Cylance Protect
ClearPass and Cylance Protect Integration

Change Log

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Modified By</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1 &amp; 0.2</td>
<td>Feb 2017</td>
<td>Danny Jump</td>
<td>Draft Version</td>
</tr>
<tr>
<td>v2018-01</td>
<td>August 2018</td>
<td>Danny Jump</td>
<td>First Published Version</td>
</tr>
<tr>
<td>V2020-01</td>
<td>March 2020</td>
<td>Tony Kord</td>
<td>Revisions for Cylance v.2 Extension</td>
</tr>
</tbody>
</table>

Copyright

© Copyright 2020 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 ................................................................................................................. 6
Software Requirements ......................................................................................................................... 6
ClearPass Installation and Deployment Guide ....................................................................................... 6
ClearPass Extensions .......................................................................................................................... 6
Access to the extension store ............................................................................................................... 7
Pictorial View of the Integration ......................................................................................................... 8
Extension capabilities in ClearPass ..................................................................................................... 9
  Extensions and web proxy support .................................................................................................... 9
  Extensions and IP address configuration support ............................................................................. 9
Installation of Extension on Subscribers ............................................................................................... 10
Cylance Extension installation ........................................................................................................... 11
Cylance Configuration ....................................................................................................................... 15
  Cylance Protect Tenant Configuration ......................................................................................... 15
ClearPass Cylance Protect Extension Configuration ........................................................................... 16
Additional Configuration Notes ......................................................................................................... 19

Configure ClearPass Policy Manager ................................................................................................. 20
  Periodic Sync Mode: Using stored attributes ................................................................................. 20
  HTTP Authorization Source Mode: A Real-Time authorization check ...................................... 21
    Defining a Cylance HTTP Authorization Source ..................................................................... 21
    Using Results from the Cylance HTTP Auth Source ................................................................. 23
Appendix A – List of returned Cylance endpoint attributes ................................................................. 24
Appendix B – Troubleshooting and Support ....................................................................................... 25
  Check API Access Application Control restrictions ...................................................................... 25
  Checking on the Extension Service ................................................................................................. 26
  Extension Logs/Debugging ............................................................................................................... 27
  Accessing the extension logs using ‘Collect Logs’ system function .............................................. 28
Appendix C – Considerations for Installing in a Cluster ................................................................. 32
  Periodic Sync Mode ....................................................................................................................... 32
  HTTP Authorization Source Mode ................................................................................................. 32
Appendix D – Considerations when installing outside of the USA ...................................................... 33
Appendix E – endpointSyncSchedule settings .................................................................................. 34
Appendix F – Using the Cylance Extension information in a ClearPass Service ............................... 35
  Service Authorization ..................................................................................................................... 35
  Role Mapping Policy ...................................................................................................................... 36
  Custom Time Source Attribute ..................................................................................................... 37
  Enforcement Policy ......................................................................................................................... 37
Figures

Figure 1: Entering HP Passport credentials ................................................................. 7
Figure 2: Pictorial view of the integration ................................................................. 8
Figure 3: Extension Framework GUI ........................................................................ 9
Figure 4: Defining the base IP SUBNET and LOCALHOST for the Extensions Framework ................................................................. 10
Figure 5: Extensions Framework GUI ....................................................................... 11
Figure 6: Search for Extension .................................................................................... 11
Figure 7: GUI Extension Search .................................................................................. 12
Figure 8: Install Extension ......................................................................................... 12
Figure 9: GUI Extension Configuration at Install time ............................................. 13
Figure 10: GUI Reviewing and Setting the Extension configuration .................... 14
Figure 11: Cylance Integration Menu ....................................................................... 15
Figure 12: Adding a Cylance Application .................................................................. 16
Figure 13: Saving the Cylance application .................................................................. 16
Figure 14: Cylance Configuration Attributes ............................................................ 17
Figure 15: Creating an Admin user on ClearPass .................................................... 19
Figure 16: Role Mapping Policy using Cylance Endpoint Attributes ..................... 20
Figure 17: Adding HTTP authorization source ....................................................... 21
Figure 18: Defining the authZ Extension IP address ............................................... 22
Figure 19: Building the Cylance Extension query filter ....................................... 22
Figure 20: Adding the required Cylance endpoint fields/attributes ...................... 23
Figure 21: Building a Role Mapping Policy to utilize Cylance HTTP Auth Source attributes ................................................................. 23
Figure 22: Example of Extension authorization failure due to Policy Manager Application Control ................................................................. 25
Figure 23: Locking down access to the Policy Manager API for extensions .......... 25
Figure 24: Services Control ....................................................................................... 26
Figure 25: Using the GUI to change the DEBUG level ........................................... 27
Figure 26: Locating the Extension logs from 'Collect Logs' diagnostic GZ file ................................................................. 28
Figure 27: Enable extension statistics ....................................................................... 29
Figure 28: Show Extension Details ........................................................................... 30
Figure 29: Show Extension Statistics URL .............................................................. 30
Figure 30: Extension Statistics ................................................................................ 31
Figure 31: Cylance Region Codes .......................................................................... 33
Figure 32: Regional Cylance Cloud settings ............................................................. 33
Figure 33: Cron Job Reference ................................................................................. 34
Figure 34: Add Authorization to ClearPass Service ................................................. 35
Figure 35: Authorization Sources ............................................................................. 36
Figure 36: Modify Role Mapping Policy ................................................................. 36
Figure 37: Add Custom Time Source Attributes ................................................... 37
Figure 38: Modify Enforcement Policy ................................................................. 37
**Introduction and Overview**

This TechNote covers how to deploy and configure the ClearPass Extension to interface with Cylance Protects advanced endpoint threat detection. In this TechNote we will cover the complete installation, configuration and integration between the Extension and ClearPass Policy Manager. The Extension effectively becomes an authorization source to service policies.

**Software Requirements**

The minimum software version required for ClearPass is 6.7.2. At the time of writing, ClearPass 6.8.5 is the latest available and recommended release. Any subsequent ClearPass software release will support this integration. The minimum Cylance Protect version supported is dated May 2018.

**ClearPass 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: [http://www.arubanetworks.com/techdocs/ClearPass/Aruba_DeployGd_HTML/Default.htm](http://www.arubanetworks.com/techdocs/ClearPass/Aruba_DeployGd_HTML/Default.htm)

**ClearPass Extensions**

The integration between ClearPass Policy Manager and Cylance Protect is driven through a ClearPass capability known as Extensions, a sub-component of the ClearPass Exchange Integration framework. ClearPass Extensions are micro-services running on top of the base ClearPass platform. These micro-services enable Aruba to deliver new features outside of the main software release cycle and facilitate a faster time to market for specific features and integrations. Configuration and control of ClearPass Extensions is accomplished through the ClearPass Guest GUI, as covered later in this document.
Access to the extension store

Access the Extension Store to download and install ClearPass extensions. The Extension store utilizes the same HPE Passport account credentials used to validate support entitlement in the Software Updates Portal. This is configured under **Administration > Agents and Software Updates > Software Updates** as shown below. Ensure that valid HPE Passport credentials have been entered in these fields to enable Extension download capabilities.

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

The extension can be configured for two different modes of operation.

**Periodic Sync Mode**

In this mode ClearPass polls Cylance periodically and updates the ClearPass Endpoint database with attributes obtained from Cylance Protect. These attributes can be utilized in ClearPass during endpoint Authorization. This mode has a simpler configuration demand, however, the data provided may not be completely up to date at authentication time.

**HTTP Authorization Source Mode:**

In this mode we configure an HTTP Auth source that results in a real time API call to Cylance during endpoint authorization. Information in this case is more up to date, but there is the higher overhead of an API call during endpoint authorization. A view of this process is below.

*Figure 2: Pictorial view of the integration*
Extension capabilities in ClearPass

Extensions are installed via the GUI reachable from within the ClearPass Guest, as shown below. Access it from Guest > Administration > Extensions.

Extensions and web proxy support

Extensions support communications with 3rd parties via a web proxy. This adds incremental proxy functionality. If a proxy is defined in ClearPass Policy Manager, then an extension will use that configuration.

**Note that the Policy Manager web proxy configuration is ONLY read at 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.**

*Figure 3: Extension Framework GUI*

Extensions and IP address configuration support

ClearPass uses a non-externally routed IP address range to communicate with the Extension. The default is 172.17.0.0/16. You may configure a different range, if desired. This is especially useful when deploying extensions across nodes within a cluster where there is the requirement for a fixed consistent IP address for the extension across the cluster.

Changing the “Extensions Network Address” range is only necessary if either the ClearPass MGMT or DATA interface are using an IP address in the extension default range of 172.17.x.x/12, or if ClearPass needs to communicate with some external device in that range.

To Configure the base Extension IP subnet within Policy Manager navigate to Administration > Server Manager > Server Configuration [chose your node] Service Parameters [ClearPass system service].

Note that the subnet defined here for the extension framework must fall within the following subnet range 10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16 as defined by RFC1918.
Figure 4: Defining the base IP SUBNET and LOCALHOST for the Extensions Framework

Note that changing the extension base IP address will require the extension service to be restarted. For best results, set the network address range to a subnet that does not exist in your enterprise, and restart the extension service for this change to take effect.

Never set the DATA or MGMT IP address to use an address that matches the Extension Network

Installation of Extension on Subscribers

If you are installing the extension on a ClearPass cluster, please see Appendix C – Considerations for Installing in a Cluster
Cylance Extension installation

To access the extension GUI, from the ClearPass Guest System, under Administration navigate to the Extension User Interface as shown below.

Figure 5: Extensions Framework GUI

From here, click on ‘Install Extension’, and the search box below appears.

Figure 6: Search for Extension

Enter “Cylance” and click on ‘Search’, see the example below.
**Figure 7: GUI Extension Search**

Click on the extension name and then click "Install."

**Figure 8: Install Extension**
In the “Install Extension” dialog box, set the IP address if necessary, as described earlier. Do not check the box to start the extension at this time. Click the “Install” button.

**Figure 9: GUI Extension Configuration at Install time**

In this example, we've manually entered an IP address for the extension to use.

The extension will download and appear in a “Stopped” state.

Notice the options to Start, Delete, Reinstall, Show Logs, and view Configuration. Click on “Configuration” to view settings.

After the extension has been installed, proceed to configure Cylance and ClearPass.
Figure 10: GUI Reviewing and Setting the Extension configuration

A copy of the default Cylance Extension configuration is shown above, this will need to be modified for your deployment.

Note: Password and sensitive configuration items are obfuscated when presented in both the Extension GUI or in the Explorer configuration.
Cylance Configuration

The ClearPass Cylance Protect extension communicates with your Cylance tenant via REST API calls. By default your Cylance Protect tenant is not enabled to receive and respond to API calls, so you'll need to create a Cylance Integration Custom Application to enable API access. In this process a number of configuration items should be collected as they will be required to configure the ClearPass Cylance Extension.

Cylance Protect Tenant Configuration

Follow the below steps to configure and collect the data from your Cylance tenant.

After login, go to Settings > Integrations on the Cylance nav-bar as shown below.

First make a copy and save the Tenant ID.

Then from under the Integrations options, click on "+ Add Application".

*Figure 11: Cylance Integration Menu*

In the dialog box enter an application name and check Devices=READ, as shown below.
Figure 12: Adding a Cylance Application

Click on Save. Ensure you copy and save the Application ID and Application Secret from the next screen, then Click on OK.

Figure 13: Saving the Cylance application

At this stage you should have three pieces of saved data from creating the Application; these will be required shortly when configuring the ClearPass Extension.

- cylanceTenantId
- cylanceApplicationId
- cylanceApplicationSecret

ClearPass Cylance Protect Extension Configuration

After installing the Extension, the default configuration will need to be updated.

Edit the three configuration value pairs from the previous step with your saved values.
Below is an explanation of all of the configuration value pairs. The **logLevel**, **verifySSLCerts** and **cylanceSubDomain** should be left as their default settings unless advised by ClearPass TAC or your Aruba SE/Partner. Once you configure these parameters, you can start your extension. Any subsequent reconfiguration requires a restart of the extension.

**Figure 14: Cylance Configuration Attributes**

<table>
<thead>
<tr>
<th>Configuration attribute</th>
<th>Description</th>
<th>Example/Values</th>
</tr>
</thead>
<tbody>
<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>verifySSLCerts</td>
<td>Specifies whether SSL certificates be validated when making requests.</td>
<td>true or false</td>
</tr>
<tr>
<td>cylanceSubDomain</td>
<td>This is the sub-domain of the Cylance API URL. The default is protectapi,</td>
<td>See Appendix D.</td>
</tr>
<tr>
<td></td>
<td>if you are in another region outside of North America you may need to update</td>
<td></td>
</tr>
<tr>
<td></td>
<td>this setting.</td>
<td></td>
</tr>
<tr>
<td>cylanceTenantId</td>
<td>The Cylance Tenant ID from the integration application configuration.</td>
<td></td>
</tr>
<tr>
<td>cylanceApplicationId</td>
<td>The Cylance Application ID from the integration application configuration.</td>
<td></td>
</tr>
<tr>
<td>cylanceApplicationSecret</td>
<td>The Cylance Application Secret from the integration application configuration.</td>
<td></td>
</tr>
<tr>
<td>enableEndpointCache</td>
<td>Enable to disable endpoint caching when using the extension device query</td>
<td>true or false</td>
</tr>
<tr>
<td></td>
<td>APIs. When this is enabled the ClearPass endpoint database will be checked</td>
<td></td>
</tr>
<tr>
<td></td>
<td>for the device in the Endpoint Database, if found and there is valid Cylance</td>
<td></td>
</tr>
<tr>
<td></td>
<td>data it will be returned. If the device is out of date (based on the</td>
<td></td>
</tr>
<tr>
<td></td>
<td>endpointCacheTimeSeconds setting), it will be updated.</td>
<td></td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
<td>Value</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>endpointCacheTimeSeconds</td>
<td>The time in seconds that a cached endpoint is valid in ClearPass.</td>
<td>300</td>
</tr>
<tr>
<td>enableEndpointFullSync</td>
<td>Enables or disables the endpoint sync scheduler.</td>
<td>true or false</td>
</tr>
<tr>
<td>endpointSyncSchedule</td>
<td>The CRON schedule used for the sync process.</td>
<td>10 3 * * *</td>
</tr>
<tr>
<td></td>
<td>See Appendix E.</td>
<td></td>
</tr>
<tr>
<td>includeThreatSummary</td>
<td>Specifies whether the sync process should include device threat information in the endpoint attributes.</td>
<td>true or false.</td>
</tr>
<tr>
<td></td>
<td>Note: Setting this attribute to true causes the extension to make extra API calls for each device and may slow the sync process down. To use this feature requires enableEndpointCache to also be set to true.</td>
<td></td>
</tr>
<tr>
<td>endpointSyncOnStart</td>
<td>Specifies whether to run the endpoint sync immediately when the extension starts.</td>
<td>true or false</td>
</tr>
<tr>
<td>endpointPageSize</td>
<td>The page size for retrieving endpoint data for the sync processes. The maximum is 200.</td>
<td>100</td>
</tr>
<tr>
<td>threatPageSize</td>
<td>The page size for retrieving endpoint threat information. The maximum is 200.</td>
<td>100</td>
</tr>
<tr>
<td>cppmUserName</td>
<td>A ClearPass user name. This is for accessing the device profiler.</td>
<td></td>
</tr>
<tr>
<td>cppmPassword</td>
<td>The password for the user name. This is for accessing the device profiler.</td>
<td></td>
</tr>
<tr>
<td>enableStats</td>
<td>Enable basic extension statistics tracking. - true or false</td>
<td>true or false.</td>
</tr>
<tr>
<td></td>
<td>See Appendix B.</td>
<td></td>
</tr>
</tbody>
</table>
Additional Configuration Notes

At a minimum, you’ll need to set these attributes:

- `cylanceTenantId`
- `cylanceApplicationId`
- `cylanceApplicationSecret`

If using the extension to sync device information to the endpoint database, set the `enableEndpointFullSync` and `endpointSyncSchedule` attributes, and optionally the `endpointSyncOnStart` attribute.

The `enableEndpointCache` attribute can be helpful to reduce API calls if an endpoint is authenticating often (for example, a wireless user roaming to multiple access points in a short period of time).

If `includeThreatSummary` is set to true, the following attributes will be added to each endpoint:

- "Cylance Total Files"
- "Cylance Quarantined"
- "Cylance Whitelisted"
- "Cylance Suspicious"
- "Cylance FileRemoved"
- "Cylance Corrupt"
- "Cylance Score"

If the `cppmUserName` and `cppmPassword` attributes are set, the extension can change endpoint profiling data based on information returned from Cylance. To create this user in ClearPass, navigate to Administration > Users and Privileges > Admin Users. Click on Add. Set the Privilege Level to Network Administrator.

**Figure 15: Creating an Admin user on ClearPass**

Pay special attention to the values in the extension configuration. Where a configuration attribute is a Boolean value (true/false), it must not be enclosed with literals.
Configure ClearPass Policy Manager

Multiple methods exist for how ClearPass can utilize the returned data from Cylance Protect, such as:

- Ensure that the endpoint is managed by Cylance. If not, perhaps restrict access for the device on the Corporate Network with a role that only allows the user to remediate and install the Cylance Protect endpoint agent. This provides a controlled environment where all endpoints are protected.
- Check if the endpoint has the latest Cylance agent installed. If not, perhaps quarantine the device with the option to remediate and install and updated Cylance client.
- Another very powerful use case is to check if the actual Cylance agent is running.
- Verify whether the endpoint is identified as ‘safe’ by the Cylance agent. This status flag is used by Cylance to identify whether the endpoint has failed the Cylance security/cyber security checks.
- If you're using the optional includeThreatSummary attribute, and the endpoint is reported as having files/executables that have been identified as malicious/abnormal, then consider dropping the endpoint into a role/vlan where no connectivity is allowed. Optionally, a helpdesk ticket might be raised, perhaps notifying the CISO or Call/Text the user of the infected device to let them know of the criticality of the situation and their options to fix/remediate the identified issue.

As mentioned previously, the extension can be run in two modes.

**Periodic Sync Mode: Using stored attributes**

The Extension can sync data from Cylance Protect periodically, write this data into the ClearPass EndpointDB, then use the EndpointDB as an authorization source. Be aware that this data will not be completely ‘real-time,’ but may be adequate for many use cases.

Here’s an example of a ClearPass Role Mapping Policy that utilizes these Cylance Endpoint Attributes.

**Figure 16: Role Mapping Policy using Cylance Endpoint Attributes**
HTTP Authorization Source Mode: A Real-Time authorization check

In this mode we build an HTTP authorization source, add it to an existing Service Policy, then process the returned data from Cylance, typically utilizing a role-mapping or an enforcement profile action. Note that with an HTTP authorization source configured in a Service, ClearPass will make a call to Cylance every time an authenticating device matches this Service. Making real-time API calls to Cylance every time a device authenticates may not be scalable if there are network delay constraints.

The HTTP Authorization source is the conduit between the extension and ClearPass Policy Manager. It’s through the authorization source that we expose the Cylance endpoint attributes to Policy Manager so they can be used to perform role-mapping or an enforcement policy action.

Defining a Cylance HTTP Authorization Source

The first step is to add the authorization source. Under Configuration > Authentication > Sources, click Add and choose type HTTP.

Figure 17: Adding HTTP authorization source

Provide a Name for the Authentication Source and click Next. On the Primary Tab, provide the IP Address of the extension. Ensure a trailing ‘/’ is added after the extension IP address.

The Base URL is the IP address of the running extension. The Login Username and Password can be set to ANYTHING; they are not used by this extension but the parameters are mandatory.
Figure 18: Defining the authZ Extension IP address

![Fig18]

Click on **Next**. This will advance to the **Attributes** Tab where you must configure at least one attribute. Click on **Add More Filters**. Provide a **Name** for the filter and then a **Filter Query**. The query is indexed off the MAC Address of the endpoint. Copy the line below and paste into the Filter Query box.

```
{% (Connection:Client-Mac-Address-Colon)
```

**Caution:** It's extremely important that the Filter Query is defined correctly. This is the query string sent to the Cylance Extension requesting context about the endpoint.

Figure 19: Building the Cylance Extension query filter

![Fig19]

The data returned from Cylance can be extensive, but not all the data is relevant to making a security policy decision about the endpoint. Choose which attributes are required for your enforcement profile. As an example, the filter shown below has six attributes configured. Your filter may differ depending on the use-cases you are trying to meet. A complete list of the available fields is shown in **Appendix A**.
Using Results from the Cylance HTTP Auth Source

Below is an example of using these attributes in a ClearPass Role Mapping Policy.

Figure 21: Building a Role Mapping Policy to utilize Cylance HTTP Auth Source attributes

This is just a simple configuration to show what you can do with the returned attributes. In the Appendix we step through the process of adding Cylance to enhance the enforcement in an existing ClearPass Service.
Appendix A – List of returned Cylance endpoint attributes

{
    "Cylance ID": "59284448-3ce6-4570-bcb3-b88686a6b176",
    "Cylance Name": "NS-TME-X230",
    "Cylance Score": "0",
    "Cylance State": "Offline",
    "Cylance Policy": "Default",
    "Cylance Corrupt": "0",
    "Cylance Is Safe": "true",
    "Cylance Is Found": "true",
    "Cylance Host Name": "ns-tme-x230.socialwifilogin.net",
    "Cylance OS Version": "Microsoft Windows 8.1 Enterprise",
    "Cylance Suspicious": "0",
    "Cylance FileRemoved": "0",
    "Cylance Last Update": "2020-03-18 16:40:05",
    "Cylance Quarantined": "0",
    "Cylance Total Files": "0",
    "Cylance Whitelisted": "0",
    "Cylance Date Offline": "2020-02-20 04:58:48",
    "Cylance IP Addresses": "15.111.206.62",
    "Cylance Agent Version": "2.0.1540",
    "Cylance MAC Addresses": "60-67-20-01-E4-2A",
    "Cylance Update Available": "false",
    "Cylance Last Logged in User": "SOCIALWIFILOGIN\danny",
    "Cylance Background Detection": "false",
    "Cylance Date First Registered": "2017-03-23 23:09:56"
}

Note: The following attributes are included only if "includeThreatSummary" is set to true.

- “Cylance Total Files”
- "Cylance Quarantined"
- "Cylance Whitelisted"
- "Cylance Suspicious"
- "Cylance FileRemoved"
- "Cylance Corrupt"
- “Cylance Score”
Appendix B – Troubleshooting and Support

Check API Access Application Control restrictions

If you've previously hardened your ClearPass deployment with Application Access Controls, it's possible that the Extension will not work. Reviewing the Extension Log might show something like the following after immediately starting the Extension. This likely indicates the ClearPass Application API's are in place.

*Figure 22: Example of Extension authorization failure due to Policy Manager Application Control*

```
<head>
  <title>Error 403 (Forbidden)</title>
  <script language="javascript">
    function reloadPage() {
      var locHref = window.location.protocol + "//" + window.location.hostname;
      window.location.href = locHref;
    }
  </script>
</head>
```

To resolve this issue, add the IP address of the Extension to the list of nodes permitted to access the API by navigating to Administration > Server Manager > Server Configuration {choose your node} > Network

*Figure 23: Locking down access to the Policy Manager API for extensions*

```markdown
Note: For this reason its good practice to fix the IP address of the extension at installation time such that it doesn't change over time and break the application controls.
```
Checking on the Extension Service

The ClearPass Extensions are supported by a system service which must be running.

**Note:** Restarting this service will affect all deployed and running extensions.

To check on the state of the Extension Service, or to restart the service, go to **Administration > Server Manager > Server Configuration > [SERVER] > Service Control**. By default this service is automatically started.

*Figure 24: Services Control*

<table>
<thead>
<tr>
<th>Service Name</th>
<th>Status</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Async group notification service</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>2. Async DB write service</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>3. Async network services</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>4. ClearPass Role service</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>5. DB change notification server</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>6. DR replication service</td>
<td>Running</td>
<td>Stop</td>
</tr>
<tr>
<td>7. Extensions service</td>
<td>Running</td>
<td>Stop</td>
</tr>
</tbody>
</table>
**Extension Logs/Debugging**

If you have a requirement to access and view the logs from the Extension, you can turn on different logging levels from the Extension GUI. Adjust the **logLevel** to ‘**DEBUG**’ and restart the extension as shown below.

Logs can then be viewed from the ‘**Show Logs**’.

**Figure 25: 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 the extension logs using ‘Collect Logs’ system function

In addition to viewing the logs as shown above, logs can also be collected and examined via the Policy Manager Collect Logs system function (Administration > Server Manager > Server Configuration > [Select SERVER] > Collect Logs). This is extremely useful should you have a need to call for technical assistance.

If the support team needs to investigate a system issue, one of the items they regularly ask for is the system logs to aid with their diagnostic investigation. By default the “logLevel” is set to INFO, but TRACE, DEBUG, INFO, WARN, ERROR, FATAL can also be set as required. Any of the levels will display the information for the selected state and lower; if INFO is selected, it will show messages for INFO, WARN, ERROR, FATAL.

After the logs have been collected, downloaded and expanded, you can locate the extension logs in the following location in the folder structure PolicyManagerLogs > extension > your-extension-id as shown below.

Figure 26: Locating the Extension logs from ‘Collect Logs’ diagnostic GZ file
Monitoring extension statistics

There is a way to monitor extension's critical statistics with the configurable parameter added as part of the extension's configuration. To enable extension statistics set the “enableStats” parameter to true.

Figure 27: Enable extension statistics
To navigate to statistics page, click **Show Details**.

**Figure 28: Show Extension Details**

Open extension statistics URL:

**Figure 29: Show Extension Statistics URL**
This will show statistics similar to the following:

**Figure 30: Extension Statistics**

### Outgoing HTTP(s) Requests

- **Monday, Mar 2, 2020**
  - HTTP 2xx: 12
  - HTTP 3xx: 0
  - HTTP 4xx: 0
  - HTTP 5xx: 0
  - Connection Errors: 0

### CPU Usage

- **Monday, Mar 2, 2020**
  - CPU (%): 5.9%

### Memory Usage

- **Monday, Mar 2, 2020**
  - Resident Set Size (RSS): 43.39 MB
  - Heap Total: 22.66 MB
  - Heap Used: 18.70 MB
  - External: 25.22 KB
Appendix C – Considerations for Installing in a Cluster

Extensions are not synced between ClearPass cluster members, and thus must be installed on each member separately.

The Cylance Extension can run in two modes: Periodic Sync Mode and Authorization Source Mode.

**Periodic Sync Mode**

If you are configuring the extension to poll Cylance periodically and utilize the resulting ClearPass Endpoint database during endpoint Authorization, then you only need to install the extension on one cluster member, often the publisher.

You may wish to install the extension on a second cluster member as a backup, but remember that both extensions will individually be updating the endpoint database. You may want to stagger the updates between the two extensions, for example, Subscriber1 updates at the top of the hour and Subscriber2 updates at 30 minutes after the hour.

Also, in this mode there is no need to explicitly enter an IP address during installation. The defaults will suffice and ClearPass will select an IP in the range specified in the server configuration.

**HTTP Authorization Source Mode**

In this mode we configure an HTTP Auth source that results in a TCP call to Cylance during endpoint authorization. In this deployment model the extension must be installed on every cluster node that process authentications. Also in this scenario every cluster member’s extension must be set to the exact same IP address during installation time, as the HTTP Auth source configuration is propagated globally across all cluster members.

For example, if the extension IP range is 172.17.0.0/16, we would set the extension to 172.17.0.5 on every cluster member during installation of the extension.

While we normally want to avoid duplicate IP addresses in a network, this is not a concern with ClearPass extensions. Each ClearPass node communicates internally only with its own extension, and this traffic is not routed outside of ClearPass.

Subscriber nodes support the same ability as publishers to install an Extension from the Extension store.
Appendix D – Considerations when installing outside of the USA

When installing this integration with a Cylance Cloud outside of the USA one specific configuration attribute in the extension must be changed. By default the assumption is made the Cylance Cloud console is running in North America. If that is not the case the below config line must be updated to reflect this.

"cylanceSubDomain": "protectapi"

The below table is from the Cylance Admin Guide.

**Figure 31: Cylance Region Codes**

<table>
<thead>
<tr>
<th>Region Code</th>
<th>Description</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia-Pacific – North</td>
<td>apne1</td>
<td></td>
</tr>
<tr>
<td>Asia-Pacific – Southeast</td>
<td>au</td>
<td></td>
</tr>
<tr>
<td>Europe – Central</td>
<td>euc1</td>
<td></td>
</tr>
<tr>
<td>Government</td>
<td>us</td>
<td>No code required.</td>
</tr>
<tr>
<td>North America</td>
<td></td>
<td>protectapi</td>
</tr>
<tr>
<td>South America</td>
<td>sae1</td>
<td>protectapi-sae1</td>
</tr>
</tbody>
</table>

Based on the following chart set the **cylanceSubDomain** attribute appropriately.

**Figure 32: Regional Cylance Cloud settings**

<table>
<thead>
<tr>
<th>Region</th>
<th>Value</th>
<th>Extension setting for cylanceSubDomain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asia-Pacific -North</td>
<td>apne1</td>
<td>protectapi-apne1</td>
</tr>
<tr>
<td>Asia-Pacific -Southeast</td>
<td>au</td>
<td>protectapi-au</td>
</tr>
<tr>
<td>Europe</td>
<td>euc1</td>
<td>protectapi-euc1</td>
</tr>
<tr>
<td>Government</td>
<td>us</td>
<td>protectapi-us</td>
</tr>
<tr>
<td>North America</td>
<td></td>
<td>protectapi</td>
</tr>
<tr>
<td>North America</td>
<td>sae1</td>
<td>protectapi-sae1</td>
</tr>
</tbody>
</table>
Appendix E – endpointSyncSchedule settings

The `syncAllSchedule` parameter sets how often ClearPass reaches out to Cylance Protect to update the endpoint database. This setting is based on a slightly modified version of the CRON job scheduler found in Unix-like operating systems. It can be used to schedule jobs to run periodically at fixed times, dates or intervals.

A ‘cron’ is a job scheduler. Any scheduled task is called a ‘cron job’. The syntax for a cron job schedule is as follows:

*Figure 33: Cron Job Reference*

In our use of the cron scheduler, we’ve dropped the use of the last instruction `<command to execute>` and use only the time/date functions, see below for a number of examples of scheduling a sync process.

- Schedule a sync to run at 2am daily: `- 0 2 * * *`
- Schedule a sync to run twice a day at 5am and 5pm: `- 0 5,17 * * *`
- Schedule a sync to run on every Sunday at 5pm: `- 0 17 * * sun`
- Schedule a sync to run every 30 minutes: `- */30 * * * * *`
- Schedule a sync to run at 5pm on selected days: `- 0 17 * * sun,fri`

You can see from the above that the scheduling process is extremely flexible.
Appendix F – Using the Cylance Extension information in a ClearPass Service

In this appendix, we’ll provide more detail on how to incorporate information from the Cylance Extension in an existing ClearPass service. We’ll add the Cylance HTTP Auth source to an existing basic Wired 802.1X service that processes authentications from ArubaOS switches. This service checks Active Directory to set roles, then uses those roles in an enforcement policy to send different RADIUS attributes to the switch. The configuration of your existing policies will likely differ, so please use this appendix as a general guide.

Service Authorization

The first step is to add Authorization to our service by checking the Authorization box on the Service tab. Checking this box makes the Authorization tab visible in the service definition.

Note: your existing services may already have this box checked.

*Figure 34: Add Authorization to ClearPass Service*

Proceed to the Authorization tab and add the Cylance HTTP Auth Source you created during configuration. We also added the [Time Source] [Local SQL DB] for evaluating dates and times.
**Role Mapping Policy**

Our Role Mapping Policy originally only contained the first two lines seen below.

We first created the role names in ClearPass by navigating to **Configuration > Identity > Roles**. We then added the subsequent lines to the role mapping policy to set roles based on the returned values from Cylance. Note this is an evaluate-all Role Mapping Policy, so all matching roles will be applied.

**Figure 35: Authorization Sources**

**Figure 36: Modify Role Mapping Policy**
Custom Time Source Attribute

Note line 3 of the Role Mapping Policy contains a custom value of

\%{Authorization:[Time Source]:two_weeks_ago}

This value allows us to quarantine a device if it has not checked in with Cylance in two weeks. The creating of this custom value is outside the scope of this document, a screenshot of how we created it is provided here.

**Figure 37: Add Custom Time Source Attributes**

![Add Custom Time Source Attributes](image)

Enforcement Policy

Finally we evaluate the roles set in the Role mapping policy to send different enforcement values to the switch. In this case we sent RESTRICTED enforcement for Employees who’s Cylance is out of date or have a Cylance Quarantined role. Employees who meet the Cylance posture requirements receive the EMPLOYEE enforcement profile.

**Figure 38: Modify Enforcement Policy**

![Modify Enforcement Policy](image)