Tenable.sc

ClearPass

a Hewlett Packard Enterprise company
Change Log

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modified By</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>2018-01</td>
<td>Nov 2018</td>
<td>Arpit Bhatt</td>
<td>Initial Release</td>
</tr>
<tr>
<td>2019-01</td>
<td>July 2019</td>
<td>Arpit Bhatt</td>
<td>Minor Corrections</td>
</tr>
</tbody>
</table>

Copyright

©Copyright 2018 Hewlett Packard Enterprise Development LP.

Open Source Code

This product includes code licensed under the GNU General Public License, the GNU Lesser General Public License, and/or certain other open source licenses. A complete machine-readable copy of the source code corresponding to such code is available upon request. This offer is valid to anyone in receipt of this information and shall expire three years following the date of the final distribution of this product version by Hewlett-Packard Company. To obtain such source code, send a check or money order in the amount of US $10.00 to:

Hewlett-Packard Company  
Attn: General Counsel  
3000 Hanover Street  
Palo Alto, CA 94304  
USA

Please specify the product and version for which you are requesting source code. You may also request a copy of this source code free of charge at HPE-Aruba-gplquery@hpe.com.
Contents
Introduction and Overview.................................................................................................................. 5
Software Requirements....................................................................................................................... 6
Installation and Deployment Guide .................................................................................................. 6
Access to the Extension Store ............................................................................................................ 6
Pictorial View of the Integration ....................................................................................................... 7
New Extension Support in ClearPass 6.7+ ....................................................................................... 8
Extensions and IP address configuration support ......................................................................... 8
Extensions and web proxy support ................................................................................................. 8
Configuration Steps ......................................................................................................................... 10
Step I: Tenable.sc setup .................................................................................................................. 10
Step II: Installation and configuration of Tenable Extension on ClearPass .................................. 16
Step III: Adding and Endpoint Context Server, Context Server Action and an Enforcement Profile on ClearPass ................................................................. 21
Creating Enforcement Policies on ClearPass Policy Manager ......................................................... 25
Periodic Poll ................................................................................................................................ 25
Trigger a Scan ............................................................................................................................... 29
Authorization Source ..................................................................................................................... 30
Appendix A – Additional Diagnostics and Support ......................................................................... 32
The Extensions Service ................................................................................................................. 32
Extension logs and debugging ....................................................................................................... 32
Accessing extension logs within ClearPass ‘Collect Logs’ .............................................................. 33
Appendix B – Lab Examples ........................................................................................................... 34
Periodic Poll ................................................................................................................................ 34
Trigger a Scan ............................................................................................................................... 35
Appendix C – XMLs ......................................................................................................................... 39
Figures
Figure 1: Entering HP Passport credentials ................................................................. 6
Figure 2: Pictorial view of ClearPass Policy Manager integration with Tenable SecurityCenter ................................................................. 7
Figure 3: Extension framework GUI ............................................................................ 8
Figure 4: Defining the base IP SUBNET and LOCALHOST for the Extensions framework ......................................................................... 9
Figure 5: Adding a Scan Zone ....................................................................................... 11
Figure 6: Create a Security Manager user ................................................................. 11
Figure 7: Add a Scan Policy I ..................................................................................... 12
Figure 8: Add a Scan Policy II .................................................................................... 12
Figure 9: Add a Scan Policy III .................................................................................. 12
Figure 10: Add a Repository I ..................................................................................... 13
Figure 11: Add a Repository II ................................................................................... 13
Figure 12: Add a Repository III .................................................................................. 13
Figure 13: Add an Active Scan I ................................................................................. 14
Figure 14: Add an Active Scan II ............................................................................... 14
Figure 15: Add an Active Scan III .............................................................................. 15
Figure 16: Add an Active Scan IV .............................................................................. 15
Figure 17: Extensions framework GUI ....................................................................... 16
Figure 18: GUI Extension search .............................................................................. 16
Figure 19: GUI Extension install ............................................................................... 17
Figure 20: GUI Extension configuration at install time .......................................... 17
Figure 21: Extension configuration options ............................................................. 18
Figure 22: Creating an API Admin user ................................................................. 19
Figure 23: GUI review and setting the Extension configuration ................................ 20
Figure 24: Log validation ......................................................................................... 20
Figure 25: Adding an Endpoint Context Server .................................................... 21
Figure 26: Adding a Context Server Action I ......................................................... 22
Figure 27: Adding a Context Server Action II ....................................................... 22
Figure 28: Adding a Context Server Action III ...................................................... 23
Figure 29: Add an Enforcement Profile I ................................................................. 23
Figure 30: Add an Enforcement Profile II ............................................................... 24
Figure 31: Endpoint Repository ............................................................................... 26
Figure 32: Fetched attributes .................................................................................. 26
Figure 33: Sample Enforcement Policy 1 ............................................................... 27
Figure 34: Adding Dictionary Attributes ............................................................... 27
Figure 35: Sample Enforcement Policy 2 ............................................................... 28
Figure 36: Adding Dictionary Attributes ............................................................... 28
Figure 37: Sample Enforcement Policy 3 ............................................................... 29
Figure 38: Sample Enforcement Policy 4 ............................................................... 30
Figure 39: Context Server Action to GET endpoint details .................................... 31
Figure 40: Checking on the extensions service and how to start/stop the service ......................................................................................... 32
Figure 41: Using the GUI to change the DEBUG level ........................................... 32
Figure 42: Extension logs location in 'Collect Logs' diagnostic GZ file ................ 33
Figure 43: Access Tracker result ............................................................................ 36
Figure 44: Triggered scan in Tenable.sc ............................................................... 37
Figure 45: Endpoint attributes ingested from the scan ....................................... 38
Figure 46: Access Tracker result after scan ......................................................... 38
Figure 47: Import Enforcement Profiles ............................................................... 39
Introduction and Overview

Tenable’s SecurityCenter, recently changed to Tenable.sc is a platform that provides a comprehensive and integrated view of enterprise security posture to reduce business risk and ensure compliance. SecurityCenter’s vulnerability analytics engine evaluates vulnerability data gathered across multiple Nessus scanners distributed across your enterprise, illustrates vulnerability trends over time and assesses risk to prioritize the actions needed. The solution helps identify weaknesses by scanning assets for known vulnerabilities, misconfigurations and malware. The scan results are leveraged by ClearPass Policy Manager to take actions based on the policies defined.

This integration guide covers the deployment and configuration of a ClearPass Extension to interface with Tenable.sc which is an On-Prem vulnerability analytics solution. The Extension integrates with SecurityCenter system allowing API access to threat and vulnerability attributes obtained from scan results. This allows the Extension to provide the following integration capabilities.

1. Periodic Poll: enable periodic polling of scanned results for endpoints in Tenable.sc with a valid mac address. This allows Policy Manager to access a number of endpoint attributes which can be leveraged for creating policies. For example
   a. Check if the endpoint is known to Nessus scanner
   b. Check if the Last Scan time was less than 7 days
   c. Check if the endpoint has any Critical vulnerabilities
2. Trigger a Scan: allows the Extension to trigger a configured scan on SecurityCenter for non-compliant endpoints. The scan results are sent to ClearPass to update the attributes in the endpoint database which can be leveraged for defining policies.
3. Authorization source (optional): trigger the Extension to get the latest posture results or attributes for the authenticated endpoint.
Software Requirements

The minimum software version required for ClearPass is 6.7.2. At the time of writing, ClearPass 6.7.7 is the latest available and recommended release. Any subsequent ClearPass software release will support this integration. ClearPass runs on either hardware appliances with pre-installed software, or as a Virtual Machine under the following hypervisors. Hypervisors that run on a client computer such as VMware Player are not supported.

- VMware ESXi 5.5, 6.0, 6.5 or higher
- Microsoft Hyper-V Server 2012 or 2016 R2
- Hyper-V on Microsoft Windows Server 2012 or 2016 R2
- KVM on CentOS 7.5

The Tenable.sc version used to verify interoperability for this guide was 5.6.1

Installation and Deployment Guide

This document assumes your ClearPass environment is already configured and operational. If you require assistance with basic deployment, refer to the following deployment guide:


Access to the Extension Store

Access to the Extension Store to download extensions is simplified in ClearPass 6.7. The ability to download extensions from the store and to validate support entitlement for access to the Software Updates Portal (e.g. Posture & Profile Data Updates, Software Updates, & Skins) now uses the HPE Passport account credentials that are associated with the customers’ ClearPass licenses. This is configured where previously the subscription-id was defined, under Administration -> Agents and Software Updates -> Software Updates as shown below. Ensure you enter your HPE Passport credentials to enable Extension download capabilities.

Figure 1: Entering HP Passport credentials
Pictorial View of the Integration

The diagram below shows a pictorial overview of the components and how they interact with each other.

**Figure 2: Pictorial view of ClearPass Policy Manager integration with Tenable SecurityCenter**
New Extension Support in ClearPass 6.7+

With the release of 6.7, several new features have been added to enhance the functionality of the extension framework. Previously, all extension installation and operation tasks required use of the API Explorer to interoperate with the Extension and the underlying framework. Now this functionality has been exposed with a new GUI. The GUI is accessed from within the Guest UI and is shown below, Administration -> Extensions.

Extensions and IP address configuration support

The other major additions in the 6.7 release are the ability to define the extension framework base IP network and statically define the IP address of the individual extensions. The latter being useful when deploying extensions in a cluster and the requirement for a fixed IP address for the same extension across a cluster regardless of which ClearPass node or nodes it is installed on.

Extensions and web proxy support

Prior to 6.7 support for web proxy was limited to the installation of the extensions. Starting in ClearPass 6.7, extensions now support communications with 3rd parties via a web proxy. This adds incremental web proxy functionality. If a web proxy is defined in ClearPass Policy Manager, then an extension will use that configuration.

The Policy Manager web proxy configuration is ONLY read by the extension at installation time. If the web proxy configuration is changed in Policy Manager, then the extension must be re-installed so the new settings are re-read and bonded to the extension.
Configuring the base Extension IP subnet, this is defined within Policy Manager as shown below under Administration -> Server Manager -> Server Configuration [chose your node] Service Parameters [ClearPass system service]. The default is 172.17.0.1/16, this address is the non-routed address of the ClearPass node itself. The IP addresses range for the extensions are based upon the network prefix used.

The subnet defined here for the extension framework must be 172.17.0.1/16.

**Figure 4: Defining the base IP SUBNET and LOCALHOST for the Extensions framework**

Changing the extension base IP address will require the extension service to be restarted.

Changing the “Extensions Network Address” range is necessary if either the MGMT or DATA interface are also using an address in the extension default range of 172.17.x.x/16. Set the new network address range as needed and restart the extension service for this to take effect.
Configuration Steps

There are primarily 3 steps involved in getting this integration configured.

- Tenable.sc setup
- Installation and configuration of Tenable Extension on ClearPass
- Adding an Endpoint Context Server, Context Server Action and an Enforcement Profile on ClearPass

Before proceeding with the configuration, it is important to understand the customer use case that we are trying to solve.

Please note that you only need to configure Step II, if the goal of this integration is to fetch security context for the endpoints using periodic polls assuming the customer has a Tenable.sc scan configured and running in the infrastructure. If a scan needs to be configured, Step I can be used as a reference and a “Scheduled Scan” can be added.

Tenable.sc collects MAC address for every endpoint scanned on the network. This mac address is ingested by ClearPass which is then used to add or update an endpoint in the ClearPass Endpoint repository. It is very important to receive a correct MAC address from SecurityCenter. Please ensure the scan configured provides the correct MAC address for the endpoints, if not, please contact Tenable support.

If the goal is to leverage the SecurityCenter APIs to trigger a scan for the endpoints in case of a non-compliance of a policy follow all the three steps below.

Step I: Tenable.sc setup

This section covers the steps for adding a basic scan using SecurityCenter to showcase the integration capabilities. This scan gathers and collects information for endpoints in the configured subnet.

This guide assumes you have a Tenable.sc installed with a minimum of one Nessus Vulnerability scanner. Configuration and installation of these is beyond the scope of this guide. The scan configured is for demo purposes only. Please consult with your Tenable engineers for advanced scan designs and configurations.

1. Login to SecurityCenter with an administrator account. Add a Tenable Nessus Scanner. In the SecurityCenter framework, the Nessus Scanner acts as a server, while SecurityCenter serves as a client that schedules and initiates scans, retrieve results, reports results and perform a wide variety of other important functions.

   This step is not required if the scanner has already been configured and added to Tenable.sc. The scanner is used by SecurityCenter to trigger a scan, the information for the scan is sent to the Nessus scanner on port 8834 using valid credentials on the Nessus scanner.

   Go to Resources > Nessus Scanners. Fill in the necessary details to add the scanner.

2. The next step is to add a Scan Zone. This is the network ranges you would want to trigger an active scan for. It would associate a range of an IP to one or more scanners in your deployment.
Go to Resources > Scan Zones and click on Add. Give a Name and IP Ranges for this Scan Zone and select the Scanners responsible to trigger a scan for this zone as shown below.

Figure 5: Adding a Scan Zone

3. Create a User with the role Security Manager. This user will be used by ClearPass Extension to communicate with SecurityCenter APIs.

The User can be added by navigating to Users > Users. Click on Add to create as shown below.

Figure 6: Create a Security Manager user
4. Create a Policy to be used for the scan. This can be created under Scanning > Policies. Click on Add.

**Figure 7: Add a Scan Policy I**

![Image of Tenable interface showing policies section]

You can either select one of the templates or create an advanced scan in the policy. For this example, we have configured a Basic Network Scan.

**Figure 8: Add a Scan Policy II**

![Image of Tenable interface showing policy template options]

Add a name for the policy. Leave everything else as default and click on Submit to save the Policy.

**Figure 9: Add a Scan Policy III**

![Image of Tenable interface showing policy creation screen]
5. The next step is to create a Repository in Tenable.sc. Repositories are databases within SecurityCenter that contain vulnerability data.

Go to Repositories and click on Add.

Figure 10: Add a Repository I

![SecurityCenter interface with Repositories menu open](image)

Select Local > IPv4 Repository

Figure 11: Add a Repository II

![Add Repository interface](image)

Specify the Name and applicable IP Ranges and click on Submit to create the Repository.

Figure 12: Add a Repository III

![Add Repository form](image)
6. The final step is to configure a scan in the SecurityCenter. The scans can only be created by a user account with the role of a “Security Manager”.

This scan is triggered by ClearPass Policy Manager using the Extension configuration settings. Go to **Scans > Active Scans**. Click on **Add**.

**Figure 13: Add an Active Scan I**

![Add an Active Scan I](image)

Specify a **Name** for the scan and make a note of it. This will be used directly in the ClearPass Extension configuration. Also, select the **Policy** that was created in the previous step here as shown below. Once done, click on **Settings**.

**Figure 14: Add an Active Scan II**

![Add an Active Scan II](image)

As can be seen above, this is an **On Demand** scan. This scan is triggered by a ClearPass extension using the Tenable.sc APIs.

A scan in Tenable.sc can also be scheduled to trigger at specific intervals. ClearPass extension can be configured in a way that it can consume the results of this periodic scan and update the attributes associated with scanned endpoints in its own endpoint database at a specific interval. We will cover this use-cases in the sections to follow.
Under **Settings**, select a **Scan Zone** and the **Import Repository** created before. Click on **Targets**.

**Figure 15: Add an Active Scan III**

Under **Targets**, select **Target Type** as “IP/DNS Name” and specify an IP address within the Scan zone. As mentioned before, ClearPass will trigger this scan using the SecurityCenter APIs and send IP as an attribute. The IP sent will replace the one specified here when the scan gets triggered. Hence the IP specified here is a placeholder and is not the one used when the scan is triggered from ClearPass.

**Figure 16: Add an Active Scan IV**

Other settings need not be changed for a basic scan. Click on **Submit** to save the scan.
Step II: Installation and configuration of Tenable Extension on ClearPass

Starting in ClearPass 6.7, a Graphical User Interface (GUI) was introduced to make the process of interacting with the extension framework easier. To access the extension GUI, from the Guest System, under Administration find the Extension User Interface as shown below.

Figure 17: Extensions framework GUI

From here, click on ‘Install Extension’, and the search box below appears. Enter the keyword “Tenable” and click on Search.

Figure 18: GUI Extension search

Starting 6.7, in a cluster environment an extension can be installed on the subscriber nodes directly.
Click on the extension and then the **Install** option, and if necessary, set the IP address. It will automatically pick an IP address if not assigned. Also, it can be set later if required.

**Figure 19: GUI Extension install**

![GUI Extension install](image)

**Figure 20: GUI Extension configuration at install time**

![GUI Extension configuration at install time](image)
After the Extension has been installed, review the configuration and adjust as needed. Notice the options to Start, Delete, Reinstall or Show Logs and the option to edit and set the Extension configuration.

The default configuration used for Extension is below

```json
{
  "verifySSLCerts": true,
  "logLevel": "INFO",
  "securityCenterHost": "<<SecurityCenter Host>>",
  "securityCenterUser": "<<SecurityCenter User>>",
  "securityCenterPassword": "********",
  "cppmUserName": "<<CPPM Admin User>>",
  "cppmPassword": "********",
  "endpointSyncDelayMinutes": 10080,
  "fullSyncOnStart": false,
  "enableEndpointScan": false,
  "endpointActiveScanName": "cppm-endpoint-scan",
  "endpointScanDelaySeconds": 60
}
```

Each of the attributes are explained in the table below in detail.

**Figure 21: Extension configuration options**

<table>
<thead>
<tr>
<th>Configuration attribute</th>
<th>Description</th>
<th>Example/Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>verifySSLCerts</td>
<td>Should the extension validate SSL certificates.</td>
<td>true/false</td>
</tr>
<tr>
<td>logLevel</td>
<td>Logging level for troubleshooting</td>
<td>&quot;DEBUG&quot;, &quot;INFO&quot;, &quot;WARN&quot;, &quot;ERROR&quot;</td>
</tr>
<tr>
<td>securityCenterHost</td>
<td>The host name or IP address of the SecurityCenter host.</td>
<td>Example: 192.168.1.10 or scanner.arubasecurity.net</td>
</tr>
<tr>
<td>securityCenterUser</td>
<td>The user name of a Security Manager account in SecurityCenter.</td>
<td>Username for account created in Figure 6</td>
</tr>
<tr>
<td>securityCenterPassword</td>
<td>The password for the user entered in the securityCenterUser setting.</td>
<td>Password for account created in Figure 6</td>
</tr>
<tr>
<td>cppmUserName</td>
<td>The user name of an Admin user in ClearPass. This is used for device profiling.</td>
<td>API administrator username</td>
</tr>
<tr>
<td>cppmPassword</td>
<td>The password for the user entered in cppmUserName</td>
<td>API administrator password</td>
</tr>
<tr>
<td>enableEndpointCache</td>
<td>If set to true, every time an IP address is looked up the information will also be added to the endpoint cache, assuming the IP has an associated MAC Address.</td>
<td>true/false</td>
</tr>
<tr>
<td>enableEndpointFullSync</td>
<td>Enable a full sync process that attempts to pull all IP addresses and MAC Addresses into the ClearPass endpoint database.</td>
<td>true/false</td>
</tr>
<tr>
<td>endpointSyncDelayMinutes</td>
<td>The delay in minutes between attempted full sync runs. This option is only used when enableEndpointFullSync is true.</td>
<td>Example: 10080</td>
</tr>
<tr>
<td>fullSyncOnStart</td>
<td>If set to true (and enableEndpointFullSync is true) when the extension starts, a full sync will be run.</td>
<td>true/false</td>
</tr>
</tbody>
</table>
The `cppmUserName` and `cppmPassword` should be an API Administrator account. The device profiling attributes obtained from Tenable need to be written into the endpoint repository leveraging the REST APIs.

The ClearPass API Administrator account can be created under Administration > Users and Privileges > Admin Users. Click on Add. A user with the following Privilege Level needs to be created.

**Figure 22: Creating an API Admin user**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>enableEndpointScan</td>
<td>If set to true, submitting an IP address to the extension will queue it for scanning by the Active Scanner identified by <code>endpointActiveScanName</code>. Scanned IP addresses will have their data pushed back into the ClearPass Endpoint database when the scan completed.</td>
<td>true/false</td>
</tr>
<tr>
<td>endpointActiveScanName</td>
<td>The name of a valid Active Scan that has been created in SecurityCenter. This can be any scan that honors a list of IP Addresses for scanning. You can set any value in the list of IP Addresses when creating the scan and the extension will overwrite the IP List automatically before triggering the scan.</td>
<td>Name of the scan created in Figure 14</td>
</tr>
<tr>
<td>endpointScanDelaySeconds</td>
<td>The delay in seconds that the system should wait to check if there are any IP Addresses to be scanned. Only one scan can be running at a time and new endpoints will be queue until the scanner has finished.</td>
<td>Example: 60</td>
</tr>
</tbody>
</table>

Always use an account with the API Administrator privileges only. Do not use a Super Administrator account.

A copy of the Tenable Extension with the desired configuration is shown below, this will need to be modified for your deployment. Include the `securityCenterHost`, `securityCenterUser`, `securityCenterPassword`, `cppmUserName` and `cppmPassword` that will be specific to your environment.
Change or include any other values based on the use cases covered in the next section. Select **Restart** and click on **Save Changes** to restart the extension.

**Figure 23: GUI review and setting the Extension configuration**

![Configuration screenshot](image)

After the configuration and the restart of the extension, click on **Show Logs**.

**Figure 24: Log validation**

![Log validation screenshot](image)

The above log states that the extension is now ready to trigger the scans or consume the information received from Tenable SecurityCenter. Please note the IP address of the Extension. This will be used in the next step to add the Endpoint Context Server within ClearPass.

You can change the logLevel to DEBUG under configuration for detailed logs. These are very useful for troubleshooting and should be included before raising a support case if necessary.
Step III: Adding and Endpoint Context Server, Context Server Action and an Enforcement Profile on ClearPass

This step can be easily configured by importing the XML available on Aruba GitHub. Kindly refer Appendix C for details.

The Endpoint Context Server internally POSTs the Context Server Action and the IP address of the endpoint to the Extension. The Extension would then call the Tenable.sc APIs to trigger a scan sending the IP address of the endpoint along with the scan name.

This can be added under Administration > External Servers > Endpoint Context Servers. Click on Add and use the extension IP address highlighted in Figure 24.

Figure 25: Adding an Endpoint Context Server

Select the Server Type as “Generic HTTP”. The Server Name is the IP address of the Extension. Once you enter that, the Server Base URL gets automatically populated and adds https to begin with.

Ensure you change this to http else the internal POST fails. Please note that this is an internal post within ClearPass from the Policy Manager to the Extension.

The next step is to tell the Extension to trigger a scan which is done using a Context Server Action that can be defined within ClearPass. The Context Server Action allows us to define the HTTP Method, the URL and the JSON contents that will be used for the REST API call to the SecurityCenter.
This can be added under Administration > Dictionaries > Context Server Actions. Click on Add and use the details highlighted below.

**Figure 26: Adding a Context Server Action I**

Select the Server Type as “Generic HTTP” and the Server Name as the Endpoint Context Server we defined in the previous step which in our case is 172.17.0.50. The HTTP Method that will be used to invoke the scan will be using a POST /scan with a basic authentication.

The next tab helps us specify the content type posted which is defined as below.

**Figure 27: Adding a Context Server Action II**
The **Content** tab helps us specify the contents that will be posted to the URL. For the scan to be triggered we need to post the IP address of the endpoint. This IP address is obtained via Radius Accounting. Obviously, if this is not enabled on the NAS ClearPass fails to trigger the scan. Once ClearPass receives the endpoint IP address, the contents are posted to the Extension for it to trigger the scan.

**Figure 28: Adding a Context Server Action III**

![Endpoint Context Server Details](image)

The **Attributes** tab can be left blank. **Save** the Context Server Action.

The Context Server Action is then used in a Post Authentication Enforcement Profile. The next step is to create the Enforcement Profile. Navigate to **Configuration > Enforcement > Profiles**. Click on **Add**.

**Figure 29: Add an Enforcement Profile I**

![Configuration > Enforcement > Profiles > Add Enforcement Profile](image)
Select the **Template** as “Session Notification Enforcement” and specify a **Name**. The Description field is optional.

The next step is to define the **Attributes** as shown below. Specify the **Server IP** as the Extension IP which was used before to define the Endpoint Context Server. The **Login Action** is triggered as soon as ClearPass receives an Accounting start from the NAS device when a client is authenticated on the network. Here we use the Context Server Action defined in the previous step (Figure 26).

**Figure 30: Add an Enforcement Profile II**

Click on **Save** to create the Enforcement Profile. The Enforcement Profile is triggered upon meeting a certain condition which is defined in the Enforcement Policy. This Enforcement Policy is then tied to the 802.1X/RADIUS Service which is used to authenticate endpoints on the network. The configuration of a Service and Enforcement Policies is use case specific and cannot be covered comprehensively in this guide. We will cover Enforcement Policies for a few use cases in the next section.
Creating Enforcement Policies on ClearPass Policy Manager

As discussed in the Introduction section, there are several use cases that can be solved with this integration. The Enforcement Policies and Profiles that are configured depends on the usage of this Extension. We will discuss three approaches here.

Periodic Poll

This is the most commonly used mechanism. Tenable.sc is usually configured to trigger periodic scans. Hence ClearPass should be configured to poll SecurityCenter with the same frequency in order to get the latest value of attributes every time the scan is completed. This allows the Extension to ingest all of the known and scanned IP endpoints into ClearPass endpoint repository.

The Extension configuration parameters that can help control this are `enableEndpointFullSync`, `endpointSyncDelayMinutes` and `fullSyncOnStart`. All of these attributes are defined in the Table (Figure 21) in previous section.

In our example configuration below, the extension will fetch all the endpoints to begin with and then enable full sync at an interval of 7 days (10080 minutes). Hence the extension will perform a full sync again for all the endpoints at the interval (`endpointSyncDelayMinutes`) configured.

```json
{
    "verifySSLCerts": false,
    "logLevel": "DEBUG",
    "securityCenterHost": "10.X.X.X",
    "securityCenterUser": "api-1",
    "securityCenterPassword": "*********",
    "cppmUserName": "apiadministrator",
    "cppmPassword": "*********",
    "enableEndpointCache": false,
    "enableEndpointFullSync": true,
    "endpointSyncDelayMinutes": 10080,
    "fullSyncOnStart": true,
    "enableEndpointScan": true,
    "endpointActiveScanName": "ClearPass-Endpoint-Scan",
    "endpointScanDelaySeconds": 60
}
```

Hence this permits an async process of ClearPass to download and store the contextual security data discovered by the Nessus scanner for an endpoint using the SecurityCenter APIs.
Validation of the endpoint context being updated after the initial sync can be done by navigating to **Configuration > Identity > Endpoints**. Filter using the attribute **Source = Tenable** as shown below.

**Figure 31: Endpoint Repository**

![Endpoint Repository](image)

Shown below are the security contextual parameters obtained from Nessus for an endpoint, these attributes can be evaluated by the Enforcement Policy.

**Figure 32: Fetched attributes**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Tenable</td>
</tr>
<tr>
<td>Tenable IP</td>
<td>10.1.1.1</td>
</tr>
<tr>
<td>Tenable Last Unauth Run</td>
<td>2018-10-26 05:00:31</td>
</tr>
<tr>
<td>Tenable Last Update</td>
<td>2018-12-03 01:40:59</td>
</tr>
<tr>
<td>Tenable MAC Address</td>
<td>1c:98:ec:16:67:34</td>
</tr>
<tr>
<td>Tenable Managed</td>
<td>true</td>
</tr>
<tr>
<td>Tenable Plugin Set</td>
<td>201712131815</td>
</tr>
<tr>
<td>Tenable Policy Name</td>
<td>79cd2ece-b398-5ce1-b253-a9178c30f825-165387/AB port scan</td>
</tr>
<tr>
<td>Tenable Score</td>
<td>0</td>
</tr>
<tr>
<td>Tenable Severity Critical</td>
<td>0</td>
</tr>
<tr>
<td>Tenable Severity High</td>
<td>0</td>
</tr>
<tr>
<td>Tenable Severity Info</td>
<td>13</td>
</tr>
<tr>
<td>Tenable Severity Low</td>
<td>0</td>
</tr>
<tr>
<td>Tenable Severity Medium</td>
<td>0</td>
</tr>
<tr>
<td>Tenable Total</td>
<td>13</td>
</tr>
</tbody>
</table>
Let's look at a few sample Enforcement Policies that can be created with the attributes ingested above.

1. A good practice would be to check if the device is known to the Nessus Scanner at the time of authentication.

   A simple Enforcement Policy condition for such cases would include `Endpoint:Source EQUALS Tenable`

   **Figure 33: Sample Enforcement Policy 1**

   ![Enforcement Policy Example](image)

2. Ensure that devices connecting to the network have been scanned as per policy within the last 24-hours, or perhaps the last 7-days. This is very important for a number of federal use-cases.

The Tenable Last Auth Run, Tenable Last Unauth run and Tenable Last Update attributes created by Policy Manager uses the Data Type as String by default. If these needs to be leveraged in a policy, we have to delete these attributes and define them with the exact same name using Data Type as Date-Time. This is currently a limitation which will be resolved in the later release.

The Tenable Last Unauth Run attribute can be removed and added from Administration > Dictionaries > Dictionary Attributes. This step should be done before the sync of all the endpoints. If synced, delete all the synced endpoints from the endpoint repository using the filter “Source equals Tenable”.

**Figure 34: Adding Dictionary Attributes**

![Dictionary Attributes](image)

The above screenshot shows us the attribute `Last Scan` which have been manually added with the Data Type Date-Time.
A sample enforcement policy to check if an endpoint is not scanned in last 7 days is shown below. The attribute Tenable Last Unauth Run is being used. Time Source within ClearPass is used to compare with the Last Scan time as shown below.

**Figure 35: Sample Enforcement Policy 2**

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Tips:Role: EQUALS [User Authenticated]) AND (Endpoint:Source: EQUALS Tenable) AND (Endpoint:Tenable Last Unauth Run: GREATER_THAN %([Time Source]:Minus One Week DT))</td>
<td>[Allow Access Profile] Quarantine Role</td>
</tr>
</tbody>
</table>

3. Another good check would be to see if an endpoint has any Critical vulnerability reported by Tenable.sc. If yes, we can Quarantine this endpoint.

The Tenable Severity attributes created by Policy Manager uses the Data Type as String by default. If these needs to be leveraged in a policy, we have to delete these attributes and define them with the exact same name using Data Type as Integer. This is currently a limitation which will be resolved in the later release.

The Tenable Severity attributes can be removed and added from Administration > Dictionaries > Dictionary Attributes. Again, this step should be done before the sync of all the endpoints. If synced, delete all the synced endpoints from the endpoint repository using the filter “Source equals Tenable”.

**Figure 36: Adding Dictionary Attributes**

The above screenshot shows the attribute Tenable Severity Critical which has been manually added with the Data Type Integer. The other Tenable Severity attributes are the ones that were automatically created by Policy Manager when the contextual data for an endpoint was received from Tenable.sc.
Let’s look at a sample Enforcement Policy which checks for any critical vulnerabilities on the scanned endpoint and Quarantines it.

**Figure 37: Sample Enforcement Policy 3**

![Image of Enforcement Policy](image_url)

All the 3 sample Enforcement Policies shown in this section sent a Quarantine role upon not satisfying a compliance condition. This is a basic example policy shown here. It can be enhanced by adding multiple actions or Enforcement Profiles like Notifying the Firewall, Opening a ServiceNow ticket etc.

One other action that could be so useful when sending the Quarantine Role is to trigger a scan.

Let’s consider scenario 1 where the endpoint is unknown to Tenable.sc. In such scenarios, it would be very useful to trigger a scan for this endpoint. For scenario 2, where the Tenable Last Unauth Run time was 1 week less than the current time it would be beneficial to trigger a scan and get the latest value of the security context associated with that endpoint. The next section talks about this use case.

**Trigger a Scan**

The next feature available within the integration that can be leveraged is the ability of the ClearPass Extension to trigger a pre-configured scan on Tenable.sc. Remember, in order to get this working one needs to configure Step I, II and III as described in the section “**Configuration Steps**”.

Let’s consider an example scenario where a new endpoint authenticates on the network. This endpoint would be authenticated by ClearPass Policy Manager based on the policies defined on the server. If the policy enforces that only endpoints know to Tenable can get full access on the network, then this device would stay in a quarantine or restricted access role until we get the necessary attributes from Tenable.sc. These attributes are shared after the endpoint is scanned by Nessus.
The integration allows us to trigger an On Demand scan which has been pre-configured in the Security Center. The result of this scan can then be used to ingest the attributes associated with the endpoint.

A simple Enforcement Policy for such a scenario or use case would look like below

**Figure 38: Sample Enforcement Policy 4**

As can be seen above, if the Endpoint is not known to Tenable it gets into Quarantine role. The option to trigger a scan so that the security context pertaining to the endpoint is added. A session timeout of 10 minutes is added so that the endpoint reconnects after 10 minutes post which the policy is re-evaluated, hence a new enforcement profile can be applied.

The policy can be modified to allow endpoint on the network for the first day when it got connected and only trigger a scan. Quarantine the endpoint if no security attributes are available the following day. The Time Source within ClearPass can be leveraged to create such policies.

From the data ingested, we can use the Date-Time attributes in the endpoint to check if the last scan performed was older than X days. If yes, the Extension can trigger the scan to get the latest security parameters.

**Authorization Source**

The extension also allows Tenable.sc as an Authorization source. This use case, although available does not make a lot of sense in case of this integration as the attributes do not change in real time. The endpoint scans are usually scheduled to happen once a week or a day hence it makes more sense to align the Periodic Poll rather than trigger a GET to fetch the value of attributes every time the endpoint connects.

If for any reason, you have to use this, one needs to create a new Context Server Action which performs the GET for the IP of the endpoint.
The Context Server Action can be added under **Administration > Dictionaries > Context Server Actions**. Use the settings as shown in the screen shot below and save the Action. All the other tabs are blank.

**Figure 39: Context Server Action to GET endpoint details**

![Endpoint Context Server Details](image)

This action should then be used in the Enforcement Profile. It is similar to the Tenable Scan action used as a Login Action in Figure 30 which would then be tied to the Enforcement Policy and the Service.
Appendix A – Additional Diagnostics and Support

The Extensions Service

The ClearPass extension is supported by a new system service that was initially added in 6.6. This service should be running. Note that restarting this service will affect all deployed and running extensions.

To check on the state and to restart the service, go to Administration > Server Manager > Server Configuration [select a ClearPass node] > Service Control. From here start/stop the extension service. By default, this service is automatically started.

Figure 40: Checking on the extensions service and how to start/stop the service

Extension logs and debugging

Referencing the configuration previously used, adjust the logLevel to ‘DEBUG’. In the new 6.7, GUI change the configuration and restart the extension as shown below. Logs can then be viewed from the ‘Show Logs’.

Figure 41: Using the GUI to change the DEBUG level
Remember after changing the logging level, the extension will need to be restarted for this change to take effect.

**Accessing extension logs within ClearPass ‘Collect Logs’**

In addition to the logging of messages that be examined in the extension as shown above, it’s possible to configure the extension to log messages so that they can be collected and examined via the Policy Manager ‘Collect Logs’ system function. This is extremely useful for Aruba TAC. The logs are available under **Administration > Server Manager > Server Configuration > Collect Logs**.

If there is a requirement for Aruba TAC to investigate a system issue, one of the items they regularly ask for is the system logs to aid with their diagnostic investigation. The ClearPass extension can write its logs such that they are available and can be collected with all other system diagnostics information when the ‘Collect Logs’ function is run. Remember that by default, the logLevel is set to INFO but TRACE, DEBUG, INFO, WARN, ERROR, FATAL can also be set. Any of the levels will display the information for the selected state and lower. For example, if INFO is selected, it will show messages for INFO, WARN, ERROR, FATAL.

After the logs have been collected and exported from the system, expand the GZ file and locate the extension logs in the following location ‘**PolicyManagerLogs-extension**’ as shown below.

*Figure 42: Extension logs location in ‘Collect Logs’ diagnostic GZ file*
Appendix B – Lab Examples

Periodic Poll

In case of a periodic poll, the Extension will ingest security context for all the endpoints scanned by Nessus scanner via Tenable.sc. Here we have enabled the Extension to perform a full sync on start using the parameter `fullSyncOnStart` set to true. A periodic scan when triggered would work exactly the same way.

Here are the DEBUG logs that we see in the Extension when it is started hence invoking a full sync.

```
[2018-11-18T01:52:02.345] [INFO] tenable - Starting extension web services...
[2018-11-18T01:52:02.354] [INFO] tenable - Server listening on port 80
[2018-11-18T01:52:02.453] [DEBUG] tenable - Getting a new token...
[2018-11-18T01:52:02.720] [DEBUG] tenable - Attempting to get the token timeout...
[2018-11-18T01:52:02.721] [DEBUG] tenable - Existing token is still valid, using it.
[2018-11-18T01:52:02.734] [DEBUG] tenable - Attempting to get the token timeout...
[2018-11-18T01:52:02.734] [DEBUG] tenable - Existing token is still valid, using it.
[2018-11-18T01:52:02.956] [INFO] tenable - Setting the token timeout to 3600.
[2018-11-18T01:52:02.988] [INFO] tenable - Setting the token timeout to 3600.
[2018-11-18T01:52:03.176] [INFO] tenable - Using the Active Scanner with the name "ClearPass-Endpoint-Scan" (10).
[2018-11-18T01:52:03.177] [INFO] tenable - Next queue scan at Sun Nov 18 2018 01:53:03 GMT-0800 (PST).
[2018-11-18T01:52:03.376] [DEBUG] tenable - Processing device with IP 10.X.X.1...
[2018-11-18T01:52:03.381] [DEBUG] tenable - Attempting to update endpoint...
[2018-11-18T01:52:04.308] [DEBUG] tenable - Device 00:07:43:0a:5a:f7 update error. Attempting to add the device.
```
As can be seen in the logs, ClearPass Extension adds all the endpoints that it does not have the information for and updates the ones that exist. A periodic scan would similarly update all the endpoints when the information is received. As mentioned previously, the device MAC address is received from Tenable which is then added or updated in the ClearPass Endpoint Repository. It is very important that we receive a correct MAC address from Tenable to update a valid endpoint in ClearPass.

If the device Profiling attributes are missing on ClearPass then the information received from Tenable Extension is sent to the profiler to update the device attributes like Device Category and Device OS family.

### Trigger a Scan

In this case, the ClearPass Extension triggers a pre-configured scan on Tenable.sc The use case is whenever a new device connects on the network, ClearPass will not have the security context data from Tenable for those endpoints as they would have never been scanned by Nessus. Hence if the policy is configured in a way that the endpoint needs to have an attribute returned by Tenable.sc to get full access it would not be able to satisfy that condition. This calls for an ability to trigger a scan for such endpoints. Here we have enabled the Extension to perform a scan using the parameters `enableEndpointScan` set to true and `endpointActiveScanName` set to a pre-configured scan that needs to be triggered in the SecurityCenter.

Let’s look at the Access Tracker entry for a new device getting authenticated on the network. Since the endpoint does not satisfy the condition of having the attributes from Tenable, we trigger a Tenable Scan using the Enforcement Profile shown below. We expect this scan to be completed in 10 minutes hence setting the session timeout so that the policy can be re-evaluated after the scan. A restricted role is granted until then.

Notice the mac address of the endpoint connected. We would expect the same mac address to be returned by Tenable.sc.
This results in the Endpoint Context Server submitting the action and the IP address to the Extension. Let us look at this in the Extension Debug log as shown below:

```
[2018-11-18T11:50:38.710] [DEBUG] tenable - /scan request received.
[2018-11-18T11:50:38.710] [DEBUG] tenable - Queued the scanning of IP 10.X.X.X.
[2018-11-18T11:51:30.818] [DEBUG] tenable - Existing token is still valid, using it.
[2018-11-18T11:51:31.219] [DEBUG] tenable - Starting scan...
    "scanID": "10",
    "scanResult": {
        "initiatorID": "4",
        "ownerID": "3",
        "scanID": "10",
        "resultsSyncID": -1,
        "jobID": "185258",
        "repositoryID": "1",
        "name": "ClearPass-Endpoint-Scan",
        "description": "",
        "details": "Network-scan-1",
        "status": "Queued",
        "downloadFormat": "v2",
        "dataFormat": "IPv4",
        "resultType": "active",
        "id": "343"
    }
}
```
This triggers the scan configured on SecurityCenter. Login into Tenable.sc with the Security Manager account. Navigate to Scans > Scan Results. A configured scan is triggered as shown below.

Figure 44: Triggered scan in Tenable.sc

Once the scan is completed, the results are sent to the Extension which would write the attributes returned in the Endpoint Repository. The corresponding Debug log is

Hence, we are receiving a correct MAC address for the IP scanned from the SecurityCenter.
Finally, we can also see the attributes in the Endpoint repository.

**Figure 45: Endpoint attributes ingested from the scan**

The endpoint re-authenticates after a configured session timeout of 600 seconds. A new role is then applied based on the attributes received.

**Figure 46: Access Tracker result after scan**

The policy used here was an example policy. This can also be configured to use the Time Source to allow the new devices full access on day 1 but provide them restricted access from day 2 if not scanned and does not contain the security attributes.
Appendix C – XMLs

The Enforcement Profile, Context Server Action and the Endpoint Context Server configured in the section “Step III: Adding and Endpoint Context Server, Context Server Action and an Enforcement Profile on ClearPass” can easily be imported using the XML available on Aruba GitHub

https://github.com/aruba/clearpass-exchange-snippets/tree/master/extensions/tenable-securitycenter

- Download the XML profile > “clearpass-extension_tenable-securitycenter_enf-profile.xml”.
- Open this file in your favorite editor.
- Use the Find and Replace feature of the editor to replace "<Tenable Extension IP>>" with the actual IP address of the Tenable Extension configured. Refer Figure 24.
- Save the file and Import.

The file can be imported from Configuration > Enforcement > Profiles.

Figure 47: Import Enforcement Profiles

The above action will import the following on ClearPass

1. Post Authentication Enforcement Profile: Trigger a Tenable Scan
2. Context Server Action: Tenable Scan
3. Endpoint Context Server