



NETWORKS
an HP company

ClearPass 6.5

Tech Note: ClearPass

Integration with 3rd Party Enforcement Points

ClearPass & Checkpoint utilizing RESTful
API and RADIUS Accounting

<u>Version</u>	<u>Date</u>	<u>Modified By</u>	<u>Comments</u>
0.1 / 0.2	Jan /Feb 2015	Danny Jump	Early Draft Versions
1.0	Mar 2015	Danny Jump	Initial Published Version
1.1	Apr 2015	Danny Jump	Calculating dynamically the user-role that is sent from TIPS role. Added DEBUG commands
1.2	Jul 2015	Danny Jump	Update to cater for R77.xx RESTful API HotFix
1.3	Sept 2015	Danny Jump	Capturing new context attributes added in 6.5.3

Introduction	4
Audience	5
CPPM 6.5 Post Auth Framework Additions	6
Configuring CheckPoint & ClearPass for REST	7
ClearPass Configuration	7
HTTP Context Server:.....	7
Context Server Actions Login/Logout:.....	7
Enforcement Profile:	10
Update to CheckPoint Configurations to support REST in R77.xx Product	13
Check Point Configuration – Where the UserID is verifiable.....	16
Configure Identity-Awareness API Service	16
Adding AD/LDAP Server	17
Check Point Configuration – Where the UserID is a Guest account	22
Monitoring/Debugging Identity Awareness from the CLI	27
Configuring Radius Accounting Proxy	29
Configuring RADIUS Accounting on ClearPass	29
Configuring RADIUS Accounting on a Checkpoint Firewall	31
Configure Identity-Awareness RADIUS Accounting	31
Configure CheckPoint RADIUS Accounting to expose a fourth attribute	34
CheckPoint Installation of Hotfix	34
Configuration of CheckPoint Hotfix	35
Enabling the RADIUS Accounting Group Hotfix	35
Monitoring this function on the Firewall	35
Configuration of ClearPass to add Accounting Attributes [e.g. user role]	36
Configuration of ClearPass to pass a dynamic role attribute to CheckPoint.....	37
Configuration of CheckPoint to parse Accounting Attributes [e.g. user role].....	37
Figure 1 - ClearPass Attributes sent to Firewalls	4
Figure 2 - New Session Notification Enforcement Profile	6
Figure 3 - Adding a Generic HTTP Context Server	7
Figure 4 - Context Server – Action Tab	8
Figure 5 - Context Server – Header Tab.....	8
Figure 6 - Context Server – Content Tab.....	9
Figure 7 - Context Server – Attributes Tab	10
Figure 8 - Check Login & Logout against CheckPoint endpoint.....	10
Figure 9 – ‘Session Notification’ Enforcement Profile.....	11
Figure 10 - Check Point Logout – Action Tab	11
Figure 11 - Check Point Logout – Content Tab	12
Figure 12 – New Check Point URL POST path (R77.xx + RESTful HOTFIX)	13
Figure 13 – Old Check Point URL POST path	14
Figure 14 - Enabling TS (interim placeholder for configuring RESTful API password).....	14
Figure 15 – Setting the PSK in TS to be used by RESTful API service.....	15
Figure 16 - Select the firewall to be configured	16

Figure 17 - Setting the RESTful PSK between CPPM and Check Point.....	16
Figure 18 - Set or Generate the IA PSK	17
Figure 19 - Enabling Active Directory 'Addition' wizard.....	17
Figure 20 - Adding AD Server to Check Point.....	18
Figure 21 - Checking AD is added to Check Point.....	18
Figure 22 - Adding a firewall rule to reference a username	19
Figure 23 - Install Firewall Policy	19
Figure 24 - Installation of Policy in Progress.....	20
Figure 25 - Successful Installation of the Firewall Policy.....	20
Figure 26 - AD user authenticated on Check Point	21
Figure 27 - non-AD user rejected by Check Point.....	22
Figure 28 - Context Server Action for Guest users where the user_groups attribute is set.....	23
Figure 29 - Adding an Access-Role to match the Aruba 'guest'role'	24
Figure 30 - Adding a 'Group' to match 'role' sent from ClearPass	25
Figure 31 - Adding an Access Role to use User Groups	25
Figure 32 - Guest user being 'labeled' with access tag	26
Figure 33 - Firewall Policy denying access to corporate resources	26
Figure 34 - PDP Debug/Monitor output for an IP address	27
Figure 35 - PDP Debug/Monitor for a username	28
Figure 36 - Adding a PROXY Target	29
Figure 37 - Enabling Accounting Proxy Configuration	30
Figure 38 - Configuring Accounting Proxy on an example service	30
Figure 39 - Select the firewall to be configured	31
Figure 40 - Enabling and RADIUS accounting on Check Point.....	31
Figure 41 - Defining a HOST Object in Check Point	32
Figure 42 - Configuring RADIUS accounting on the Check Point firewall	32
Figure 43 - Showing user logging-in of SmartView Tracker	33
Figure 44 - Adding Accounting attribute to ClearPass Proxy	36
Figure 45 - using authz attributes to pass 'role/group' to Checkpoint.....	36
Figure 46 - Calculating the TIPS role from AD memberOf	37
Figure 47 - Creating User Groups & Access Roles	38
Figure 48 - Aligning Groups to Roles	38
Figure 49 - Grabbing RADIUS attributes from CPPM.....	39

Introduction

This series of Tech Notes describes the integration with 3rd Party firewall vendors. Within this document we have captured the integration with CheckPoint specifically the R77.xx product line, some of the latest enhancements we have made to the joint integration depend on specific version of ClearPass and HOFIX support from CheckPoint. There are two methods of integration between ClearPass Policy Manager and CheckPoint. One uses HTTP JSON encoded RESTful APIs the other uses RADIUS Accounting. In the table below we have captured the features and restrictions of different vendors and the capabilities they support.

Note that the framework we have developed is **not** limited to only the below vendors.

Similar to the integration that exists between ClearPass and other vendors, CheckPoint supports at a basic level the ability to pass username and source IP address attributes. But other attributes shown below can also be passed. As a summary this is a list of the attributes we pass from CPPM 6.5.3 to the vendors we have tested.

Feature/ Firewall	CheckPoint	Fortinet	SonicWall	Palo Alto
Source IP	✓	✓	✓	✓
Username	✓	✓	✓	✓
User Role	✓	✓[a]	✗	✗
Domain	✓	✗	✗	✓
Device Type	✓[c]	✗	✗	✓
Machine OS	✓[c]	✗	✗	✓
Machine Name	✓[b]	✗	✗	✓
Health/Posture	✗	✗	✗	✓

Figure 1 - ClearPass Attributes sent to Firewalls

[a] = Available from RADIUS Accounting, not from HTTP REST API calls

[b] = Available from HTTP REST API calls not from RADIUS Accounting.

[c] = Requires ClearPass Policy Manager 6.5.3 and CheckPoint R77.XX and HOFIX's

To ensure you have to correct CheckPoint HOFIX for item [c] above, follow the below procedure to investigate your s/w version and check for compatibility.

You can verify you have the right version of the CheckPoint firewall and the HOTFIX's installed by running the following command from a SSH CLI session:

```
cpvinfo $FWDIR/lib/libpdplib.so
```

You should expect to see the below, pay special attention to the highlighted attributes.

Type = library

Name = pdplib

Module Name = NACServer



Build Number = 990295004

Major Release = NGX

Minor Release = gollum_hf_base_295

The intent of this Tech Note is to document the integration between CPPM and CheckPoint firewalls. This document focuses on testing and integration. CheckPoint has produced their own document “sk104958” which also covers this topic and should be referenced for details about our level of integration.

Where it is practical, best practices will be documented, although not every conceivable use case or deployment can or will be covered here in this document.


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

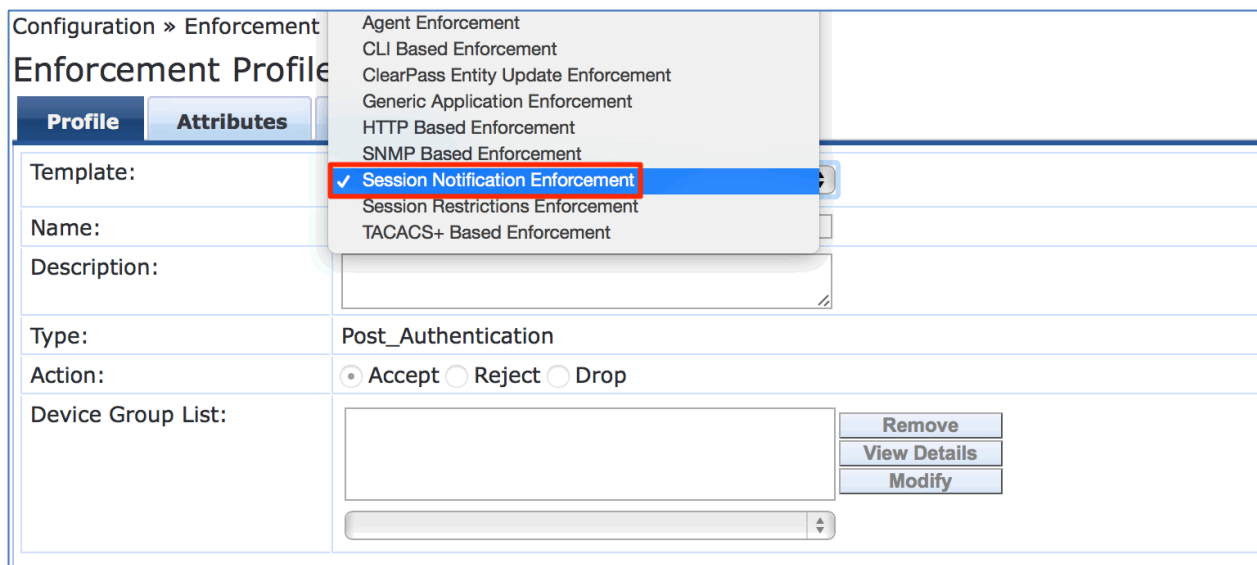
Audience

The reader is assumed to be familiar with the ClearPass family of products, including Policy Manager, Insight, Guest and Onboard. Basic knowledge of IP networks and wide-area networking is also assumed. A general understanding and previous experience in the deployment/configuration of CheckPoint is also assumed. We also make the assumption that the firewall is already deployed. We will not cover the firewall deployment or configuration beyond the steps to integrate ClearPass and a basic policy configuration.

CPPM 6.5 Post Auth Framework Additions

Historically the Post Auth framework has provided multiple functions such as Palo Alto Networks Integration, RADIUS Accounting, Session Restrictions such as Bandwidth Usage or Multiple Device Enforcement, etc. Starting in CPPM 6.5 we released an enhancement to the Post Auth framework. This will allow ClearPass to provide an enhanced level of integration and possibly allow the enduser/customer the ability to add additional 3rd party systems without Aruba having to specifically add support for the vendor directly into ClearPass. This follows our ClearPass Exchange Framework where the flexibility of ClearPass Exchange encourages and allows for the integration into 3rd party systems supporting HTTP RESTful frameworks.

 The new Enforcement Profile is a '**Session Notification Enforcement**'. Through the next few sections we will cover this in detail. This new Enforcement Profile enhances the functionality we previously provided within the 'Session Restriction Enforcement Profile'.




The screenshot shows the 'Enforcement Profile' configuration page in the ClearPass management console. The page is titled 'Configuration » Enforcement' and 'Enforcement Profile'. There are two tabs: 'Profile' (selected) and 'Attributes'. The 'Profile' tab contains the following fields:


- Template: (empty)
- Name: (empty)
- Description: (empty text area)
- Type: Post_Authentication
- Action: Accept Reject Drop
- Device Group List: (empty list area with 'Remove', 'View Details', and 'Modify' buttons)

A dropdown menu is open, showing the following options:

- Agent Enforcement
- CLI Based Enforcement
- ClearPass Entity Update Enforcement
- Generic Application Enforcement
- HTTP Based Enforcement
- SNMP Based Enforcement
- Session Notification Enforcement** (highlighted with a blue bar and a checkmark)
- Session Restrictions Enforcement
- TACACS+ Based Enforcement

Figure 2 - New Session Notification Enforcement Profile

 **Note:** When you migrate a system from a previous CPPM 6.x software level (6.4.x and lower), if previously you had configured integration with a Palo Alto Networks firewall, this would have been defined using a Session Restriction Enforcement Profile [Type – Session Check IP-Address-Change-Notify].

 These Enforcement Profiles are **NOT** supported under CPPM 6.5 and any configuration using these profiles is migrated as you upgrade to CPPM 6.5.

Configuring CheckPoint & ClearPass for REST

Configuration is required on both the ClearPass and CheckPoint nodes. For CheckPoint, some of the functionality is still in an early release program and is not due for complete GA until later in 2015, the expectation is calendar Q2 2015. Please contact CheckPoint for advice and guidance. CheckPoint has produced a document “sk104958” which also covers this topic. Within ClearPass you need a minimum s/w release of CPPM 6.5. The following sections walk you through the ClearPass and CheckPoint configuration.

ClearPass Configuration

HTTP Context Server: First you need to add a generic HTTP Context Servers Endpoint (this is the CheckPoint endpoint) at **Administration -> External Servers -> Endpoint Context Servers [Add]**. Note that there is NO Username or Password defined on the Context Server.

The screenshot shows the 'Add Endpoint Context Server' dialog box in the ClearPass Administration console. The 'Server Type' is set to 'Generic HTTP'. The 'Server Name' is '10.2.100.18' and the 'Server Base URL' is 'https://10.2.100.18'. The 'Username' and 'Password' fields are empty, highlighted with a green box and an arrow pointing to a note: 'Note NO Username/Password defined'. The 'Validate Server' and 'Bypass Proxy' options are unchecked. The background shows the 'Endpoint Context Servers' list with a table of servers and their status.

#	Server	Status
1	Enabled	Enabled
2	Enabled	Enabled
3	Enabled	Enabled
4	Enabled	Enabled
5	Enabled	Enabled
6	Enabled	Enabled
7	Enabled	Enabled
8	Enabled	Enabled
9	Enabled	Enabled
10	Enabled	Enabled
11	Enabled	Enabled
12	Enabled	Enabled
13	Enabled	Enabled
14	Enabled	Enabled
15	Enabled	Enabled
16	Enabled	Enabled
17	Enabled	Enabled
18	Enabled	Enabled
19	Enabled	Enabled
20	Enabled	Enabled
21	Enabled	Enabled
22	Enabled	Enabled
23	Enabled	Enabled
24	Enabled	Enabled
25	Enabled	Enabled
26	Enabled	Enabled
27	Enabled	Enabled
28	Enabled	Enabled
29	Enabled	Enabled
30	Enabled	Enabled
31	Enabled	Enabled
32	Enabled	Enabled
33	Enabled	Enabled
34	Enabled	Enabled
35	Enabled	Enabled
36	Enabled	Enabled
37	Enabled	Enabled
38	Enabled	Enabled
39	Enabled	Enabled
40	Enabled	Enabled
41	Enabled	Enabled
42	Enabled	Enabled
43	Enabled	Enabled
44	Enabled	Enabled
45	Enabled	Enabled
46	Enabled	Enabled
47	Enabled	Enabled
48	Enabled	Enabled
49	Enabled	Enabled
50	Enabled	Enabled
51	Enabled	Enabled
52	Enabled	Enabled
53	Enabled	Enabled
54	Enabled	Enabled
55	Enabled	Enabled
56	Enabled	Enabled
57	Enabled	Enabled
58	Enabled	Enabled
59	Enabled	Enabled
60	Enabled	Enabled
61	Enabled	Enabled
62	Enabled	Enabled
63	Enabled	Enabled
64	Enabled	Enabled
65	Enabled	Enabled
66	Enabled	Enabled
67	Enabled	Enabled
68	Enabled	Enabled
69	Enabled	Enabled
70	Enabled	Enabled
71	Enabled	Enabled
72	Enabled	Enabled
73	Enabled	Enabled
74	Enabled	Enabled
75	Enabled	Enabled
76	Enabled	Enabled
77	Enabled	Enabled
78	Enabled	Enabled
79	Enabled	Enabled
80	Enabled	Enabled
81	Enabled	Enabled
82	Enabled	Enabled
83	Enabled	Enabled
84	Enabled	Enabled
85	Enabled	Enabled
86	Enabled	Enabled
87	Enabled	Enabled
88	Enabled	Enabled
89	Enabled	Enabled
90	Enabled	Enabled
91	Enabled	Enabled
92	Enabled	Enabled
93	Enabled	Enabled
94	Enabled	Enabled
95	Enabled	Enabled
96	Enabled	Enabled
97	Enabled	Enabled
98	Enabled	Enabled
99	Enabled	Enabled
100	Enabled	Enabled

Figure 3 - Adding a Generic HTTP Context Server

Add the appropriate Server Name (IP Address), this will be translated into the Server Base URL. No Username/Password credentials are required to communicate with the CheckPoint firewall. Authentication is performed via the PSK configured in a later step.

Context Server Actions Login/Logout: Now that we have defined the Firewall endpoint, we need to set the CheckPoint context server actions '**Login & Logout**' to use this endpoint.

It's very important that you modify **both** the Login and Logout Server Actions. These are what update the Firewall of a user's session going active/de-active. ClearPass will then update the CheckPoint firewall which will permit/deny this user. We don't want to update the firewall of a session starting and not clear it when the user leaves.

Note: There is a timeout value on the firewall but ensuring that the Context Server Logout action is performed is cleaner and more secure.

Note: It's recommended that you copy the supplied/original **CheckPoint Login/Logout** context server actions templates and modify the copied items.

There are two Context server actions that need to be modified. These can be found under **Administration -> Dictionaries -> Context Server Actions** filter on 'check' with a Filter = 'Action Name'... then look for the below 'Check Point' action.

The first Context Server Action that needs to be modified is the 'Check Point Login'....

Note: The below URL path is set to `/idasdk/add-identity`. As noted earlier in this document at the time of writing this document CheckPoint had not generally released its software, and as such the url path could change between now the the time of its release.

Within the '**Action**' tab the 'Server Name' needs to be changed to that of the HTTP server you added in the previous section. The default will be localhost, select as appropriate from the drop down.

Field	Value
Server Type:	Generic HTTP
Server Name:	10.2.100.17
Action Name:	Check Point Login
Description:	Inform Check Point that user logged in
HTTP Method:	POST
Skip HTTP Auth:	<input type="checkbox"/> Enable to skip HTTP Basic Authentication
URL:	/idasdk/add-identity

Figure 4 - Context Server - Action Tab

Within the '**Header**' tab, nothing needs to be changed beyond the supplied default. Below is a copy of the parameters in the default context server for your reference.

#	Header Name	Header Value
1.	Content-Type	= application/json
2.	Click to add...	

Figure 5 - Context Server - Header Tab

Within the **'Content'** tab, a couple of parameters might need to be modified. Note that as well as the substitutional variables we send to the CheckPoint firewall, the identity-source attribute is set permanently as 'Aruba ClearPass Policy Manger'. There are several other important parameters within this section, shared-secret, session-timeout, calculate-roles, fetch-user-groups and fetch-machine-groups. The timeout setting has been set to 28800 seconds (8 hours), you can change this if required, however the logout context-server-action is the action that should deal with logging out Users from the firewall. With the new API, it is required to specify "fetch-user-groups" and "fetch-machine-groups" (and set them both to "0") if you want to make sure the identity will not be verified against CheckPoints identity sources. Note that "fetch-machine-groups" is only relevant for machine identities.

Note: Now, this is very important for **6.5 and Guest Users**. When using the CheckPoint Identity Awareness feature (RESTful API or RADIUS Accounting) the userID that is received by the firewall typically has to be verifiable as a valid user. CheckPoint will ensure the user exists within an authoritative Identity Store, like Active Directory. Obviously for Guest users their userIDs do not exist as they are transient users. Some guest accounts could exist within a directory but that is not usual. So, as part of the integration we have to identify these users and link them to a user group (a configurable CheckPoint attribute called access role). The other really important attribute below in the Content tab is the calculate-roles and fetch-user-groups. When we are posting a guest userID to CheckPoint these values need to be set to '0', that's a zero.

Details of the above is covered in detail in the section [Check Point Configuration – Where the UserID is a Guest account](#) starting on page 22.

To achieve a full integration between CPPM and CheckPoint, you **may** have to have **TWO** Context-Server's definitions, each definition will reference different Context-Server-Action, i.e. one group of configurations for AD/verifiable users the other definition for Guests.

The screenshot shows the 'Endpoint Context Server Details' window with the 'Content' tab selected. The 'Content-Type' is set to 'JSON'. The 'Content' field contains the following JSON object:

```
{
  "shared-secret": "arubans123",
  "user": "%{name}",
  "ip-address": "%{ip}",
  "machine": "%{machine}",
  "identity-source": "Aruba ClearPass Policy Manger",
  "session-timeout": 28800,
  "user-groups": ["aruba-guest"],
  "machine-groups": ["aruba-guest-machine"],
  "roles": [],
  "fetch-user-groups": 0,
  "fetch-machine-groups": 0,
  "machine-os": "%{device_family}",
  "host-type": "%{device_type}"
}
```

Figure 6 - Context Server – Content Tab

Note that in the above where I have highlighted **device_family** & **device_type**, these attributes are ONLY exposed in CPPM 6.5.3 and later.



Within the '**Attributes**' tab, nothing is configured. A copy for your reference is below.

#	Attribute Name	Attribute Value
1.	Click to add...	

Figure 7 - Context Server - Attributes Tab

To check that all of the Context Server actions have been successfully configured, if you return to the Endpoint Context Server, and look on the 'Actions' tab, you should see the actions (hopefully you will have made a copy of the default server actions) listed against your endpoint, ensure you see **Login** and **Logout** as shown in our example below.

Name	Description
LAB-Check Point Login	Inform Check Point that user logged in.
LAB-Check Point Logout	Inform Check Point that user logged out.

Figure 8 - Check Login & Logout against CheckPoint endpoint

Enforcement Profile: Finally, a **Session Notification Enforcement profile** must be configured. The Session Notification profile is a new item in CPPM 6.5. Below is an example of a configured profile. Note that there are four **session-notify** attributes you must add to the profile to make it complete, failure to add all four will certainly ensure your integration will fail...!!!



1. **Server-Type** [Generic HTTP]
2. **Server IP** [IP address of previously defined HTTP Context Server]
3. **Login Action** [Configured and assigned to the Context Server Login Action]
4. **Logout Action** [Configured and assigned to the Context Server Logout Action]

Configuration » Enforcement » Profiles » Edit Enforcement Profile - CP-test-danny

Enforcement Profiles - CP-test-danny

Summary Profile **Attributes**

Type	Name	Value
1. Session-Notify	Server Type	= Generic HTTP
2. Session-Notify	Server IP	= 199.209.74.88
3. Session-Notify	Login Action	= Check Point Login
4. Session-Notify	Logout Action	= Check Point Logout
5. Click to add...		

Figure 9 - 'Session Notification' Enforcement Profile

Once the above configuration has been completed the normal process of applying the context server actions to an enforcement profile should be followed. Then, following a 'standard' network authentication (e.g. dot1x) CPPM will post the userID attributes etc. to the firewall, the posting is pretty much real-time. UserID should appear within 2 seconds.

NOTE: ENSURE YOU SET THE SHARED SECRET ON THE Check Point **LOGOUT** ACTION.

Endpoint Context Server Details

Action Header Content Attributes

Server Type: Generic HTTP

Server Name: [Redacted]

Action Name: Check Point Logout

Description: Inform Check Point that user logged out

HTTP Method: POST

Skip HTTP Auth: Enable to skip HTTP Basic Authentication

URL: /idasdk/delete-identity

Figure 10 - Check Point Logout - Action Tab

Note: The above URL may change when the CheckPoint software supporting this integration is fully released.

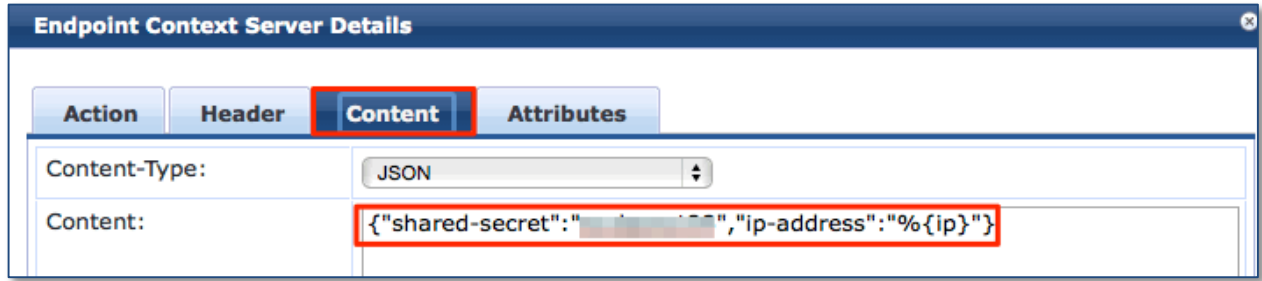


Figure 11 - Check Point Logout – Content Tab

Update to CheckPoint Configurations to support REST in R77.xx Product



Following the release of this document (1.0 & 1.1) we discovered that the release vehicle for the RESTful functionality had changed. At the time of testing and integration we (ARUBA) believed the CheckPoint plan was to release a new major code update that included this functionality. Subsequently Check Point released a number of hotfixes to their existing R77.xx code train and made the functionality available that way. This has required us to update and re-release this TechNote. Note that after this section are the instructions that will be appropriate when CheckPoint releases the feature beyond the HOTFIX support.

With the update of this TechNote we captured the changes required to interoperate with the R77.xx code train. In summary these changes are fairly subtle, and listed below.

1. Change the URL path configuration on ClearPass when posting information to the Check Point firewall
2. A minor change to enable the RESTful API in the GUI on the CheckPoint Gateway

but the key take away is that **none** of the functionality is reduced.

URL Update - After installing the R77.xx hotfix which can be obtained directly from Check Point you must use a slightly different path for the HTTP POST from what we initially documented and continue to be shown/documented later in the document. We are leaving the old documentation content for now as is, following conversations with CheckPoint we expect them to release an updated software product where the documented in the following sections will be inline with that product.

The URL format as shown below adds ***/_IA_MU_Agent*** before the ***/idasdk/<command>***. Within ClearPass update the Context Server if you've previously defined the URL path as shown below in Figure 13 below.

Endpoint Context Server Details			
Action	Header	Content	Attributes
Server Type:	Generic HTTP		
Server Name:	10.2.100.18		
Action Name:	Check Point Login		
Description:	Inform Check Point that user logged in		
HTTP Method:	POST		
Skip HTTP Auth:	<input type="checkbox"/> Enable to skip HTTP Basic Authentication		
URL:	<i>/_IA_MU_Agent/idasdk/add-identity</i>		

Figure 12 - New Check Point URL POST path (R77.xx + RESTful HOTFIX)

Below for clarity and comparison is the URL path we previously configured, this might represent the path you have configured in your environment. To interoperate with a CheckPoint R77.xx node with the RESTful HOTFIX will require that you amend this URL path as shown above in Figure12.

HTTP Method:	POST
Skip HTTP Auth:	<input type="checkbox"/> Enable to skip HTTP Basic Authentication
URL:	/idasdk/add-identity

Figure 13 - Old Check Point URL POST path

Ensure that you amend the context server actions for normal users and guest-users and that the logout as well as the login actions are updated.

Enable REST API Integration - After the HOTFIX has been applied, the process to enable the RESTful integration and define the API PSK will need to be configured on CheckPoint, unlike in the fully-released version where there is an actual panel to enable and configure the API Service. (As shown below later in Figure 17). To complete the configuration for this section, turn on Terminal Servers (if not already enabled) and configure a shared-secret.

This will be the PSK password that is configured to match within ClearPass when defining the Context Server Actions, once configured its shows as highlighted as the example below.

The screenshot shows the 'Identity Awareness' configuration page. On the left, a tree view has 'Identity Awareness' selected. The main panel shows several identity sources:

- Browser-Based Authentication (Settings...)
- Active Directory Query (Settings...)
- Identity Agents (Settings...)
- Terminal Servers (Settings...)
- RADIUS Accounting (Settings...)
- Remote Access

Under 'Terminal Servers', the text reads: 'Using the following pre-shared secret: arubans123'. A red box highlights this text, and a red arrow points to the secret value.

Figure 14 - Enabling TS (interim placeholder for configuring RESTful API password)

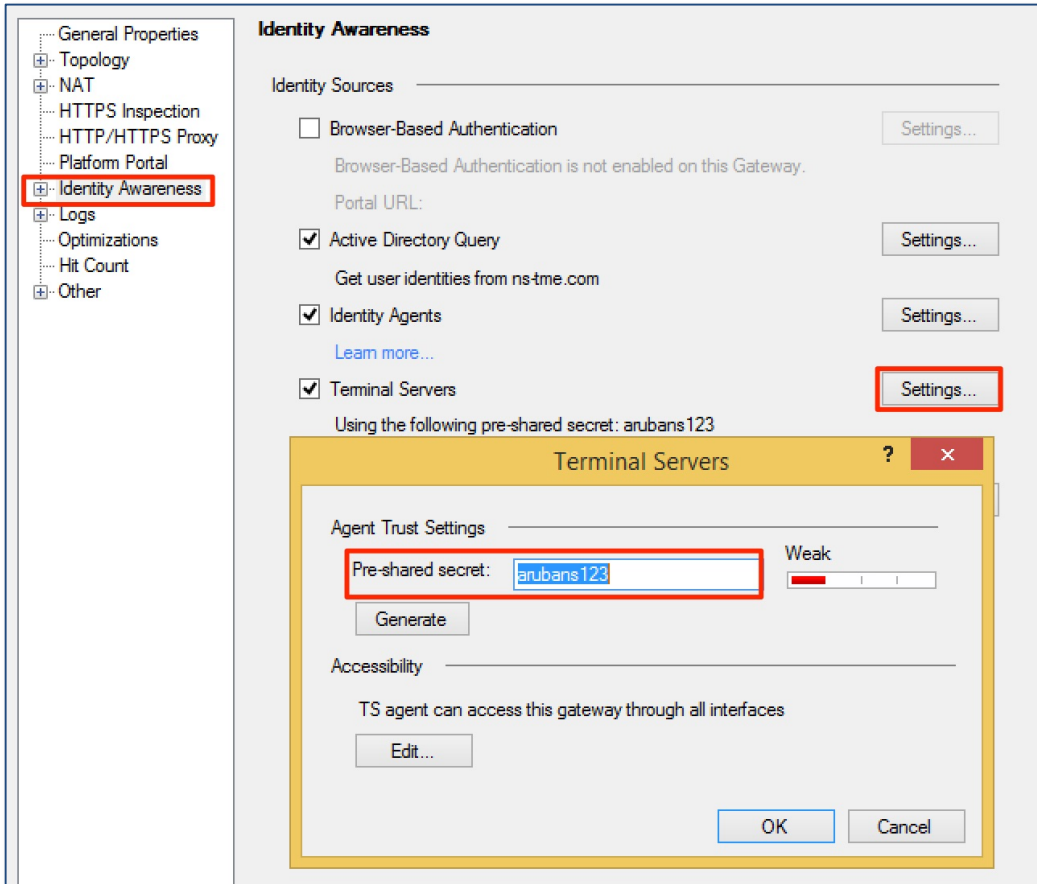


Figure 15 – Setting the PSK in TS to be used by RESTful API service

You may want to ensure that the correct interfaces are enabled to accept request for the RESTful API POST. As shown above under the 'Accessibility' option, certain interface can be controlled to allow/deny this interaction.

Following the above updates, you can continue with your configuration below on Page17 @ [Adding AD/LDAP Server.](#)

Check Point Configuration – Where the UserID is verifiable

As previously noted, the software we tested with is a not GA until later in 2015. Note that the configuration shown below could change when the software is fully released.

Signin to the CheckPoint firewall with the SmartDashboard management application.

Configure Identity-Awareness API Service The first item of configuration required is that the Identity Awareness (IA) **API Service** is enabled and the pre-shared secret is configured.

From the 'Network Objects - Checkpoint' section, select (double-click) the enforcement point you want to configure, in our case the firewall is the **IArest** object as shown below.

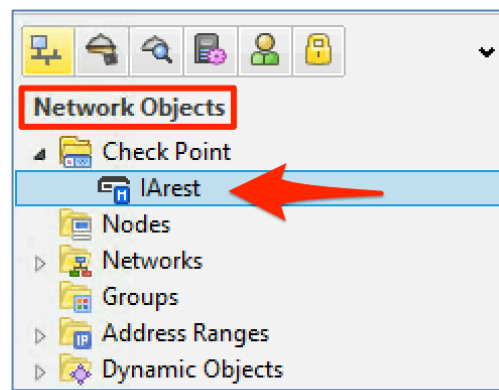


Figure 16 - Select the firewall to be configured

From the following configuration screen ensure that '**API Service**' is selected to enable the feature. Then click on Settings to generate or set the PSK that will be configured in CPPM.

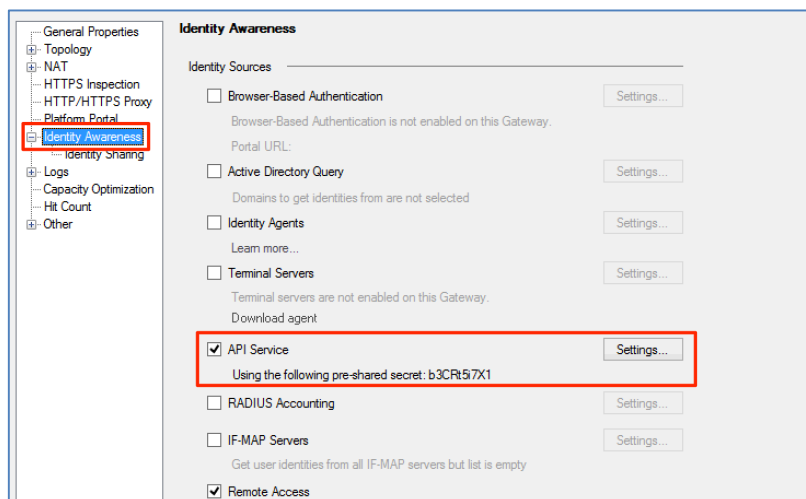


Figure 17 - Setting the RESTful PSK between CPPM and Check Point

Note: that the **API Service** option shown above is a new option expected to appear in a later product version. As documented above for now configure the PSK under TS.

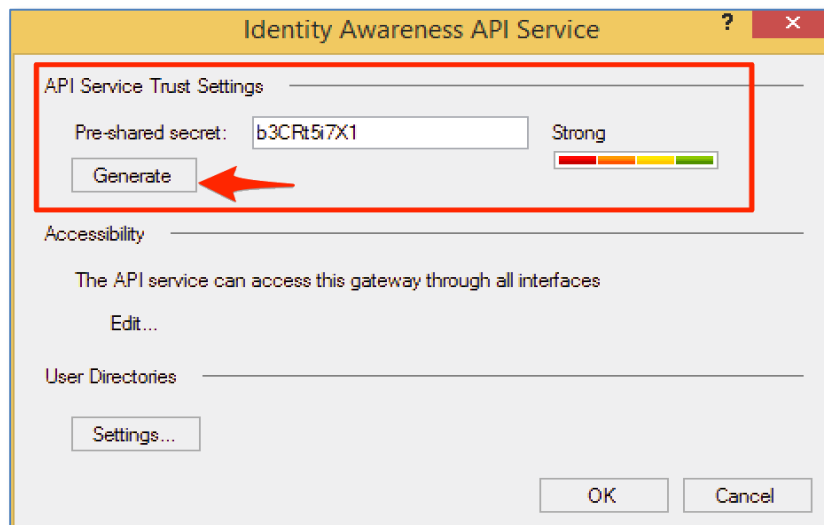


Figure 18 - Set or Generate the IA PSK

You can use the ability of the application to randomly generate a PSK or you can choose to configure your own. The pre-shared secret configured here is what must match that of the Login, Logout context server actions configured in CPPM.

Adding AD/LDAP Server If you need to add AD servers to CheckPoint, again edit the CheckPoint firewall object. If you select/un-select **Identity Awareness**, the following screen will be shown where you can use the AD Query wizard to add your AD Server to the Check Point firewall.

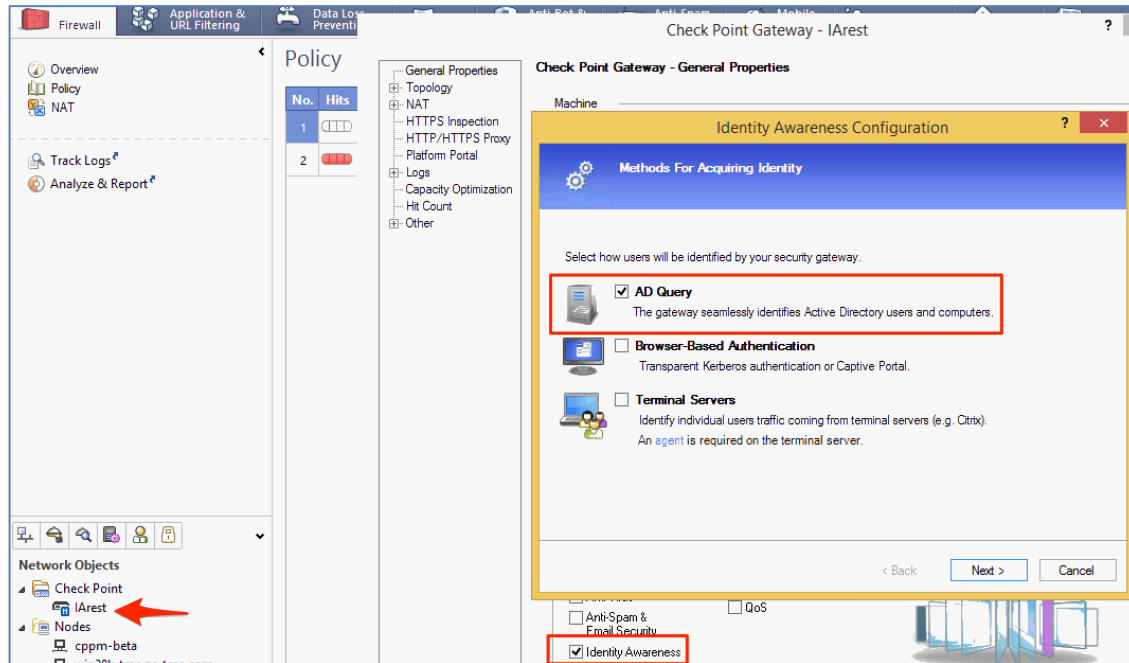


Figure 19 - Enabling Active Directory 'Addition' wizard

Q? - Why would I want to do this? Well, the userID details that are sent by CPPM need to either match a user-group or be verifiable with some other identity source, i.e. AD.

Add the required details, admin account, FQDN, etc. to add an AD to the CheckPoint firewall. Take notice of the successful completion message at the bottom of the wizard.

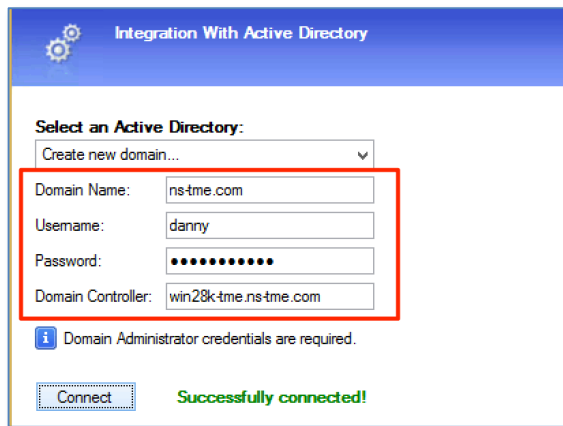


Figure 20 - Adding AD Server to Check Point

Once the AD server has been added, you can check this is successful by navigating to the Users and Administrators tab (shown below). The AD server configured should show up and you can expand it to see the configured DN.

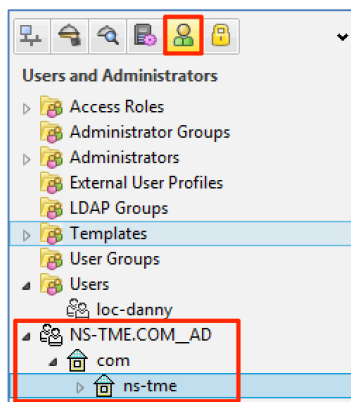


Figure 21 - Checking AD is added to Check Point

Finally, you can see if the users are available to CheckPoint by writing a Policy rule referencing one of the expected AD users. It's a good way to be sure AD was added correctly and that the CheckPoint firewall can read the AD directory.

Add a new rule, for the Source, right click and '**Add User/Access role**'. Then select the **Users** tab, then '**Source users/groups**' and the **green +** to bring up the query box where you can enter a user-name. Below we set this to 'danny' for the purpose of this configuration step. If this user is located from the search of AD it will be shown in the results-TME search. This rule will now match on a user where the source traffic is from a username matching 'danny', and in this case referenced and validated back to AD.

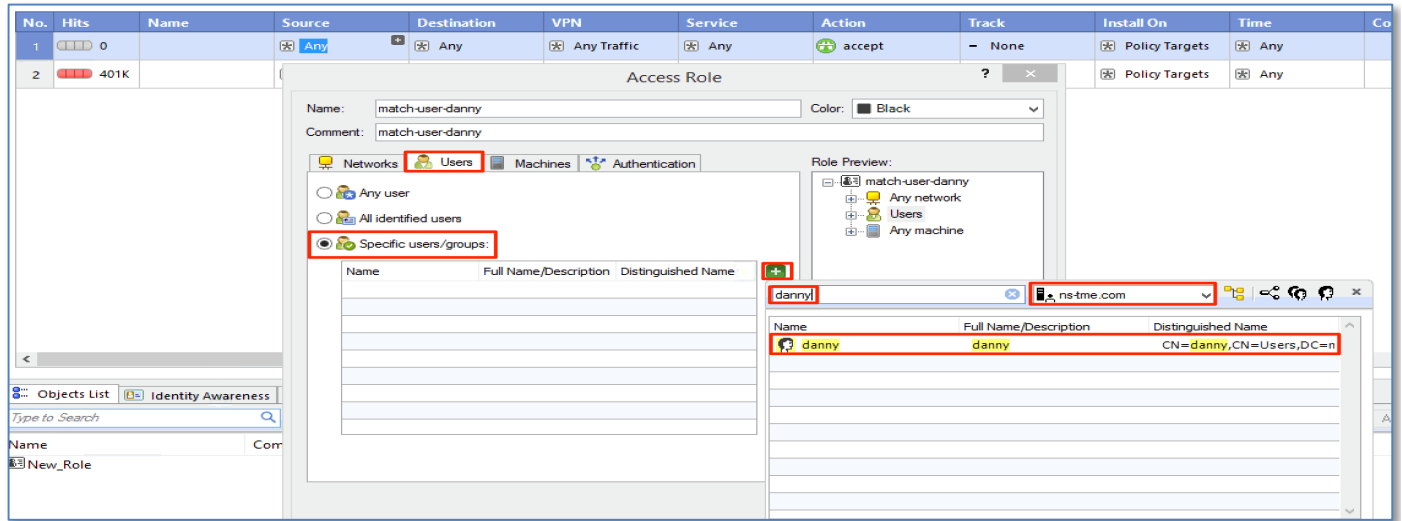


Figure 22 - Adding a firewall rule to reference a username

Finally, you need to install the policy, click on **'Install Policy'** from Smart Dashboard, to install and activate the configuration on the firewall blade.

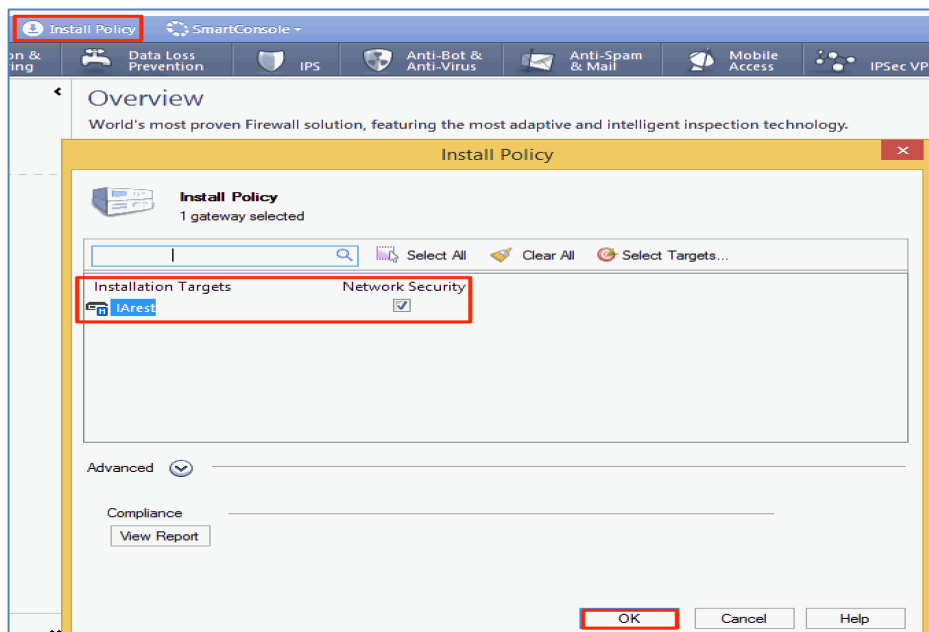


Figure 23 - Install Firewall Policy

In the screen above, you can see the policy being installed on the firewall IArest.

Below you can see the policy being verified and the installation progress bar, advancing.

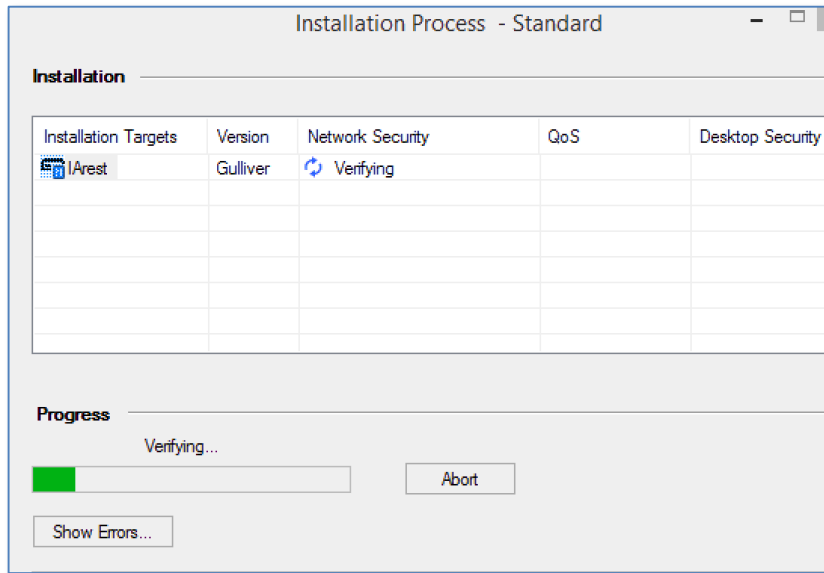


Figure 24 - Installation of Policy in Progress

At the end of the installation you should see a screen similar to the below, showing a successful installation of the firewall policy.

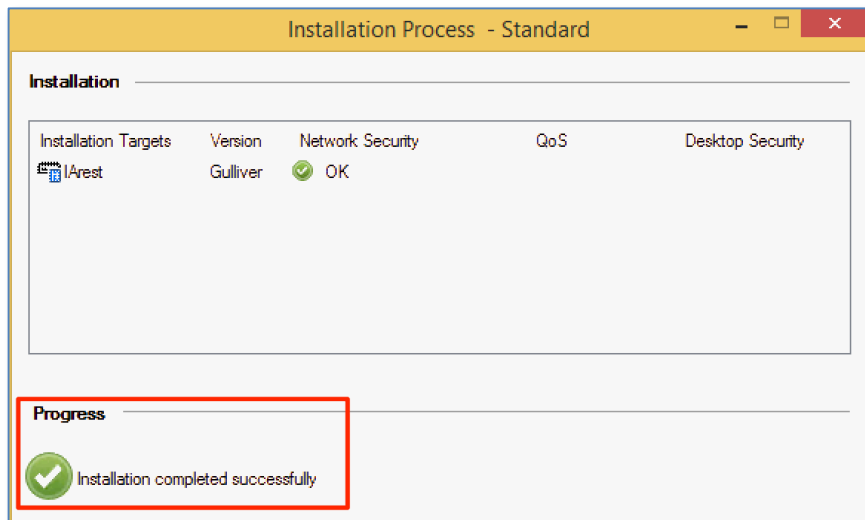


Figure 25 - Successful Installation of the Firewall Policy

Assuming the above configuration is complete, if we load the CheckPoint SmartView Tracker, and filter using the Identity Awareness Blade we can look at the authenticated users on the system. Below you can see (if you squint..!!) the user djump authenticated and the Identity Source shows as 'Aruba ClearPass Policy Manager'.

The screenshot displays the SmartView Tracker interface with the Identity Awareness Blade selected in the left-hand navigation pane. A record for user 'djump' is highlighted in the main table. The record details pane shows the following information:

- Log Info:** Product: Identity Awareness, Date: 9Feb2015, Time: 17:28:14, Number: 263422, Type: Log, Origin: IArest.
- Traffic:** Source: 10.2.100.169, Service: ---, Protocol: ---, Interface: ---, Source Port: ---.
- User Information:** Source User Group: All Users, Source Machine Group: ---, Associated Roles: ---.
- Log In:** Action: Log In, Authentication Status: Successful Login, Authentication Method: User/Machine Authentication (Identity Awareness API), Description: ---, Identity Source: Aruba ClearPass Policy Manager.

The main table shows a list of records with columns for No., Date, Time, Source, Src. User Gr..., Src. Machine..., Src. User Gr..., Source Ma..., Description, Auth. Meth..., Identity Src., and Auth. Status. The record for 'djump' is highlighted, showing a successful login from source 10.2.100.169.

Figure 26 - AD user authenticated on Check Point

Note that the “user-groups” and “machine-groups” parameters can take a comma-separated list, it’s not mentioned above as our above example shows one group only.

So this is all good if the user belongs to an identity directory where CheckPoint is able to verify the user. Next we cover the scenario where the user is essentially a Guest and CheckPoint cannot verify the user’s identity beyond what ClearPass sends.

Check Point Configuration – Where the UserID is a Guest account

For a userID sent by ClearPass where the CheckPoint firewall is unable to verify the account, we need to complete some additional configuration. We need to basically tell CheckPoint that the userID belongs to a group that CheckPoint is aware of (i.e. a Guest group). This requires us to pre-create the group and then have ClearPass include the group information when we post the userID to the Check Point firewall. In the below, we have tried to authenticate a user 'qwerty'. In this example the user is a locally defined user in ClearPass, but is not known to Active Directory. The CheckPoint firewall tried to verify the userID exists but fails. From the log you can see this user is rejected. In essence this simulated a guest (i.e. an unknown/unverifiable account) being denied by CheckPoint.


The screenshot displays a log entry for a rejected login attempt. The log is titled 'Identity Awareness Reject' and shows the following details:

Log Info		Reject	
Product	Identity Awareness	User Information	
Date	9Feb2015	Source User Group	---
Time	19:00:27	Source Machine Group	---
Number	263545	Associated Roles	---
Type	Log	More	
Origin	IArest	Authentication Method	User/Machine Authentication (Identity Awareness API)
Traffic		Authentication Status	Failed Login
Source	10.2.100.169	Description	Group membership of the required account (user or machine) could not be retrieved from the AD. Make sure the account exists in the AD. User:
	qwerty	Product Family	Network
Service	---	Identity Source	Aruba ClearPass Policy Manager
Protocol	---	Information	Authentication trial: this is a reauthentication for session
Interface	---		
Source Port	---		

Figure 27 - non-AD user rejected by Check Point

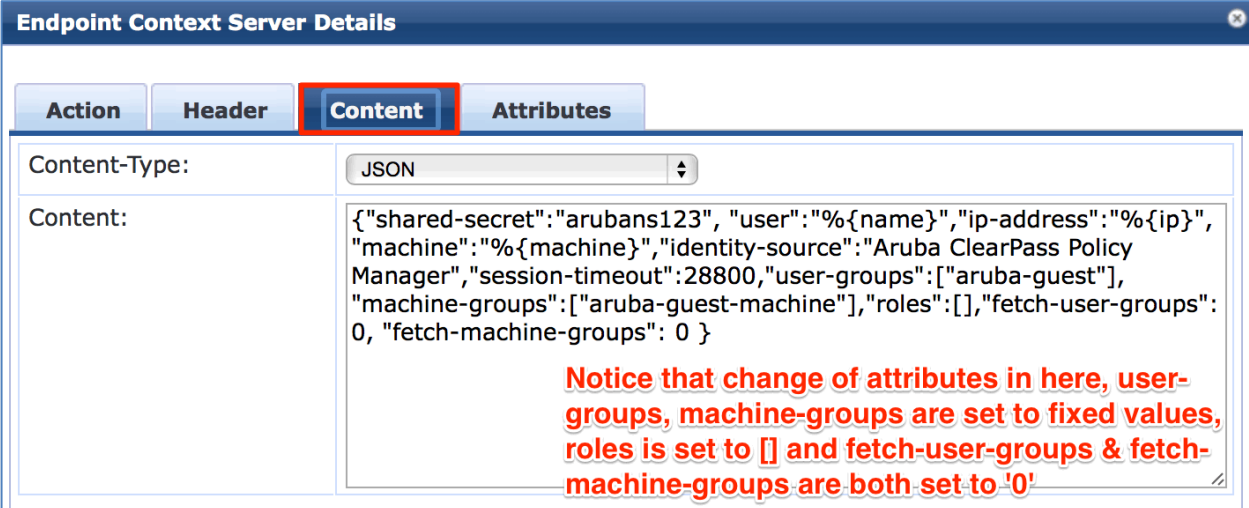
To support Guest users, we need to make changes to the context-server-actions to make it clear to the CheckPoint firewall that the userID sent from ClearPass is a GUEST and the firewall must not try to verify the userID but take it as being a known and trusted user as sent by ClearPass.

A new Context Server Action will have to be created specifically for Guests users accounts, then modify the **Content** Tab as required. In the example below we define a **user-groups** in the context-server-action called **aruba-guest**, this group will have to be created on the CheckPoint firewall. Take care as no validation is performed and the spelling needs to be **exactly** the same. The creation of this item on the firewall is covered later in the document.

 **Note:** The user-groups attribute shown below in the 'CONTENT' tab is NOT included in the initial 6.5 code we released. You will have to **manually** add this and configure it as appropriate. Note that following the release of ClearPass 6.5.3 this is not included.

The important points to call out below are that we have added some additional fields, **user-groups**, this is set in our example to **aruba-guest**. We have also we added **machine-groups**, this is set in our example to **aruba-guest-machine**, we also added a field called **roles**. Note that the roles field is set to **[]**, that's a left and right square bracket.

The final change is that we have modified the two group fields, **fetch-user-groups** and **fetch-machine-group** to a **0**, that's a zero.



The screenshot shows the 'Endpoint Context Server Details' window with the 'Content' tab selected. The 'Content-Type' is set to 'JSON'. The 'Content' field contains the following JSON configuration:

```
{
  "shared-secret": "arubans123",
  "user": "%{name}",
  "ip-address": "%{ip}",
  "machine": "%{machine}",
  "identity-source": "Aruba ClearPass Policy Manager",
  "session-timeout": 28800,
  "user-groups": ["aruba-guest"],
  "machine-groups": ["aruba-guest-machine"],
  "roles": [],
  "fetch-user-groups": 0,
  "fetch-machine-groups": 0
}
```

Notice that change of attributes in here, user-groups, machine-groups are set to fixed values, roles is set to [] and fetch-user-groups & fetch-machine-groups are both set to '0'

Figure 28 - Context Server Action for Guest users where the user_groups attribute is set

The **user-groups** setting we POST from ClearPass must match configuration on the CheckPoint firewall. The configuration must match an **access-role**, of the same name, configure this under **Users and Administrator**, then **Access-Roles** as shown below.

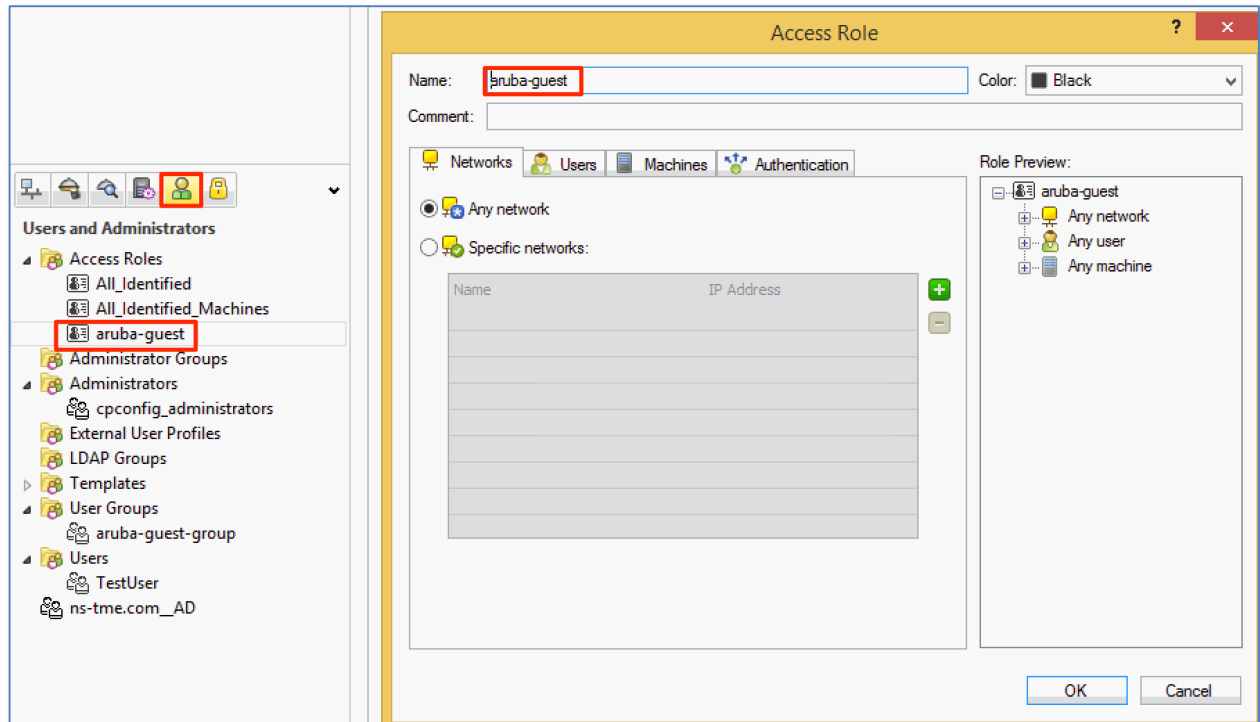


Figure 29 - Adding an Access-Role to match the Aruba 'guest'role'

If you want to add an additional level of control to the access-role, you can add a User Group as shown below. This is an optional step.

Note: The naming of the **Group** must be different from the **Access-Role**. You cannot have multiple items of the same name, even if they are configured in different functional areas of the CheckPoint firewall. Create the User-Group under **Users and Administrator**. In the below example we call out Group, aruba-guest-group.

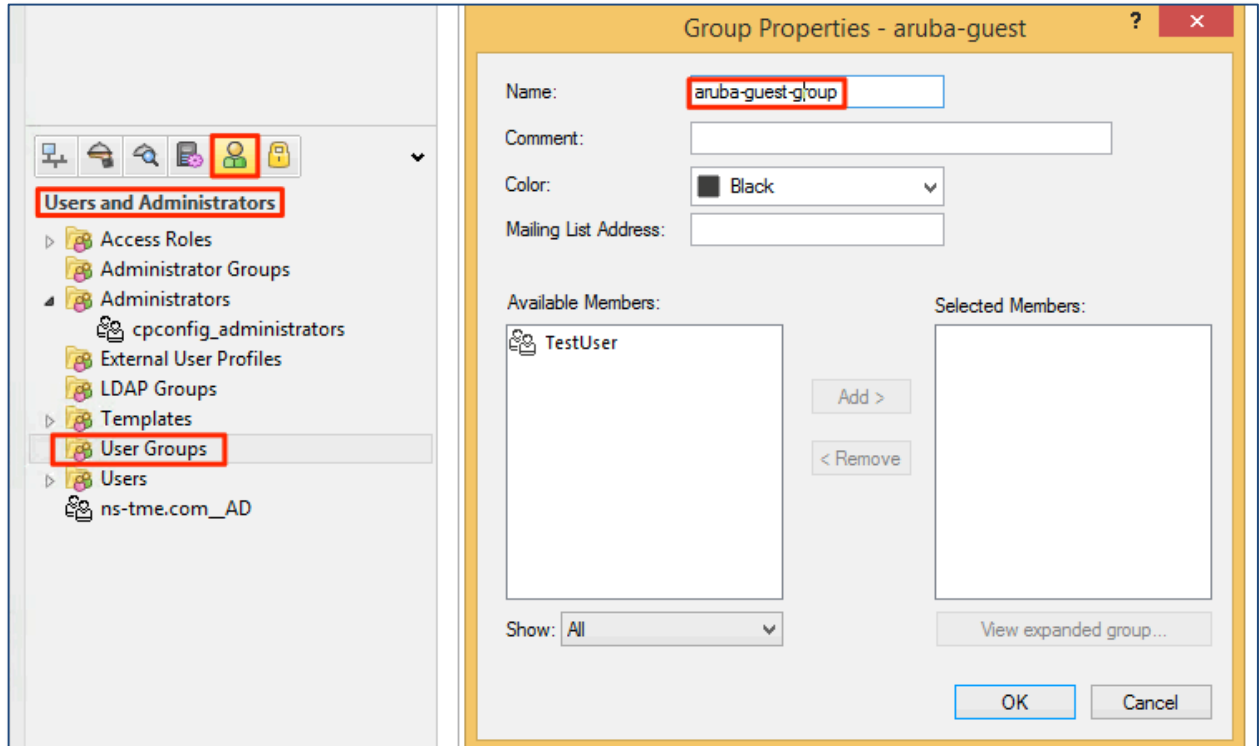


Figure 30 - Adding a 'Group' to match 'role' sent from ClearPass

Following the creation of the Group, you can add this to the existing **Access Roles**. The Access Role **aruba-guest** is configured can to match specific Users. Note that the aruba-guest-group is under **'Internal User Groups'**. Follow the below configuration example.... **Access Role, Users, Specific users/groups, [add a user/group], Internal User Groups, [select User Group]** which was just configured.

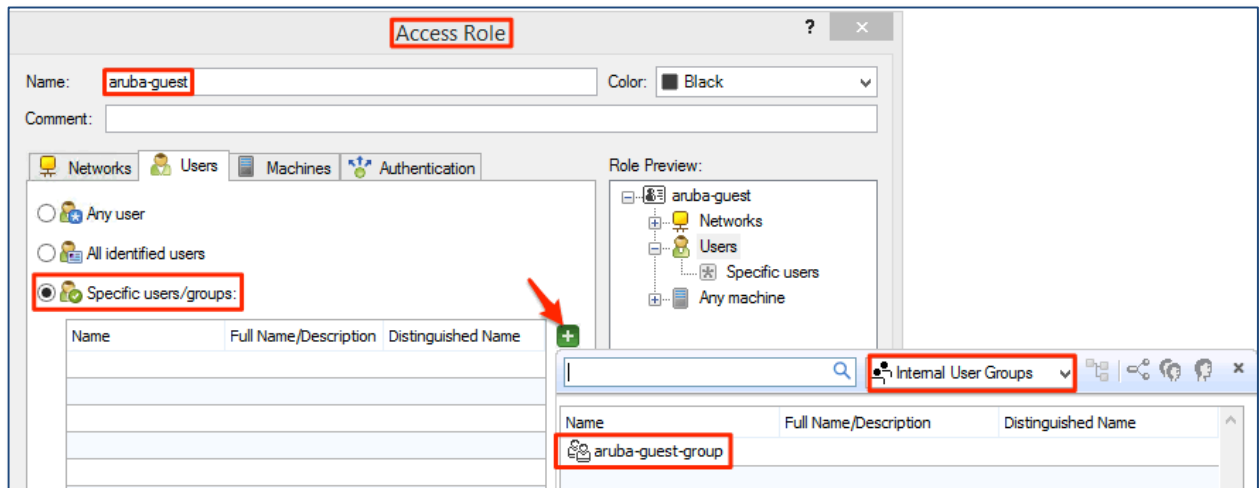


Figure 31 - Adding an Access Role to use User Groups

We can see from the SmartView Tracker below, guest-user 'djj' being authenticated and labeled with a role of aruba-guest.

The screenshot displays the SmartView Tracker interface for Identity Awareness. It is divided into several sections:

- Log Info:** Product: Identity Awareness, Date: 3Apr2015, Time: 20:15:10, Number: 85243, Type: Log, Origin: restapi-gw.
- Log In:** Action: Log In, Authentication Status: Successful Login, Authentication Method: User/Machine Authentication (Identity Awareness API), Description: ---, Identity Source: Aruba ClearPass Policy Manager.
- Traffic:** Source: 10.2.100.167, User: djj, Identity: dannysipadmini.
- User Information:** Source User Group: aruba-guest, Source Machine Group: aruba-guest-machine, Associated Roles: aruba-guest.
- More:** Product Family: Network, Information: ---.

Figure 32 - Guest user being 'labeled' with access tag

Now that we have users being accepted and labeled in CheckPoint, Policy can be used to reference the Guest users by the **aruba-guest** label. So in the below example, I created an address-range object for our 10.0.0.0 internal network. In the simple rule below I used this to deny guests effectively accessing any of our corporate network objects.

No.	Hits	Name	Source	Destination	VPN	Service	Action	Track	Install On	Time
1	17K		match-user-dar	Any	Any Traffic	Any	accept	None	Policy Targets	Any
2	0		aruba-guest	internal-10-ran	Any Traffic	Any	drop	None	Policy Targets	Any
3	672K		Any	Any	Any Traffic	Any	accept	Log	Policy Targets	Any

Figure 33 - Firewall Policy denying access to corporate resources

This completes the Check Point firewall gateway configuration to enable the RESTful integration between ClearPass and CheckPoint.

Monitoring/Debugging Identity Awareness from the CLI

Several very useful commands are available to monitor the role/group associated of users that are being sent from ClearPass Policy manager.

You need to access the cli, generally this is done via a SSH command.

Once you have access the following command you may find useful.

pdp monitor ip a.b.c.d

```
[Expert@restapi-gw:0]# pdp m ip 10.2.100.167

Session: 473d7552
Session UUID: {5EB93636-3FDC-7955-608E-5BCCCE7DFDA1}
Ip: 10.2.100.167
Machine:
  dannysipadmini {5bb0e37f}
  Groups: aruba-guest-machine
  Roles: aruba-guest
  Client Type: Identity Awareness API (Aruba ClearPass Policy
Manager)
  Authentication Method: Trust
  Connect Time: Fri Apr 3 20:21:56 2015
  Next Reauthentication: Fri Apr 3 20:50:29 2015
  Next Connectivity Check: -

Users:
  djj {2fa55b51}
  Groups: aruba-guest
  Roles: aruba-guest
  Client Type: Identity Awareness API (Aruba ClearPass Policy
Manager)
  Authentication Method: Trust
  Connect Time: Fri Apr 3 20:21:56 2015
  Next Reauthentication: Sat Apr 4 04:22:26 2015
  Next Connectivity Check: -

Packet Tagging Status: Not Active
Published Gateways: Local
*****
```

Figure 34 - PDP Debug/Monitor output for an IP address

The above shows the output for the user and machine authentication and the group/role assignment.

To check just on a users role/grouping the below command provides that insight.

pdp m user [user-name]

```
[Expert@restapi-gw:0]# pdp m user djj

Session: 473d7552
Session UUID: {5EB93636-3FDC-7955-608E-5BCCCE7DFDA1}
Ip: 10.2.100.167
Users:
  djj {2fa55b51}
    Groups: aruba-guest
    Roles: aruba-guest
    Client Type: Identity Awareness API (Aruba ClearPass Policy
Manager)
    Authentication Method: Trust
    Connect Time: Fri Apr 3 20:21:56 2015
    Next Reauthentication: Sat Apr 4 04:22:26 2015
    Next Connectivity Check: -

Packet Tagging Status: Not Active
Published Gateways: Local
*****
```

Figure 35 - PDP Debug/Monitor for a username

Configuring Radius Accounting Proxy

An additional integration method to support 3rd Party vendors was also added to the CPPM 6.5 release. We support the ability to configure a RADIUS Accounting Proxy. This allows CPPM to proxy the RADIUS accounting data that is received to an external system such as an external firewall or SIEM. When CPPM processes an authentication, as part of the session configuration on CPPM a RADIUS Accounting Proxy target can also be configured. This allows CPPM to forward the interim accounting updates it receives from the NAS to this external target.

Configuring RADIUS Accounting on ClearPass

Configure Accounting Proxy on CPPM as shown below. First configure your targets, under **Configuration-> Network -> Proxy Targets** just like previously as if you were configuring RADIUS authentication proxy-ing.

The screenshot displays the 'Proxy Targets' configuration page in the ClearPass management console. The breadcrumb navigation 'Configuration » Network » Proxy Targets' is highlighted with a red box. Below the navigation, there is a filter section with 'Name' selected and a search box. A table lists existing proxy targets, including 'checkpoint-pr' and 'fortinet-proxy'. An 'Add Proxy Target' dialog box is open, showing fields for Name, Description, Hostname, Shared Secret, Verify Shared Secret, RADIUS Authentication Port, and RADIUS Accounting Port. The 'Hostname' field is highlighted with a red box and contains the IP address '1.1.1.1'. A red arrow points to the 'Save' button at the bottom of the dialog box.

#	Name	Hostname	Description
1.	checkbox	checkbox	checkbox
2.	checkbox	checkbox	checkbox

Showing 1-2 of 2

Filter: Name contains [] Go Clear Filter

Add Proxy Target

Name: test-proxy
Description: []
Hostname: 1.1.1.1
Shared Secret: []
Verify Shared Secret: []
RADIUS Authentication Port: 1812 (Default is 1812)
RADIUS Accounting Port: 1813 (Default is 1813)
Save Cancel

Figure 36 - Adding a PROXY Target

By default the Account Proxy Tab is not shown, you must enable it in the Service Definition as highlighted below.

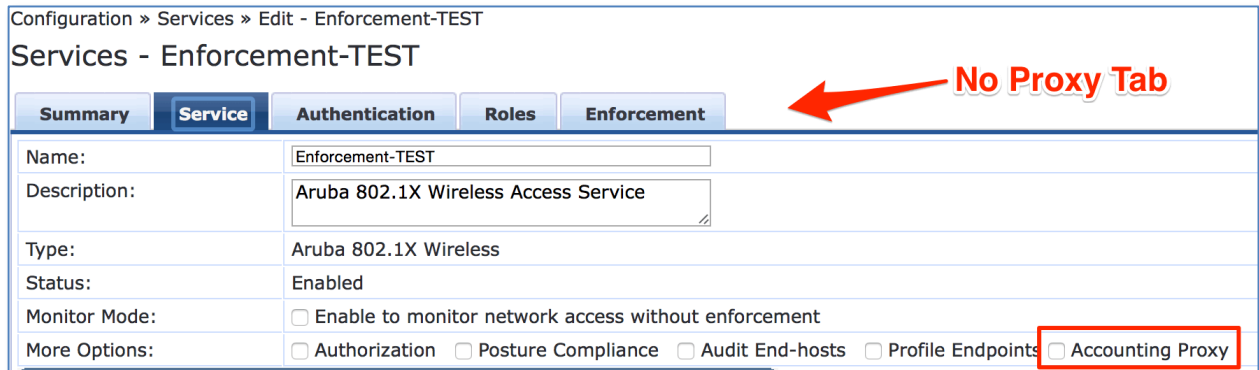


Figure 37 - Enabling Accounting Proxy Configuration

Now that the target is defined and the Accounting Proxy is enabled the remaining configuration can be completed.

Below is an example of adding the Accounting Proxy to the above service definition. We added a proxy target from one of the targets configured in the previous step.

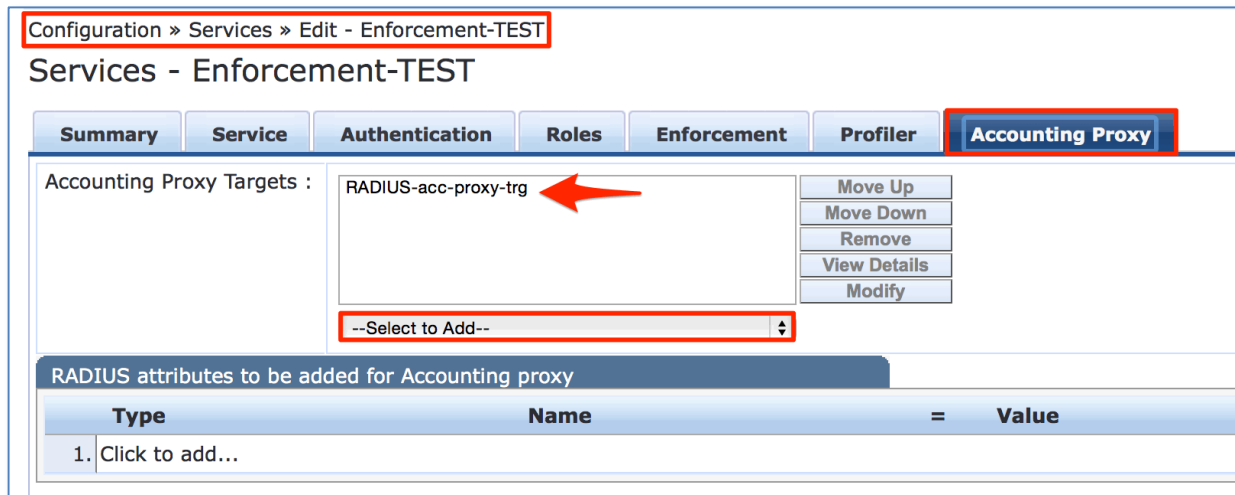


Figure 38 - Configuring Accounting Proxy on an example service



Note: Whatever RADIUS accounting data we receive from the NAS will be forwarded to the Accounting Proxy target. We can also add VSA and IETF standard attributes to the data we forward. However, to add a VSA we must have the Dictionary's for that vendor's product installed/enabled within CPPM. Some vendors have multiple RADIUS Dictionaries across their product ranges, so just because we have one for company X does not mean it will encompass all their products and the VSA's they support.

Configuring RADIUS Accounting on a Checkpoint Firewall

Configure Identity-Awareness RADIUS Accounting The first item of configuration required is that the Identity Awareness **RADIUS Accounting** is enabled

From the 'Network Objects - Checkpoint' section, select (double-click) the enforcement point you want to configure, in our case the firewall is the **IArest** object shown below.

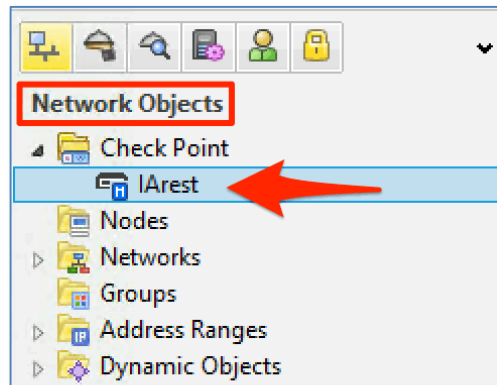


Figure 39 - Select the firewall to be configured

From the following configuration screen ensure that '**RADIUS Accounting**' is selected to enable the feature. Then click on Settings to configure the RADIUS accounting attributes.

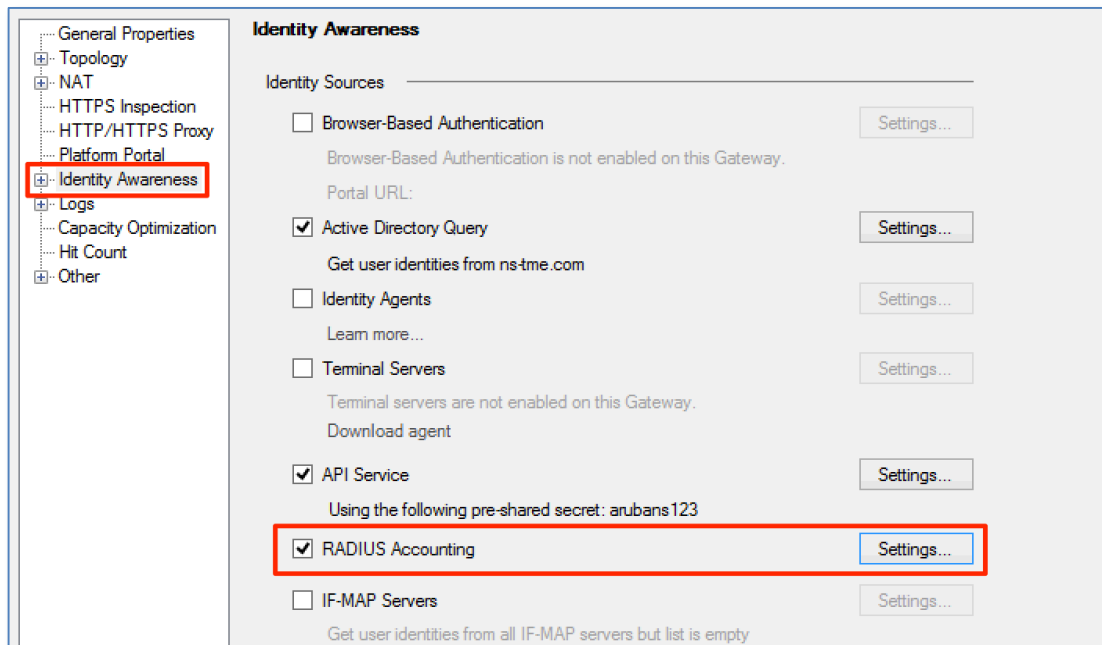


Figure 40 - Enabling and RADIUS accounting on Check Point

Note: You must configure the CPPM server as a network object (it just needs a name and a IP address) now or prior to this step. Below is an example of creating a Network Host object. This can then be referenced in the RADIUS accounting section that follows.

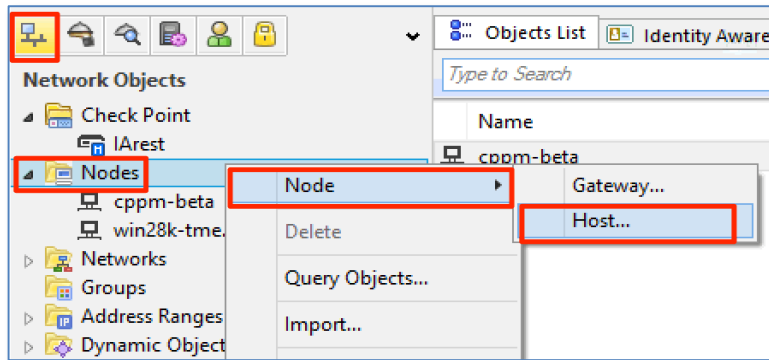


Figure 41 - Defining a HOST Object in Check Point

Below is the RADIUS accounting configuration. There are two very important sections highlighted. The first is defining/adding the CPPM host (RADIUS accounting source). You may have already added this previously as a network object or you can add it now via the **green +** below. **Note:** Don't forget to set the PSK for the CPPM node.

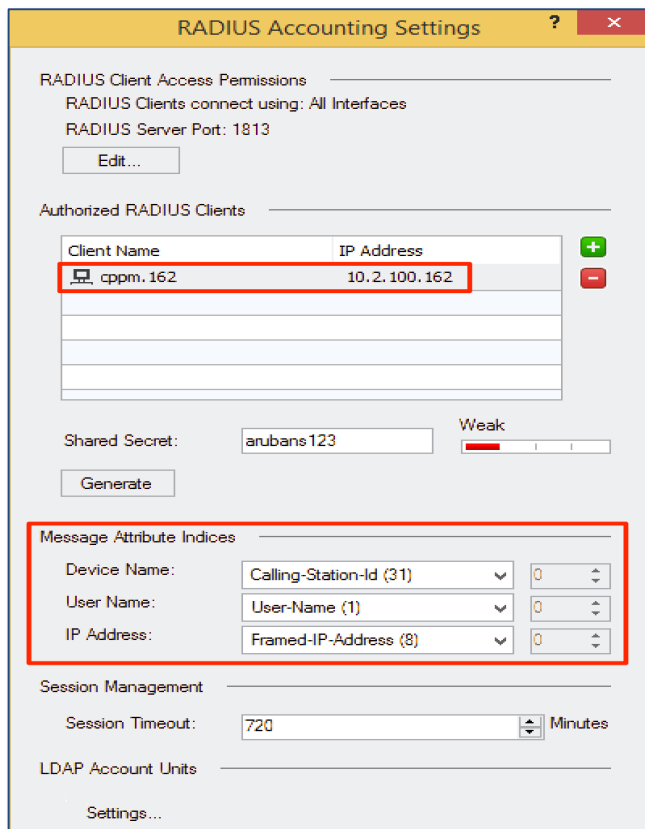


Figure 42 - Configuring RADIUS accounting on the Check Point firewall

The second section is to define the accounting attributes that Check Point will parse from the RADIUS accounting data CPPM sends. It's possible that CPPM will forward a lot of additional data that the target system neither wants nor can process, VSA attributes for example. This data is typically just ignored. In a later release, we intend to allow CPPM to remove the accounting data that is not required by the target system before we send it.

Above, we have configured accounting attributes 31, 1 and 8. These are the three attributes CheckPoint will parse and map to Machine-Name, Username and Endpoint SRC IP address. The attributes you require may differ.

Note: the default attributes configured out-of-the-box are attributes 0, 31 and 8.

Once this is configured and the Policy installed, you should be able have the Check Point firewall receive and parse the mapped attributes from the RADIUS accounting data.

Below is an example from the SmartView Tracker showing user 'danny' logging-in from a source-ip-address of 10.2.100.165 and the user identity source was radius-accounting. You can also see that danny was associated to a role 'match-user-danny' in the Associated Roles.

The screenshot displays the SmartView Tracker interface. On the left, a tree view shows 'Identity Awareness Blade' expanded to 'Login Activity'. The main window shows a log entry for 'danny (danny)' from IP 10.2.100.165. A 'Record Details' dialog box is open, showing the following information:

- Log Info:** Product: Identity Awareness, Date: 16Jan2015, Time: 15:50:17, Number: 19051, Type: Log, Origin: IArrest
- Traffic:** Source: 10.2.100.165, Service: danny (danny)
- Log In:** Action: Log In, Authentication Status: Successful Login, Authentication Method: User Identity Propagation, Description: ---
- User Information:** Source User Group: All Users(+jad_user_danny), Source Machine Group: ---, Associated Roles: match-user-danny
- More Information:** Product Family: Network

The background log table shows the following entry for 'danny (danny)':

No.	Date	Time	Source	User	Host	Group	Auth. Meth.	Auth. Meth.	Auth. Meth.
19...	16Jan2015	15:50:17	10.2.100.165	danny (danny)	hyperv_server2	All Users(-)jad_user_danny	Radius	User Identity Pr...	Radius

Figure 43 - Showing user logging-in of SmartView Tracker

The above solution provides for mapping THREE RADIUS accounting attributes. There is a method discussed below which allows for a fourth Accounting attribute to be exposed. This is extremely useful and allows for ClearPass to attach what can be thought of as a 'user-role' to the user name in the Accounting data. CheckPoint can then use this 'user-role' to enforce policy against different 'classes' of users based upon the 'user-role'.

Configure CheckPoint RADIUS Accounting to expose a fourth attribute



CheckPoint can make available a special hotfix patch that extends the number of attributes parsed to four, today we believe this to be available for s/w version 77.10 and 77.20.

CPPM will add and set an IETF standard (e.g. attribute-77 Connect-Info) with a label e.g. cppm-guests in the RADIUS accounting data. CheckPoint could then parse this attribute and associate the username to a Guest role for example. The Check Point firewall would then be able to apply policy restrictions against this user, but using the label as the differentiator.

The first step is to obtain the HotFix from CheckPoint. Its name is 'RADIUS Accounting Groups', and was released in July 2014. RADIUS Accounting is an existing feature in the Identity Awareness blade. It enables receiving identity information from external authentication servers, using the RADIUS Accounting protocol.

Typically RADIUS Accounting user group information is retrieved from identity servers (AD/LDAP servers or internal databases), but by utilizing this Hotfix we can bypass the typical logic processing of the CheckPoint firewall and 'force' the firewall to trust the group/role information ClearPass sends.

CheckPoint Installation of Hotfix Download from the CheckPoint support site the hotfix.....

fw1_wrapper_HOTFIX_GYPSY_RADIUS_286.tgz

Install this Hotfix on **each** R77.xx firewall. Then separately on each of the Firewall with Identity Awareness enabled you must manually install the fix. Start by SSH'ing to the CLI, ensure you Log into Expert mode and then upload the Hotfix to a temporary directory.

The file that has been uploaded is a Gzipped Tar file. Before it can be installed it needs to be unzipped/unpacked. Use the following command, pay particular attention to the flags...

```
tar -xzvf fw1_wrapper_HOTFIX_GYPSY_RADIUS_286.tgz
```

Now to installed the contents from the file...

```
./fw1_wrapper_HOTFIX_GYPSY_RADIUS_286_990286007_1
```



Plan this installation accordingly as once this installation has completed you must reboot the system to complete the installation.

Configuration of CheckPoint Hotfix

Following the installation of this Hotfix, make sure that the RADIUS Accounting feature is enabled on the Security Gateways with Identity Awareness. This has been covered above.

Enabling the RADIUS Accounting Group Hotfix

We need to edit the following file `$FWDIR/conf/pdp_overriding_attrs.C`

If this pathname does not exist, create it. If the file is empty, add a left parenthesis at the top and add a right parenthesis at the bottom. The file should look like the below. Note as can be seen we have used a standard IETF RADIUS attribute [77] that will be the key-pair-value holding the user role.

Add to a new line inside the parenthesis `:user_groups_index (<index>)`

```
[Expert@gw-fcac73:0]# cat conf/pdp_overriding_attrs.C
(
    :user_groups_index (77)
)
[Expert@gw-fcac73:0]#
```

Where **<index>** is the Accounting attribute number in the RADIUS Accounting Message packet. Note the parenthesis around the attribute value...!!

Monitoring this function on the Firewall

From the CLI you can monitor the users with the following command `pdp m user <user>` below is an example of the output from this command.

```
[Expert@gw-fcac73:0]# pdp m user danny

Session: 4eb46504
Session UUID: {21C2E80C-DD9A-A814-C5AE-8A06FFCFB188}
Ip: 10.2.100.167
Users:
  danny {63f9a734}
    Groups: geek-group
    Roles: geek-group_access_role
    Client Type: Radius Accounting
    Authentication Method: Trust
    Connect Time: Fri Mar 6 15:43:17 2015
    Next Reauthentication: Sat Mar 7 03:43:47 2015
    Next Connectivity Check: -

Packet Tagging Status: Not Active
Published Gateways: Local
*****
```

Configuration of ClearPass to add Accounting Attributes [e.g. user role]

Within ClearPass Policy Manager we now need to amend the Accounting Proxy to add an attribute [connect-info attribute 77] we will populate with a Role/Group value. This is the attribute we defined within the configuration of the CheckPoint firewall hotfix previously.

Note: We are not locked to this attribute, as long as both ends are using a common value.

Navigate to **Configuration -> Service [edit your service]** and click on the Proxy Tab. Within here you need to add the RADIUS Accounting attribute. From the 'Type' select Radius: IETF, from the Name we've chosen 'Connect-Info' and then as shown below set as required, we've set ours to a static value 'geek-group' for the purpose of this test.

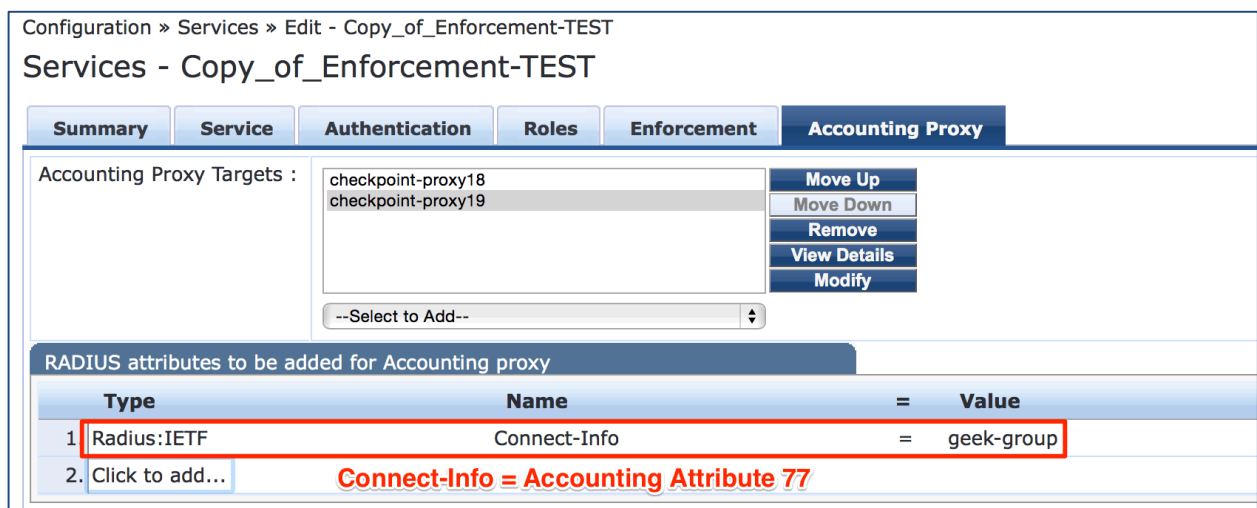


Figure 44 - Adding Accounting attribute to ClearPass Proxy

Note: The value field below can be a substitutional attribute, an example could be to choose from the Authorization AD 'memberOf'. Now that ClearPass is adding this attribute, policy can be defined on CheckPoint to make policy enforcement to different user groups.

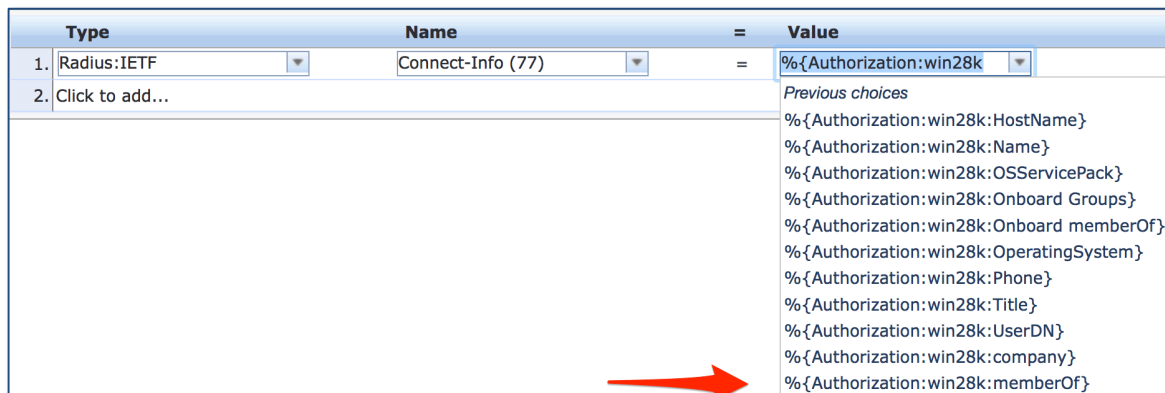


Figure 45 - using authz attributes to pass 'role/group' to Checkpoint

Configuration of ClearPass to pass a dynamic role attribute to CheckPoint



Using a fixed role or the role available from AD might not be suitable. The above method is a little bit inflexible and maybe not suitable or representative of how you want to identify a users-role. In the below example we are using the AD **memberOf** attribute to assign a Tips ROLE to the user, using the *CONTAINS* calculation.... In our example we look for PLM or TME and assign a similar TIPS role.

Conditions		Role
1.	(Authorization:win28k:memberOf CONTAINS PLM)	PLM
2.	(Authorization:win28k:memberOf CONTAINS TME)	TME

Figure 46 - Calculating the TIPS role from AD memberOf

Then within the Accounting Proxy we use the value **{Tips:Role}** to pass the assigned role to the CheckPoint enforcement point.

Note: You have to manually complete and type the text **{Tips:Role}**, as we do not auto-complete this parameter.

Configuration of CheckPoint to parse Accounting Attributes [e.g. user role]

To have CheckPoint use the 'role/group' data we append to the RADIUS Accounting data we need to configure some corresponding data within the Checkpoint firewall. In the testing we show below we will use the following three groups, PLM, TME and geek-group.

PLM and TME are groups within our AD environment whilst geek-group is a static group created for the purpose of this test. First, create your User-Groups, this is the 'label' that matches the incoming attribute from the RADIUS Accounting data. Then you need to create your Access Roles.

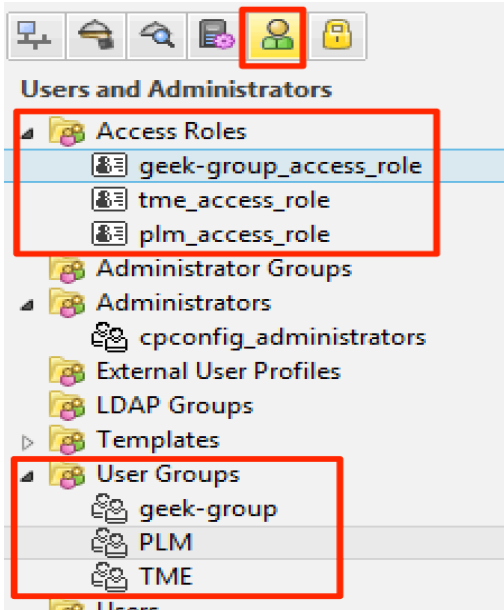


Figure 47 - Creating User Groups & Access Roles

Once the above Roles/Groups have been created you need to align the Groups to the Access Roles. This can also be done at creation time. Edit the Roles as necessary.

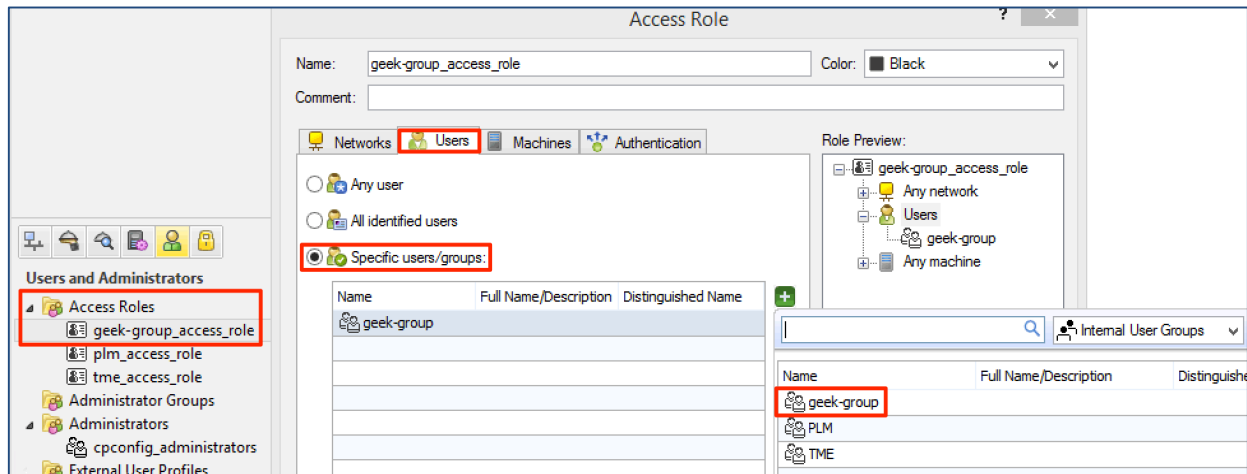


Figure 48 - Aligning Groups to Roles

Below we can see from the Logs in CPPM, the users authenticated in Access-Tracker. We are taking Calling-Station-ID and User-Name from here. The Framed IP Address comes from the Accounting data.

Request Details (Input Tab)

End-Host Identifier:	3010E4A1DB39	(SmartDevice / Apple / Apple iPad)
Access Device IP/Port:	10.2.100.20:0	(10.2.100.20 / Aruba)

RADIUS Request

Radius:Aruba:Aruba-AP-Group	MobileIronDemo
Radius:Aruba:Aruba-Device-Type	iPad
Radius:Aruba:Aruba-Essid-Name	TME-PAN-Demo
Radius:Aruba:Aruba-Location-Id	djump-RAP-3-HOME
Radius:IETF:Called-Station-Id	000B866E0450
Radius:IETF:Calling-Station-Id	3010E4A1DB39
Radius:IETF:Framed-MTU	1100
Radius:IETF:NAS-Identifier	10.2.100.20
Radius:IETF:NAS-IP-Address	10.2.100.20
Radius:IETF:NAS-Port	0
Radius:IETF:NAS-Port-Type	19
Radius:IETF:Service-Type	1
Radius:IETF:User-Name	danny

Request Details (Accounting Tab)

Account Session ID:	danny3010E4A1DB39-54FA3B94
Start Timestamp:	Mar 06, 2015 15:43:16 PST
End Timestamp:	Still Active
Status:	Active
Termination Cause:	-
Service Type:	-
Number of Authentication Sessions:	1

Network Details

NAS IP Address:	10.2.100.20:0
NAS Port Type:	Wireless-802.11
Calling Station ID:	3010E4A1DB39
Called Station ID:	000B866E0450
Framed IP Address:	10.2.100.167
Account Auth:	-

Figure 49 - Grabbing RADIUS attributes from CPPM

Below is the data received within the CheckPoint Smartview Tracker.

Identity Awareness Log In

Severity: [Bar Chart]

Log Info	Log In
Product: Identity Awareness	Action: Log In
Date: 6Mar2015	Authentication Status: Successful Login
Time: 15:43:17	Authentication Method: User Identity Propagation
Number: 4845	Description: ---
Type: Log	Identity Source: Radius Accounting
Origin: gw-fcac73	

Traffic	User Information
Source: 10.2.100.167	Source User Group: geek-group
danny	Source Machine Group: ---
Service: ---	Associated Roles: geek-group_access_role
Protocol: ---	
Interface: ---	
Source Port: ---	

More

Product Family: Network
Information: ---

Above can be seen that the user **danny**, has been assigned to the **User-Group 'geek-group'** which has further been assigned an **Access-Role of 'geek-group_access_role'**.

This completes the configuration examples for integration ClearPass Policy Manger with CheckPoint firewall.