admin userid we will use when defining the PANW endpoint on CPPM in the context-server definition. Again, under the 'Device' tab but this time under Administrators create an admin user.

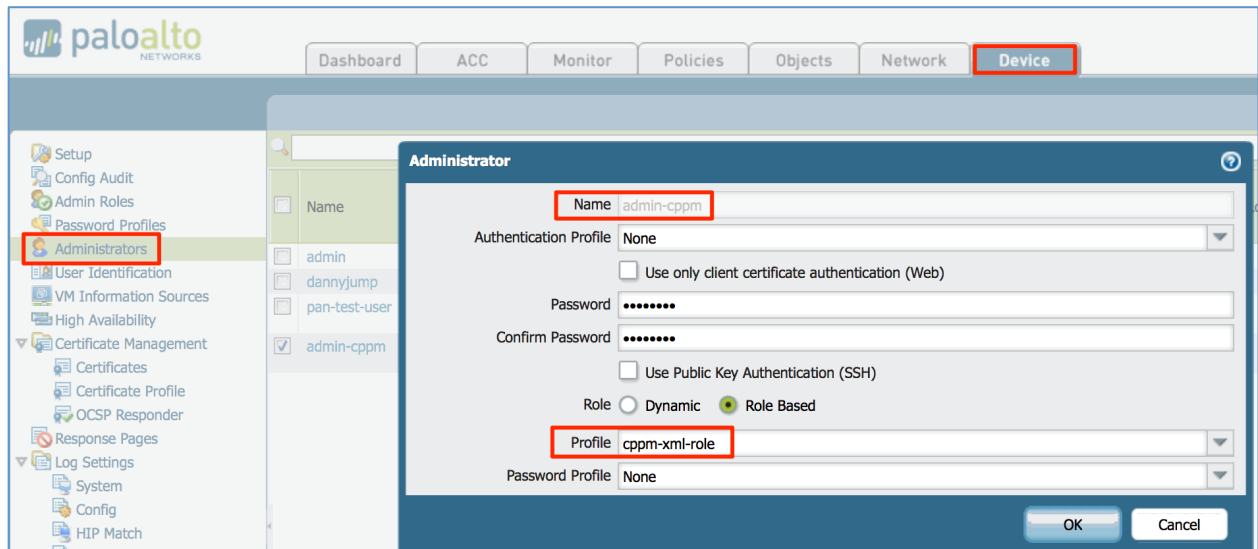


Figure 27 - Adding a User to Palo Alto Networks Firewall

In this example we have added an admin user called '**admin-cppm**', but it references the profile cppm-xml-role we created in the pervious step.

### Configuring a Policy on PANW to use CPPM context data – generic info....

In PAN-OS 6.x there is a feature called Dynamic Address Groups (DAG) that is used to create labels/tags for endpoints. These DAG's are then used for enforcement. They have replaced the Dynamic Address Objects (DAO) from PAN-OS 5.x. The XMLAPI that CPPM uses to send data has not changed, it's the way this received data is utilized and configured on the firewall that is a little different. In 5.x you created a DAO with an identifier, then via the XMLAPI you attached endpoint IP's to that identifier, and they will show up in the DAO.

Starting in PAN-OS 6.0, you create TAGS and then combine these identifiers together under an Address Group. You can use Boolean logic like AND / OR to combine multiple tags in the Address Group. Then through the XMLAPI you 'attach' the client's IP address to the tags. The tags become part of that Address Group, similar to the process under PAN OS-5.x

A Palo Alto Networks firewall can then enforce a policy utilizing dynamic objects, DAO and DAG's. In essence they provide the same type of functionality, an object type that is not tied to a fixed IP address. Aruba's ClearPass can complement a Palo Alto Networks firewall by supplying the dynamic object data and mapping an endpoint to a dynamic object or tag.

**Note:** If you only use one tag, then a DAG is the same as a DAO. But the difference is you could use more than one tag and associate them with AND and OR to make complex conditions for an IP appearing in the DAG.

## Creating Device Profile Categories

We have to manually create the device categories in PAN-OS. Starting in CPPM 6.3 we enhanced the granularity of the endpoint information we are able to send to the Palo Alto Networks endpoint. Prior to this release we were only sending the Device Category, e.g. Computer or SmartDevice. In CPPM 6.3 we now utilize the power of the CPPM Profiler to classify the endpoint and use the most granular level of information available to provide this context to the Palo Alto Networks endpoint.

A device profile is a hierarchical model consisting of 3 elements – **DeviceCategory**, **DeviceFamily**, and **DeviceName** – derived by Profile from endpoint attributes.

**DeviceCategory** - This is the broadest classification of a device. It denotes the type of the device. Examples include Computer, SmartDevice, Printer, Access Point, etc.

**DeviceFamily** - This element classifies devices into a category and is organized based on the type of operating system or vendor. For example, when the category is Computer, ClearPass Policy Manager could show a **DeviceFamily** of Windows, Linux, or Mac OS X, and when the Category is Computer, ClearPass Policy Manager could show a **DeviceFamily** of Apple or Android.

**DeviceName** - Devices in a family are further organized based on more granular details, such as operating system version. For example, in a **DeviceFamily** of Windows, ClearPass Policy Manager could show a **DeviceName** of Windows 7 or Windows 2008 Server.

This hierarchical model provides a structured view of all endpoints accessing the network. As a reference, the list of Device Category/Family or Name of a device that was authenticated in ClearPass can be viewed under **Administration > Dictionaries > Fingerprints**.

Administration » Dictionaries » Fingerprints

### Device Fingerprints

Filter: Category contains  Go Clear Filter Show 10 records

#	Category	Family	Name
11.	Access Points	Buffalo	Buffalo AP
12.	Access Points	HP	HP ProCurve AP
13.	Access Points	Cisco	Cisco AP
14.	Access Points	Enterasys/Trapeze	Enterasys/Trapeze AP
15.	Barcode Scanner	Symbol	Symbol Scanner
16.	Barcode Scanner	Intermec	Intermec Scanner
17.	Computer	Windows	Windows 95
18.	Computer	Windows	Windows 2008
19.	Computer	Apple Mac	Mac OS X
20.	Computer	Windows	Windows XP

Figure 28 - CPPM Fingerprints



## Configuring Palo Alto Networks PAN-OS 6.x - Tags and HIP Objects

There are two methods we can use to 'match' device/user context that we send from CPPM that can be used within the PANW policies. The first method is Tags the second is HIP.



Tags can be manually (static) or automatically (dynamically) created, we typically use the static tags as we know what they will be!! Dynamically created tags are typically unknown.

Once you decide on the Categories of devices you require from CPPM, create them on the Palo Alto Networks firewall as Tags. PAN-OS 6 differs from previous version, Tags now replace Dynamic Objects. We discussed creating TAGS previously.

**Note:** Profiling must be enabled or CPPM is unable to send HIP level data.

**To create the Tags** select the **Object** Tab, then **Tags** and then on the bottom LHS click **+ Add** to add a new Tag. Below you can see a number of example Tags we have created.

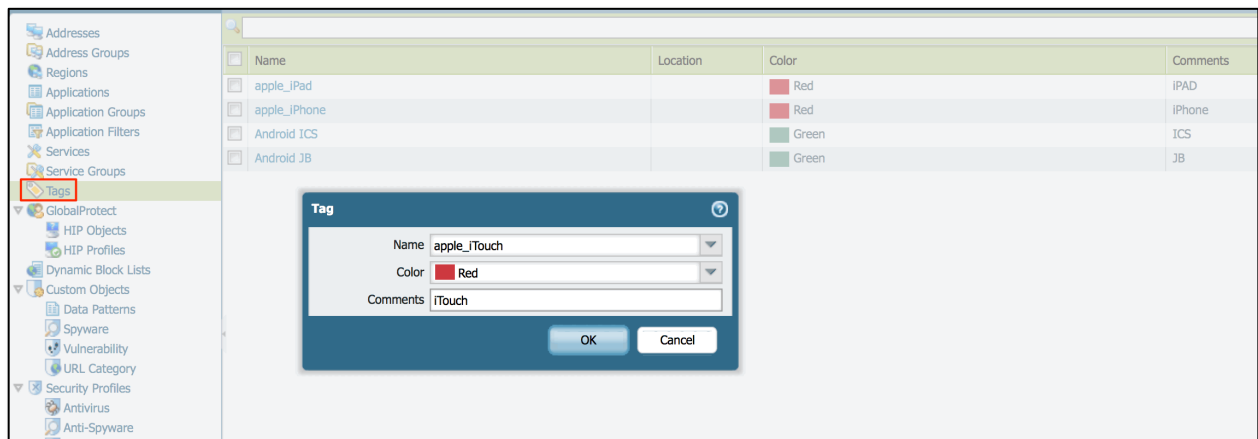


Figure 29 - Adding a TAG under PAN-OS 6.x

### Group TAGS in Address Groups

Following on from creating the individual Tags you have the option to group these together. In this example we created multiple Tags for different Apple device types, then grouped them together under a generic Apple grouping in an Address Group.



**Note:** Boolean logic can be applied to the match criteria to enhance the selection of a match. Note also that when creating the Address group, the Address Group created must be of a type 'dynamic'.

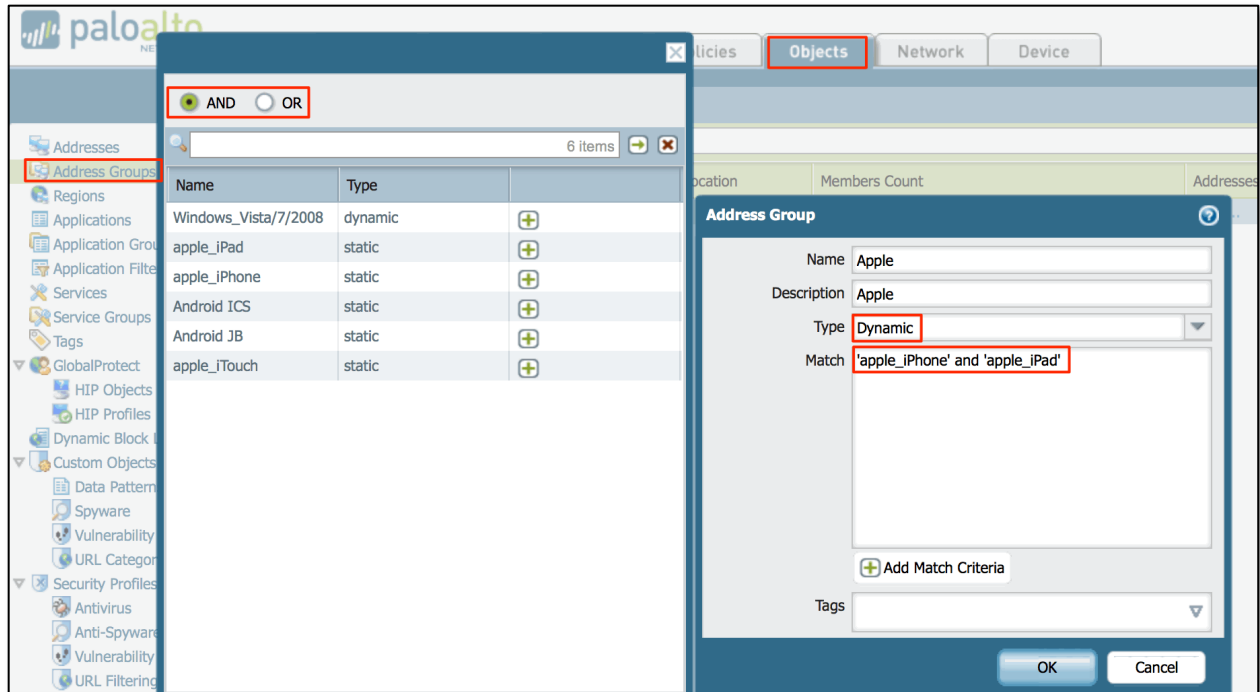


Figure 30 - Grouping Tags into a Dynamic Address Group

To create the **HIP Objects** select the **Object** Tab, then under **GlobalProtect**, you'll find HIP Objects and HIP Profiles, on the bottom LHS click to **+ Add** add a new HIP Object. HIP Profiles are a collection of HIP Objects in a similar way that Address Groups are a collection of Tags. When creating a HIP Object only the General Tab can be referenced for the match. An example below matches Host OS as shown with contains "Apple".

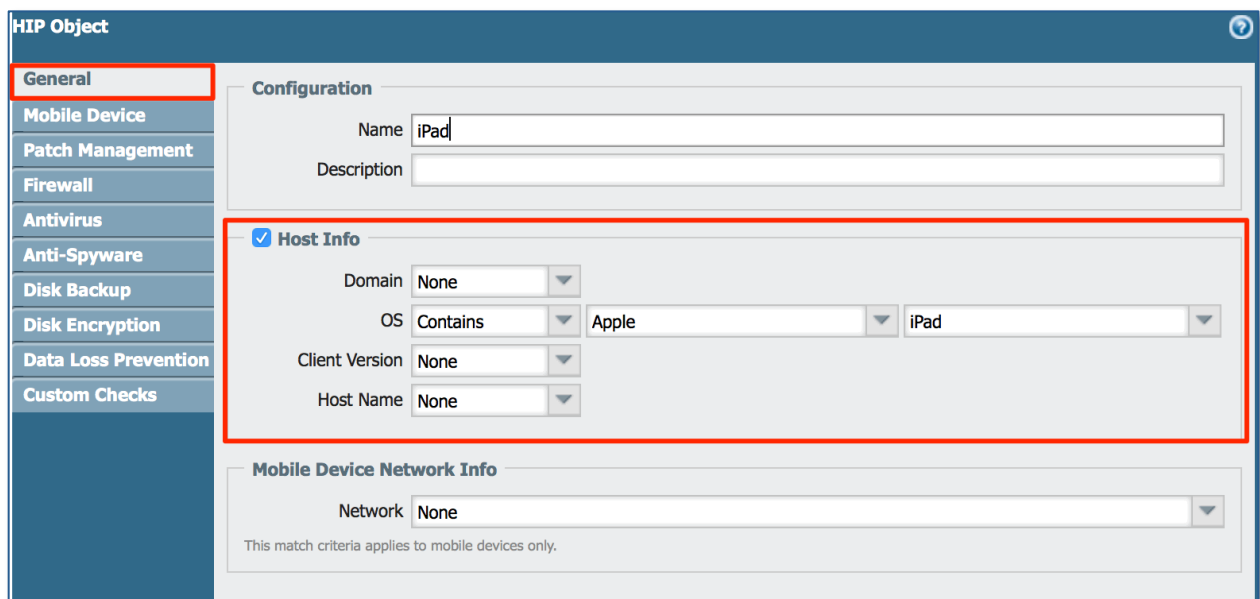


Figure 31 - Creating HIP Objects

## Other Attributes from HIP Object/General explained

**Domain** is in the context of the attribute **Domain\Username** from a users login.

**Client Version** comes from the attribute **Name** in the CPPM fingerprints DB, see below.

**Host Name** is in the context of the attribute we profile from the endpoint.

The below extract from our fingerprint DB shows a small subset of device-types that we can match against for HIP context. Currently the fingerprint DB shipped includes in excess of 350 fingerprints. We continue to add/update these through our bi-weekly fingerprint update that is automatically pushed to all Internet-connected CPPM nodes with an active subscription license. You are actively encouraged to send new fingerprints to us by opening a TAC case.

Administration » Dictionaries » Fingerprints

### Device Fingerprints

Filter: Category  contains

#	Category ▲	Family	Name
21.	Computer	Linux	Chrome OS
22.	Computer	Linux	FortiOS
23.	Computer	Windows	Windows XP
24.	Computer	Solaris	Solaris
25.	Computer	Linux	SUSE
26.	Computer	Linux	Debian/Ubuntu/Knoppix
27.	Computer	Windows	Windows 95
28.	Computer	Linux	Fedora
29.	Computer	Windows	Windows 2008
30.	Computer	Apple Mac	Mac OS X
31.	Computer	Windows	Windows ME
32.	Computer	Windows	Windows

Figure 32 - CPPM Fingerprints – Client Version

Now with the Palo Alto Networks firewall you can reference **advanced** context supplied by CPPM to allow decisions to be made in how traffic should be processed by the firewall.

paloalto NETWORKS

Dashboard ACC Monitor **Policies** Objects Network Device


Security

NAT  
QoS  
Policy Based Forwarding  
Decryption  
Application Override  
Captive Portal  
DoS Protection

	Name	Tags	Zone	Address	User	HIP Profile	Zone	Address	Application	Service	Action
1	Deny Apple	apple_iPad apple_iPhone apple_iTouch	any	any	any	any	any	Drop apple	any	application-d...	

Figure 33 - Utilizing Tags in a Firewall Rule

## Configuring Palo Alto Networks PAN-OS 5.x - Dynamic-Objects

If you're still running on PAN-OS 5.x then you still need to create the 'labels' on the PANW firewall to match the Categories of devices sent from CPPM in a similar way to how we interact with PAN-OS 6.x. You must create these via the GUI in the PANW firewall. Select the **Object** Tab and then under '**Addresses**' on the bottom LHS click  to add a new device. Pay special attention to the box on the right-hand-side as this entry must match the Category in CPPM that is transmitted via the XMLAPI.

Type must be that of '**Dynamic**' as shown below.

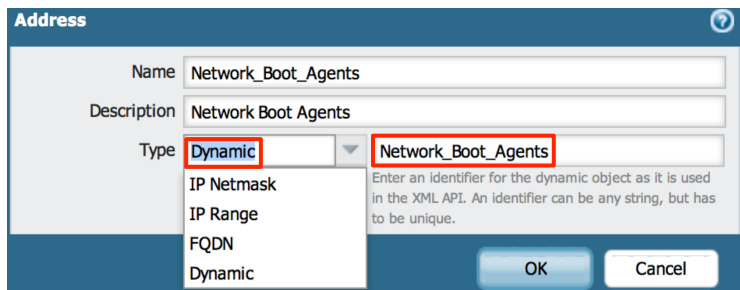


Figure 34 - Configuring Dynamic Objects under PAN-OS 5.x

**Note:** The object name **MUST** be **unique**. Only then can the object be referenced in a security policy as a source or destination address.

Below is an example of a list (not complete) of dynamic Categories that CPPM can register an IP address against via the XMLAPI update process.

<input type="checkbox"/> Medical_Device			Dynamic	Medical_Device
<input type="checkbox"/> Monitoring_Devices			Dynamic	Monitoring_Devices
<input type="checkbox"/> Network_Boot_Agents			Dynamic	Network_Boot_Agents1
<input type="checkbox"/> Network_Camera			Dynamic	Network_Camera
<input type="checkbox"/> Router			Dynamic	Router

Figure 35 - Palo Alto Networks 'dynamic' objects



**Note:** When creating definitions on the Palo Alto Networks firewall, a category type under ClearPass can use a space in the name. Ensure that on the Palo Alto Networks firewall, dynamic object definitions with a space are created with an **underscore** – for example, "**Game\_Console**" not "**Game Console**".

After the objects are created, the power of the Palo Alto Networks Policy engine can be leveraged. An example firewall rule that exploits this is shown below, allowing gaming consoles to the WEB and similarly restricting access to Corporate VoIP Phones.

Name	Tag	Zone	Address	User	HIP Profile	Zone
my-facebook	none	any	any	any	any	any
tcp-80	none	any	any	any	any	any
skype-probe	none	any	any	any	any	any
allwskspe	none	any	any	any	any	any
denyskypetcp	none	any	any	any	any	any
denyskypeudp	none	any	any	any	any	any
denytcp	none	any	any	any	any	any
allowfb	none	any	any	any	any	any
Allow_WWW_Access	none	trust	Game_Console, Network_Boot_Ag., SetTop_Box	any	any	untrust
Block_VoIP_Corporate	none	trust	VoIP_Phone, dannyjump, marc	any	any	untrust

Figure 36 - Basic Firewall Rules

Historically, traditional firewalls classify traffic based on port number and IP address. However, port number is no longer a meaningful way to classify traffic, because any application can use any port number. The Palo Alto Networks next-generation firewall classifies traffic by application, and enforces policy based on the context of business elements such as application, user, and content.

The following rule shows the use of device types rather than IP address as a source address in the Trust zone, below we are specifically making an enforcement decision on the context type of the endpoint. Note, this device-data has been shared by ClearPass Policy Manager.

Name	Tag	Zone	Address	User	HIP Profile	Zone	Address
my-facebook	none	any	any	any	any	any	any
tcp-80	none	any	any	any	any	any	any
skype-probe	none	any	any	any	any	any	any
allwskspe	none	any	any	any	any	any	any
denyskypetcp	none	any	any	any	any	any	any
denyskypeudp	none	any	any	any	any	any	any
denytcp	none	any	any	any	any	any	any
allowfb	none	any	any	any	any	any	any
Allow_WWW_Access	none	trust	Game_Console, Network_Camera, SmartDevice	any	any	untrust	any
Block_VoIP_Corporate	none	trust	VoIP_Phone, dannyjump, marc	any	any	untrust	any
Allow_FB_Android	none	trust	Game_Console, Network_Camera, SmartDevice	any	any	untrust	any
allow	none	trust	Game_Console, Network_Camera, SmartDevice	any	any	untrust	any

Figure 37 - Firewall Rule Based Upon a Source-Device-Type of an endpoint

In a similar way we can exploit the power of the Palo Alto Networks policy engine to make permit/deny decisions based upon a username. In the example below, we are selecting users 'marc' and 'dannyjump' in creating this particular policy. These UserID's would have been received directly from CPPM via the XMLAPI.

Again legacy firewalls would typically restrict or allow users sessions based upon one of the basic 5-tuple identifiers only, now we can utilize additional context to apply a firewall policy using next generation object-level context. Note, this user-data has been shared by ClearPass Policy Manager.

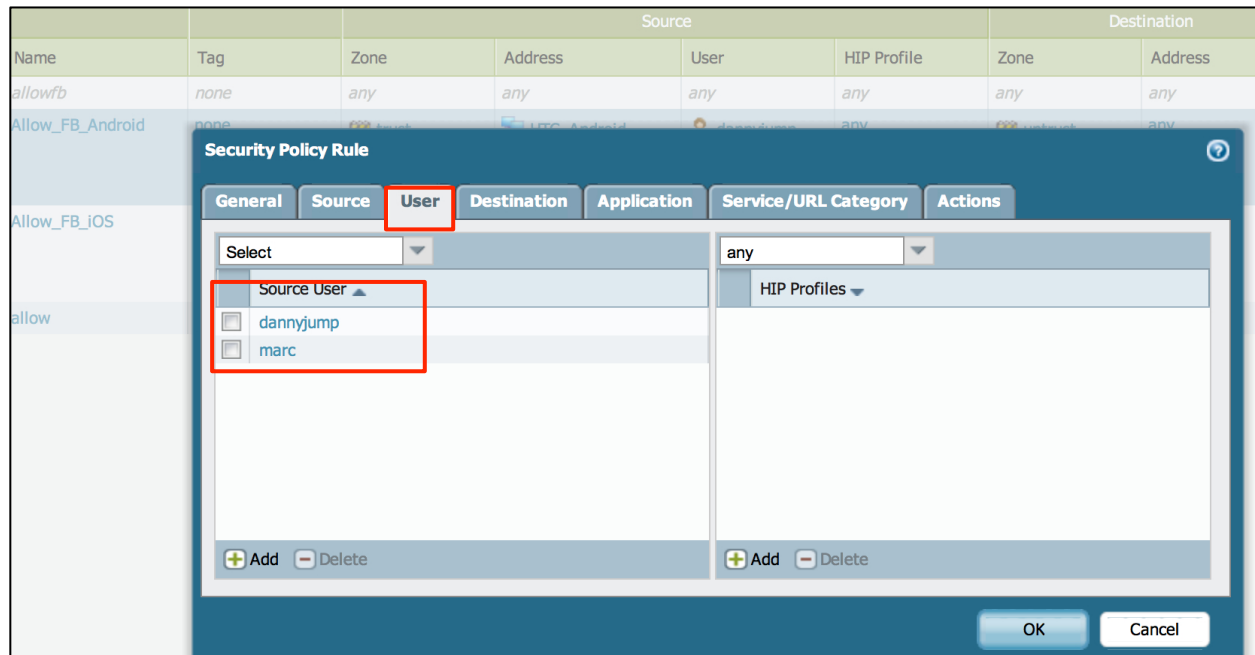


Figure 38 - Firewall Rule Based Upon a Source of a User Name

## PAN-OS 6.x Changes to DAO Limits

PAN-OS 5.x had a restriction in that you could not allocate more than 256 IP addresses per DAO. Within PAN-OS 6.x, these limits have been significantly expanded. The new limits on a per-platform basis are as follows:

- PA-7000, PA-5060, VM-300: 100K
- PA-5050: 50K
- PA-5020: 25K
- PA-4000/3000: 5K
- PA-2000/500/200/VM-100/VM-200: 1K

These limits are the number of IP addresses that can have a tag on the platform. These limits are shared across all VSYS on the platform.

## Faultfinding Tips (PANOS cli cmds/CPPM Logs)

There are several commands and log-files available within the Palo Alto Networks Firewall and CPPM to assist a user in identifying communication and integration problems.

The first section covers some useful cli commands to assist in debugging the Palo Alto Networks environment. Note that as Palo Alto has developed their features not all command below are support on all releases. Some commands were deprecated between PAN-OS 5.x and PAN-OS 6.x, we have made reference to this below. Following this is a section on CPPM debugging.

### UserID <-> IP Address Mapping (PAN-OS 5.x & 6.x cmd)

To look at the user's that are logged in and their IP address mapping, use the following command..... **show user ip-user-mapping all**

```
admin@PA-500> show user ip-user-mapping all
```

IP	Vsys	From	User	IdleTimeout(s)	MaxTimeout(s)
10.4.28.110	vsys1	XMLAPI	bob1	Never	Never
10.4.28.200	vsys1	XMLAPI	bob2	Never	Never
10.2.101.231	vsys1	Unknown	unknown	0	3
192.168.11.104	vsys1	XMLAPI	wgjtest	Never	Never
10.17.24.77	vsys1	Unknown	unknown	3	6
172.31.99.191	vsys1	XMLAPI	gjwang	Never	Never
Total: 6 users					

Figure 39 - Signed in User's to their IP Mapping

If you use the command **show user ip-user-mapping ip [ip address]** it shows you a little additional information where user attributes are being used by PANW policies.

```
admin@PA-3020> show user ip-user-mapping ip 10.2.100.178
```

```
IP address: 10.2.100.178 (vsys1)
User:      alice
From:      XMLAPI
Idle Timeout: Never
Max. TTL:  Never
Groups that the user belongs to (used in policy)
HIP profiles that user belong to (used in policy)
HIP profile(s): HIP-iPad
```

Figure 40 - Signed in Users to their IP Mapping and also matched policy hits

## Dynamic Device (Tag) <-> IP Address Mapping (PAN-OS 5.x & 6.x cmd)

The '**debug user-id dump registered-ip all**' command shows any IP addresses with 'tags'. This is part of Dynamic Address Groups (DAG) under PAN-OS-5.x. With the changes in the PAN-OS 6.x this doesn't relate to users or HIP, so this command is deprecated in PAN-OS 6.x

```
admin@PA-500> debug user-id dump registered-ip all
```

Identifier	Vsys	Address
Apple_iOS_Device	1	10.2.101.167
thisbetterwork	1	10.1.200.127
		1234:5678:90ab:cdef:2234:2678:20ab:2def
abcd	1	10.1.200.127
		1234:5678:90ab:cdef:2234:2678:20ab:2def
Computer	1	0.0.0.1
		0.0.0.10
		0.0.0.43
		0.0.0.67
		10.2.101.163
		10.4.28.109
		10.13.23.105
		10.15.214.178
Routers	1	0.0.0.55
SmartDevice	1	0.0.0.28
		10.4.28.110
HTC_Android	1	10.2.101.161
dyn-obj	1	10.1.200.127
		1234:5678:90ab:cdef:2234:2678:20ab:2def

Total: 8 objects 19 IP entries  
\*: IP entries received from user-id agent

```
admin@PA-500>
```

Figure 41 - Dynamic Object Category - IP Address Mapping

If you want to see logged in users in PAN-OS 6.x, use the command '**show user ip-user-mapping all**' as shown below.

```
admin@PA-3020> show user ip-user-mapping all
```

IP	Vsys	From	User	IdleTimeout(s)	MaxTimeout(s)
10.2.100.172	vsys1	XMLAPI	bob	Never	Never
10.2.100.167	vsys1	XMLAPI	emea	Never	Never
10.2.100.178	vsys1	XMLAPI	alice	Never	Never
10.2.100.169	vsys1	XMLAPI	aruba-apj	Never	Never
10.2.100.182	vsys1	XMLAPI	bill	Never	Never
10.2.100.180	vsys1	XMLAPI	vorawut	Never	Never
10.2.100.121	vsys1	Unknown	unknown	2	5
10.2.100.171	vsys1	XMLAPI	cam	Never	Never

Total: 8 users

Figure 42 - Logged in user in PAN-OS 6.x



## Show HIP Reports

To display the HIP data related to an endpoint (assuming it is available) use the command **'debug user-id dump hip-report ...'** note you have to specific additional context about HIP report **computer/user/ip** on the cmd

```
admin@PA-3020> debug user-id dump hip-report computer dannysipadmini user alice ip 10.2.100.178

<?xml version="1.0" encoding="UTF-8"?>
<hip-report>
  <md5-sum>9ca33e110b0da9704e36dbec3301699a</md5-sum>
  <user-name>alice</user-name>
  <host-name>dannysipadmini</host-name>
  <ip-address>10.2.100.178</ip-address>
  <generate-time>18/05/2015 09:54:45</generate-time>
  <categories>
    <entry name="host-info">
      <host-name>dannysipadmini</host-name>
      <os>Apple iPad</os>
      <os-vendor>Apple</os-vendor>
    </entry>
  </categories>
</hip-report>
```

Figure 43 - HIP Report for a user

Note that in PAN-OS 6.x there are no longer Dynamic Address Objects (DAO), so the 'dynamic' option has been removed. Instead, look under the Address Groups menu for equivalent of the Dynamic Address Groups (DAG), now known as TAGS.

## Show XMLAPI statistics

The below is a high level view of the XMLAPI statistics, if there is zero activity here then you can assume some serious configuration or network problems exist between CPPM and the Palo Alto Networks endpoints.

**debug user-id dump xmlapi-stats**

```
admin@PA-500> debug user-id dump xmlapi-stats

vsys: vsys1
num of input                : 98
num of user login           : 58
num of user logout          : 29
num of dynamic address object register : 8
num of dynamic address object unregister: 8
num of user group           : 0

admin@PA-500>
```

Figure 44 - XMLAPI Stats

## Active real-time debug monitoring of the UserID process

A very effective way to monitor the XMLAPI process in real-time is using the following commands, this will set up an interactive self updating (**like tail -f**) rolling update for the UserID process.

**debug user-id on debug**

**debug user-id set userid all**

**tail follow yes mp-log useridd.log**

**Note:** Remember to disable the logging **debug user-id off**

Our final debug command for the Palo Alto Networks Firewall shows all of the UserID Manager Data. This shows all users that have been registered through the XMLAPI process.

**debug user-id dump idmgr type user all**

```
admin@PA-500> debug user-id dump idmgr type user all

ID      Name
-----
1       mtsai@arubanetworks.com
2       arubanetworks.com\mtsai
3       abcd
4       ashwath
5       aedan
6       agustin
7       ahmad
8       ahmed
9       aidan
10      aiden
11      aileen
12      aimee
13      ainsley
14      aisha
15      alice_1
16      bob
17      marc
18      karthi
19      bineesh
20      dannyjump
21      srini
22      bini
23      venkat
24      aa
25      dd
26      labuser
27      nbalu
28      test1
29      bob1
30      bob2
31      test2
32      anonymous
33      bob3
34      wgjtest
35      gjwang
999999  pre-logon

Type: 16 Last id: 36 Mismatch cnt: 0
```

Figure 45 - List of ALL users registered through ID Manager

## Check Logs files in CPPM

CPPM collects multiple log files that can assist the user in debugging a CPPM to Palo Alto Networks integration problem. The most useful of these logs is the **postauthctrl.log** file. As you know the trigger that sends data via the XMLAPI is performed by the post\_authentication daemon. Checking this log file can provide an insight in to the working of this process on the CPPM side and possible issues related to the communication with Palo Alto Networks endpoint.

To collect and access this log file takes multiple steps, please follow these steps:

Under **Administration -> Server Manager -> Server Configuration**, select your system if you have a cluster then 'Collect Logs'.

Once this process has completed you need to download this tar file and open with an appropriate application. For OS-X, **finder** will allow you to extract the file to a folder for analysis with the built in Archive Utility. For MSFT Windows multiple applications exist, but a really good free utility is **7-Zip** <http://www.7-zip.org>.

**Note:** You only need to collect as highlighted below '**Logs from all Policy Manager services**' to obtain the postauthctrl.log file. This will save significantly on the log collection process and the corresponding download file is much smaller. If you are not able to analyze an issue and you engage Aruba TAC's it is likely they will want System logs in addition to the Policy manager services logs.

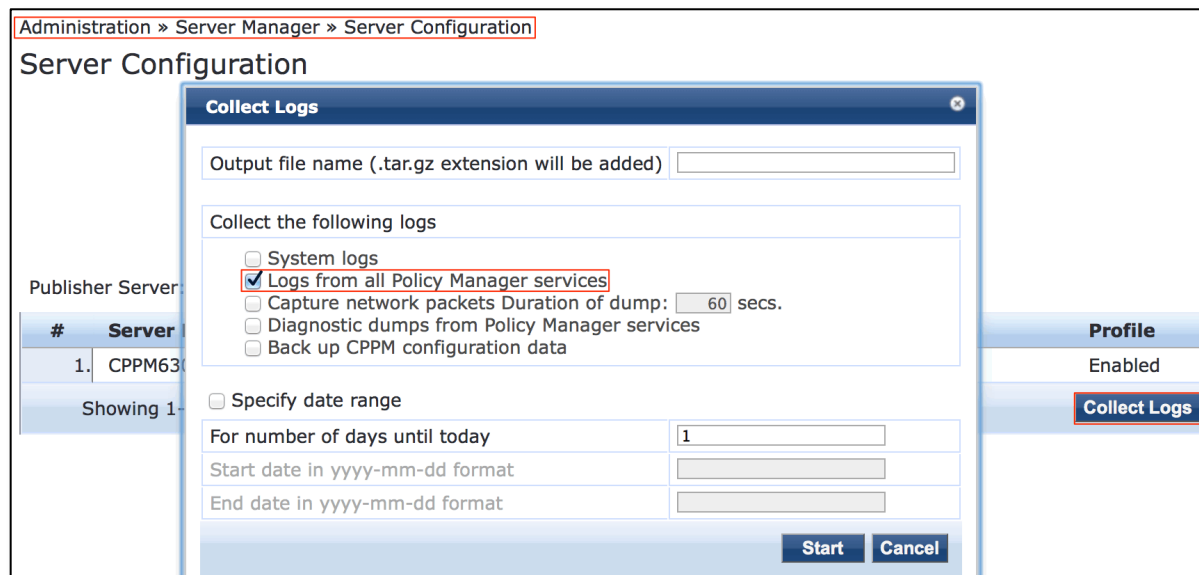


Figure 46 - How to collect CPPM Logs – limited data, but includes postautctrl.log

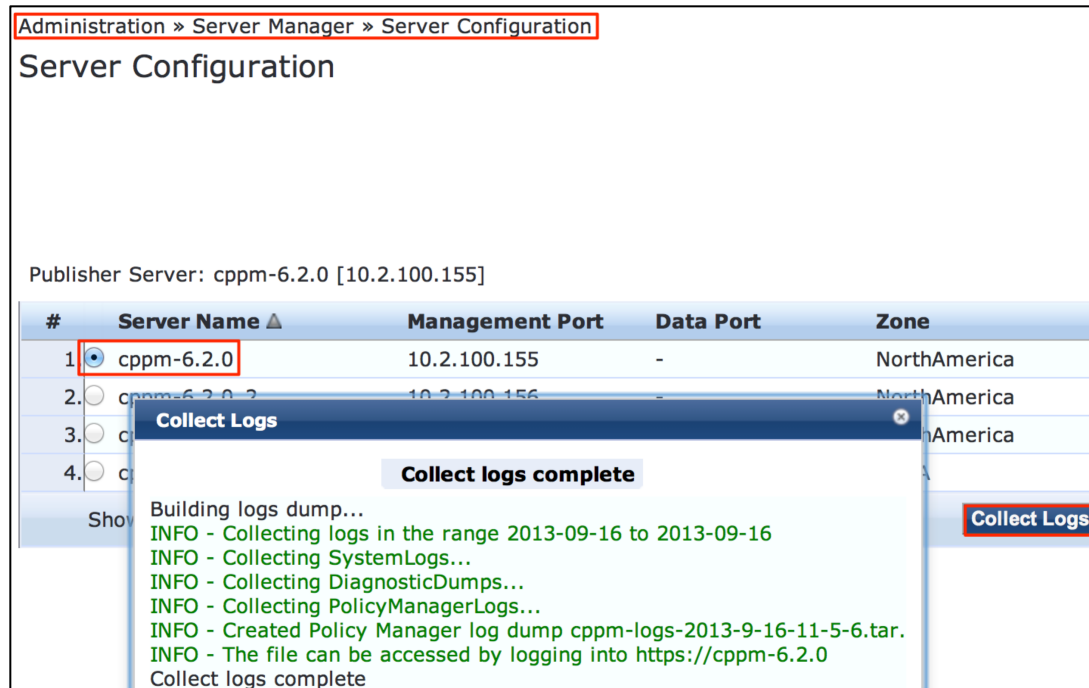


Figure 47 - Collection of CPPM Logs complete

After you have opened the archive, the **postauthctrl.log** file can be found in the following path...

PolicyManagerLogs/async-netd/postauthctrl\*.log

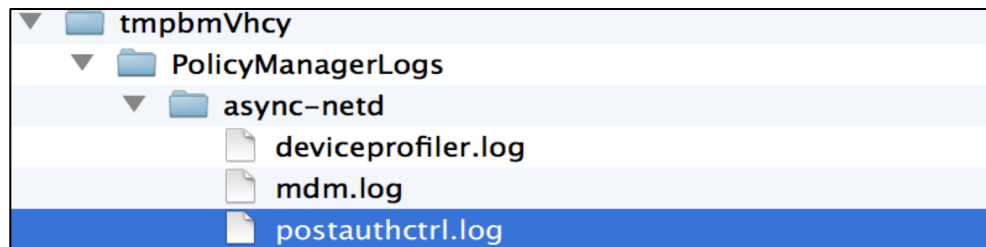


Figure 48 - Where to locate postauthctrl.log

Once you have located the postauthctrl.log file, there are certain entries you will want to look for, several examples are shown below. These provide an insight into the XMLAPI communication between CPPM and the Palo Alto Networks Firewall. Once a user has associated and been authenticated, if the service match that authenticates the user has a post\_authentication Palo Alto Networks trigger then you should be able to match that session to an entry in this log file.

Below are five **example** messages sent from CPPM to a Palo Alto Network endpoint, you'd expect to find these or very similar ones within the postauthctrl file. The last one shown is specific for HIP Objects.

## Sending login UserID + Source IP@, as user logs in

```
<uid-message>
  <version>1.0</version>
  <type>update</type>
  <payload>
    <login>
      <entry name="dannyj" ip="10.4.28.110"/>
    </login>
  </payload>
</uid-message>
```

## Adding IP@ to Category, as CPPM profiles the IP@

```
<uid-message>
  <version>1.0</version>
  <type>update</type>
  <payload>
    <register>
      <entry identifier="SmartDevice" ip="10.4.28.110"/>
    </register>
  </payload>
</uid-message>
```

## Sending logoff UserID + IP@, as user logouts

```
<uid-message>
  <version>1.0</version>
  <type>update</type>
  <payload>
    <logout>
      <entry name="dannyj" ip="10.4.28.110"/>
    </logout>
  </payload>
</uid-message>
```

## Removing IP@ from Category as device logout

```
<uid-message>
  <version>1.0</version>
  <type>update</type>
  <payload>
    <unregister>
      <entry identifier="SmartDevice" ip="10.4.28.110"/>
    </unregister>
  </payload>
</uid-message>
```

## XML example of HIP Object

Sending username, domain-name, host-name, IP@ and client-version (OS-type).

```
<uid-message>
  <version>1.0</version>
  <type>update</type>
  <payload>
    <login>
      <entry name="cppmeccert\certuser1" ip="192.168.100.1"><hip-report>
        <md5-sum>aeea39d589a1f7540d137e56a6d60b31</md5-sum>
        <user-name>certuser1</user-name>
        <domain>cppmeccert</domain>
        <host-name>toshi-driver-32</host-name>
        <ip-address>192.168.100.1</ip-address>
        <generate-time>06/03/2014 12:01:31</generate-time>
        <categories><entry name="host-info">
          <host-name>toshi-driver-32</host-name>
          <domain>CPPMECCERT</domain>
          <client-version>Windows 7</client-version>
        </entry></categories>
      </hip-report></entry>
    </login>
  </payload>
</uid-message>
```

## Conclusion

---

Aruba's ClearPass in conjunction with Palo Alto Networks can provide administrators with full context and visibility about the users and devices on the network to deliver end-to-end safe application enablement. We continue to evolve ClearPass to provide more contextual information about endpoints and users to Palo Alto Networks endpoints to allow them to make more advanced policy decision with regard to the network and its users.