Enhanced ClearPass Device Visibility

Derin Mellor  
Blue Sky Systems   
derin.mellor@blueskysystems.co.uk  
Version DRAFT 0.3  
13th July 2021

Contents

[Objective 1](#_Toc77072754)

[Distributed or Large Networks 2](#_Toc77072755)

[Define Scanning Techniques SNMP, SSH & WMI 3](#_Toc77072756)

[Define Scanning Technique NMAP 4](#_Toc77072757)

[Initiating Scans 7](#_Toc77072758)

[Results 8](#_Toc77072759)

[Identifying Common Fingerprints 12](#_Toc77072760)

[Custom Fingerprint 14](#_Toc77072761)

[Edit/Delete Existing Fingerprint 16](#_Toc77072762)

[New Classification 20](#_Toc77072763)

[Automate 21](#_Toc77072764)

# Objective

ClearPass has built-in tools that enable it to proactively scan devices on the network. This is not dependent on any authentication.

ClearPass’ most mature profiling technique is DHCP Request profiling. This usually relies on relaying the DHCP Request packet the ClearPass. All you need do is add a DHCP Relaying at the Access Router to send these to ClearPass, either interface. ClearPass will process the packet – but never responds.

This is very effective for DHCP devices, but its not always possible to change the DHCP Relaying (eg managed service environments) and most networks still have large numbers of devices with static IP addresses.

ClearPass provides two proactive scanning techniques:

1. Hierarchical scan
2. Subnet scan

The hierarchical scan identifies and profiles network access devices (NAS: switches, controllers, routers, etc) and endpoints (servers, computers, IoT devices, etc) that they know about. This is a recursive process that starts at “seed routers” and hops down interrogating devices with routing and forwarding tables ultimately probing endpoints:

1. SNMP read information from its router, bridge, ARP, LLDP and CDP MIBs. This information is used to discover:
   1. Endpoints
   2. Neighbouring NAS
2. Scan each Endpoint
3. Repeat this process for each neighbouring NAS.

This continues until the scan depth limit is reached. By default this scan depth is configured to 3 “hops”. Invariably this will discover both the MAC address and IP address – assuming the scan is deep enough. Care needs be taken to prevent this scan going too far!

The subnet scan explicitly scans all the IP addresses within the defined subnet. This is particularly useful to identify quiet devices. However, it should be used in conjunction with SNMP ARP scanning of NAS (within the NAD definition) to resolve the IP to MAC address mapping.

The scanning techniques employed comprise one or more of the following:

SSH If port 22 is open use SSH to login and collect profiling information

WMI If port 135 is open use WMI to login and collect profiling information

SNMP If port 161 is open use SNMP to collect profiling information

NMAP This scans all TCP ports on open state\* , this can be very process intensive

\*Note: By default, if “Enable Endpoint Port Scans using Nmap” is disabled it will only scans TCP ports 135 & 3389 – these are typical of a Windows PC. This can be changed to scan a raft of common open ports. If the “Enable Endpoint Port Scans using Nmap” is enabled it scans all 64K worth of ports and is very process intensive.

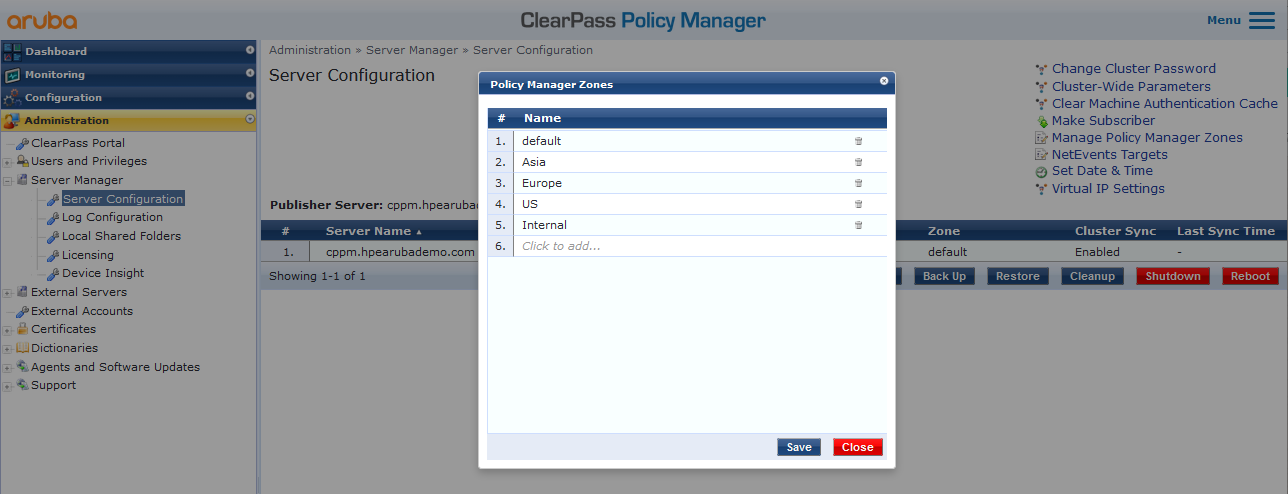
NOTE: You can craft highly focused scans that can be associated with a ClearPass profile. Hence, when a device connects ClearPass executes the device types profile scan… This document does not discuss this.

WARNING: One weakness of NMAP scanning is that it only looks for open TCP ports. Open UDP ports and closed ports generally are ignored. This is a shame.

NOTE: By the very nature of these scans ClearPass is only looking for open ports on the device, not for communications from the device. Some switches can inform ClearPass of sessions (ie IP-protocol, source IP & port, destination IP & port) via flow information. However, ClearPass only processes the ingress sessions and I have concern over the scalability.

# Distributed or Large Networks

In these environments it is highly desirable to configure Zones. Zones define a logical grouping of ClearPass – this is typically associated with physical locations – eg Europe, USA, Asia, SiteA, SiteB, etc.

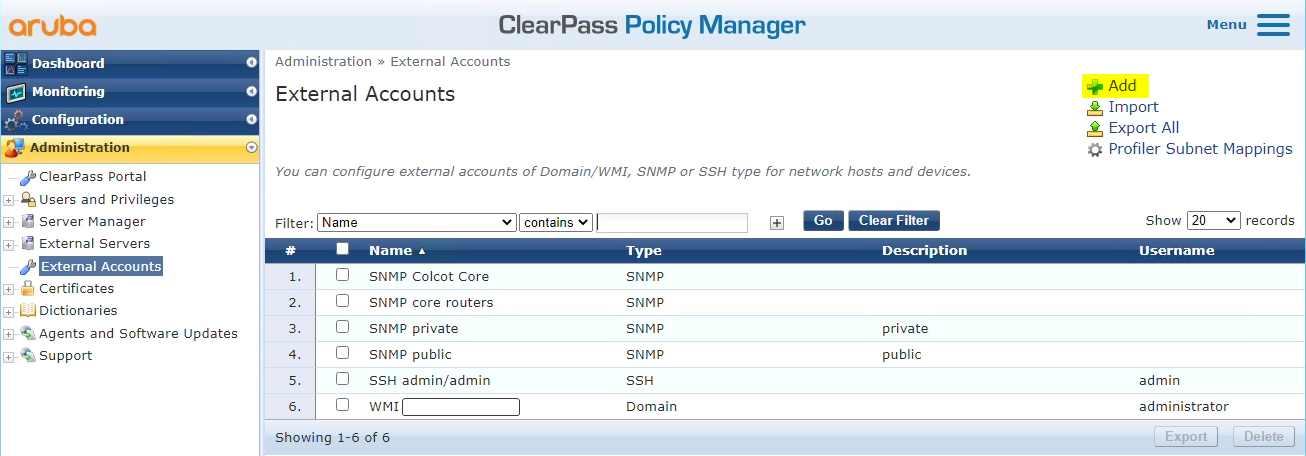


ClearPass should be associated with their particular zone – they will be responsible to scanning within this zone.

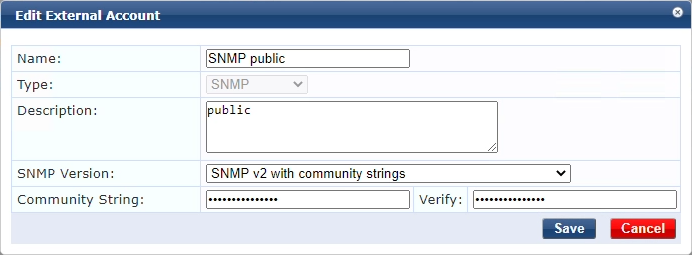


Hence, a level of localization and load-distribution can be achieved.

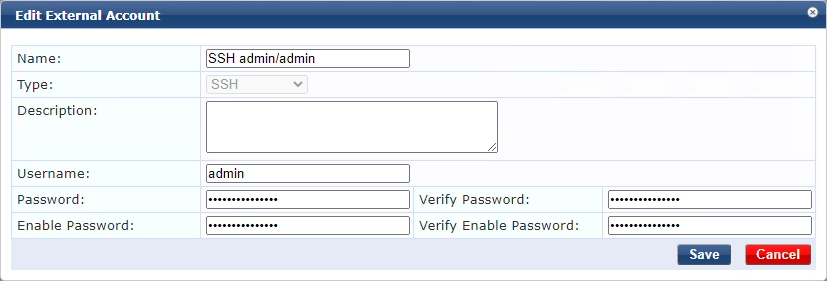
# Define Scanning Techniques SNMP, SSH & WMI



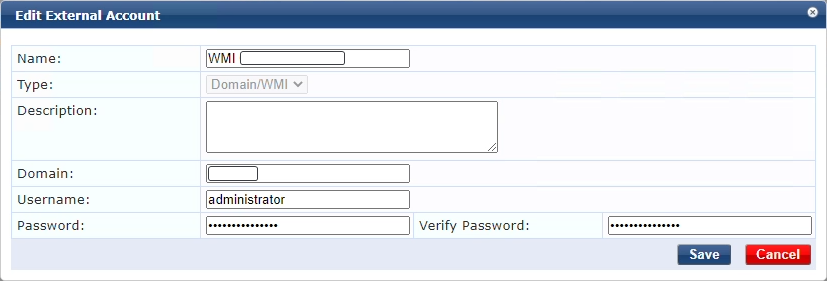
Adding a new technique is trivial:



I find the SNMP the most useful scanning technique as it’s very accurate – particularly when using V3. It’s surprising how many devices have public as a Read Only preconfigured.



Currently the SSH really is only useful for Linux environments. This is possibly useful to identify devices that have default username/password (eg IoT).

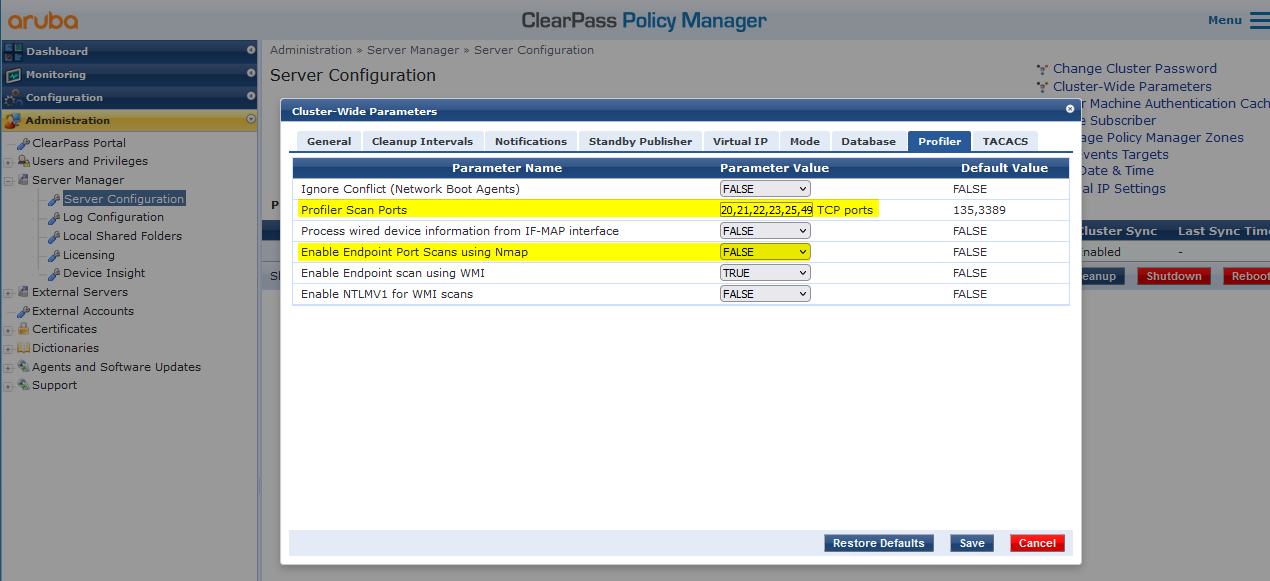


This can be useful for a fresh install that has no authentication setup. Obviously, this is restricted to Windows environments. This needs to be enabled in Cluster-Wide Parameters – see Define Scanning Technique NMAP below.

# Define Scanning Technique NMAP

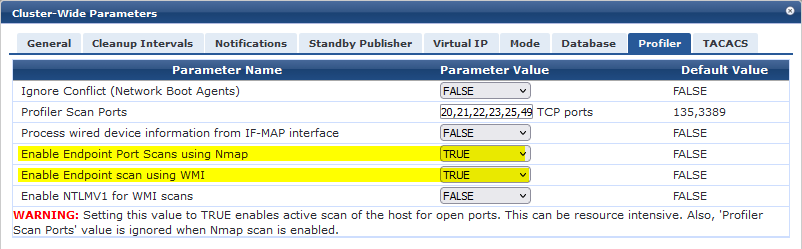
NMAP on its own can be very inaccurate, it’s best to use this in conjunction with something else – eg the MAC address’ OUI.

By default NMAP will only scan TCP 135 & 3389 ports. However, this can be changed:



A good list of well-known TCP ports are: 1,3,7,9,13,17,19,21,22,23,25,26,37,53,79,80,81,82,88,100,106,110,111,113,119,135,139,143,144,179,199,254,255,280,311,389,427,443,444,445,464,465,497,513,514,515,543,544,548,554,587,593,625,631,636,646,787,808,873,902,990,993,995,1000,1022,1024,1025,1026,1027,1028,1029,1030,1031,1032,1033,1034,1035,1036,1037,1038,1039,1040,1041,1044,1048,1049,1050,1053,1054,1056,1058,1059,1064,1065,1066,1069,1071,1074,1080,1110,1234,1433,1494,1521,1720,1723,1755,1761,1801,1900,1935,1998,2000,2001,2002,2003,2005,2049,2103,2105,2107,2121,2161,2301,2383,2401,2601,2701,2717,2869,2967,3000,3001,3128,3268,3306,3389,3689,3690,3703,3986,4000,4001,4045,4343,4899,5000,5001,5003,5009,5050,5051,5060,5101,5120,5190,5357,5432,5555,5631,5666,5800,5900,5901,6000,6001,6002,6004,6112,6646,6666,7000,7070,7937,7938,8000,8002,8008,8009,8010,8031,8080,8081,8082,8088,8443,8888,9000,9001,9071,9090,9100,9102,9999,10001,10010,32768,32771,49152,49153,49154,49155,49156,49157,49158,50000

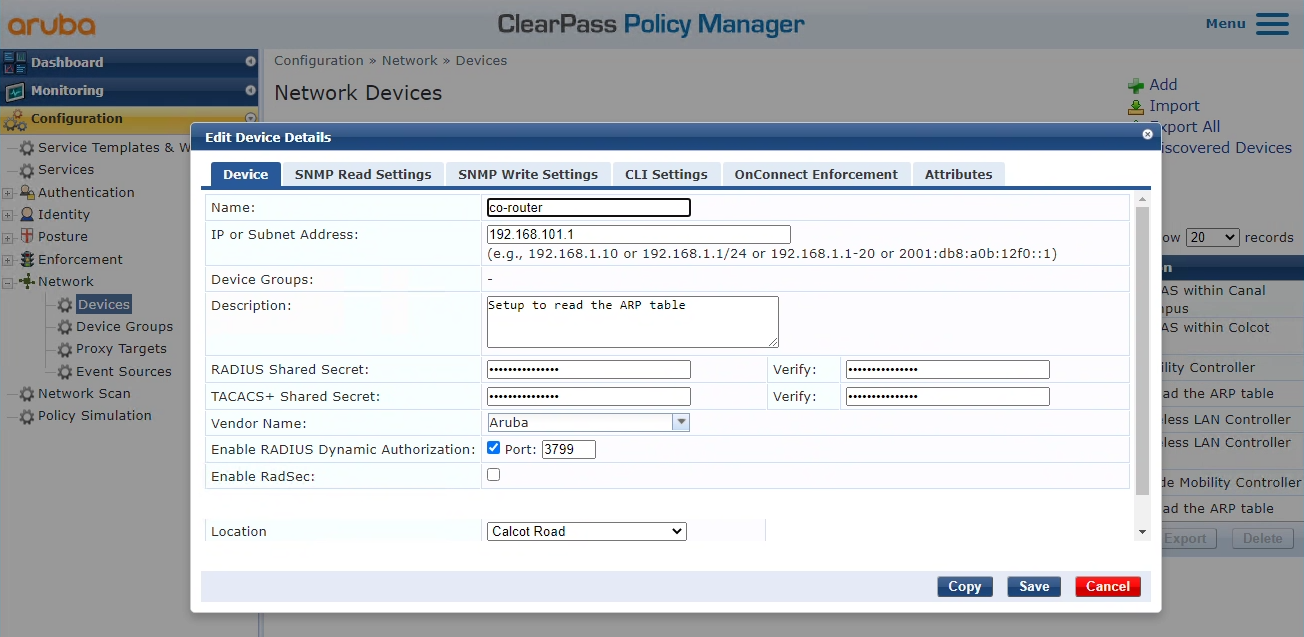
Obviously, you can adjust accordingly.

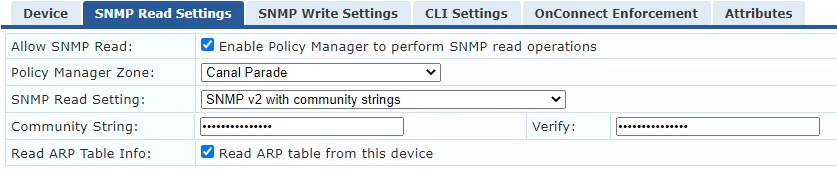
However, for the initial scan I feel it is best to be comprehensive: 

This is also where you enable WMI scanning.

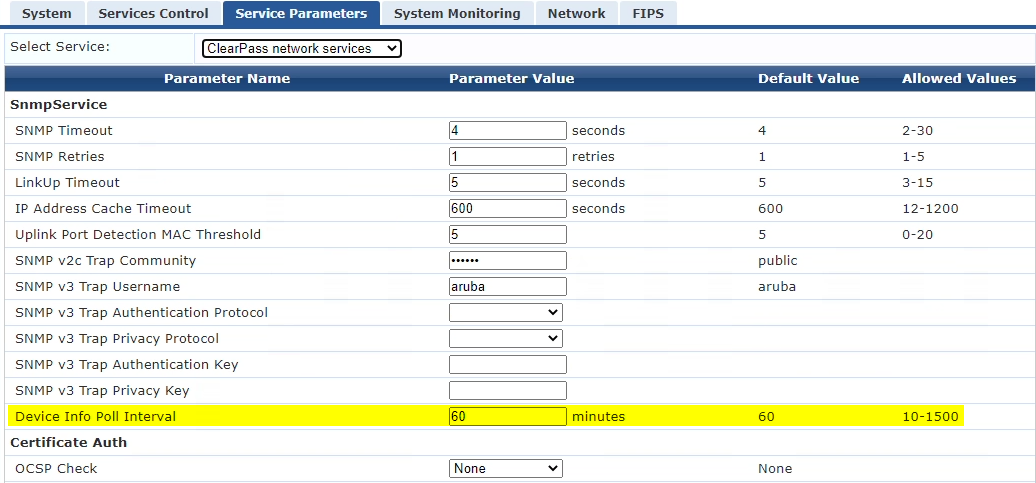
Define NAD ARP Polling

Polling the ARP table is useful to help mapping the IP address to the MAC address. Once authentication (specifically RADIUS Accounting with the Frame-IP-Address and Framed-IPv6-Address with Interim packets) is configured the ClearPass will learn the device’s IP address(es). However, prior to this the recommended approach is to configure SNMP Read of the ARP table of the key Access Routers:



within the SNMP Read Settings:

The speed of the polling is determined within the PolicyManager🡪Administration🡪ServerManager🡪ServerConfiguration: Select ClearPass appliance:

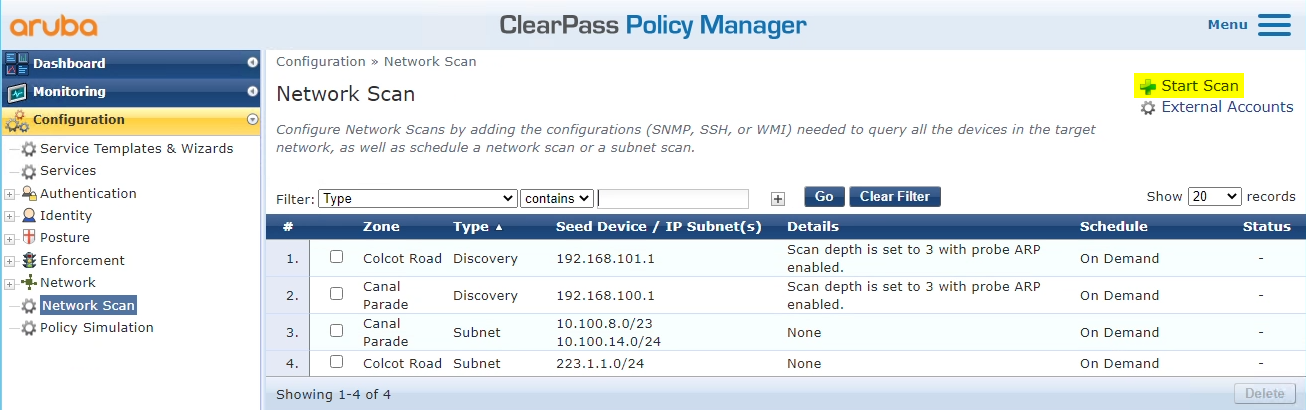


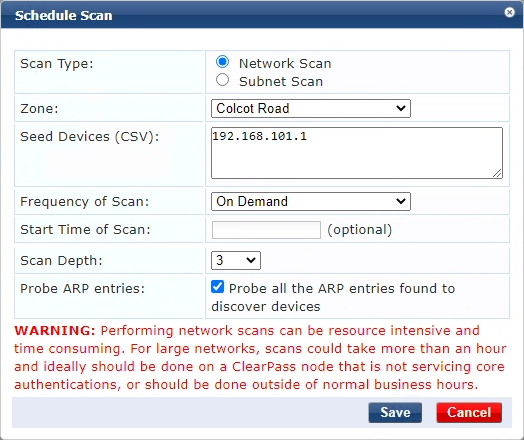
Often scanning will learn devices purely with an IP address. Internally ClearPass will create a *phantom MAC address* indicated starting with the ‘x’ character – ie. x01122334455.

WARNING: Occasionally, a device can be learnt via its MAC address (eg DHCP Request) and via its IP address (eg scanning) but ClearPass does not always tie these together. The end result is you have two entries in the Endpoints table.

# Initiating Scans

Start Scan…





On demand   
or scheduled

Zone – identifies the ClearPass appliances to do scan

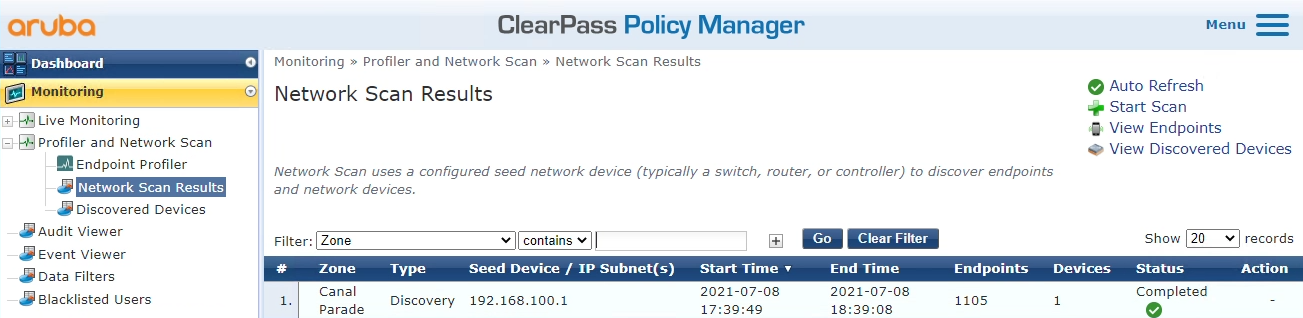
Neworks Scan=Hierarchical  
Subnet Scan

Specify the list of routers, comma separated

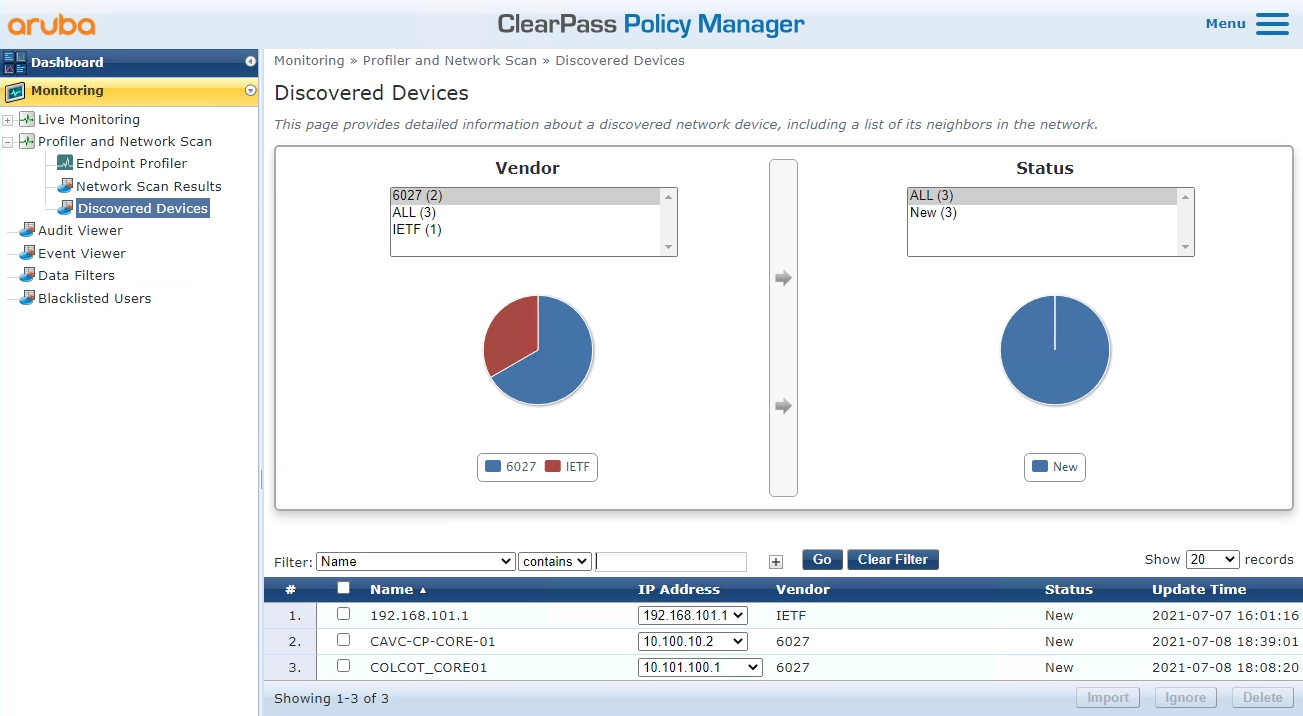
For the first scan I leave this as On Demand and use a full NMAP scan.

Subsequent scheduled scans its preferable to use a more focused scan.

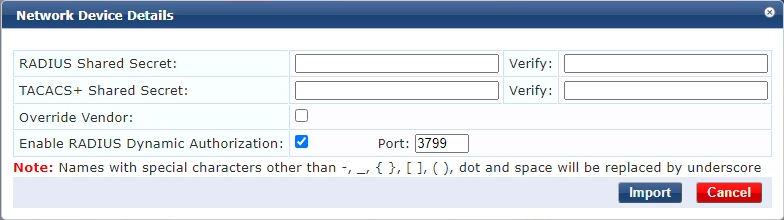
# Results

To monitor scanning, and abort scans, looking in the Network Scan Results:

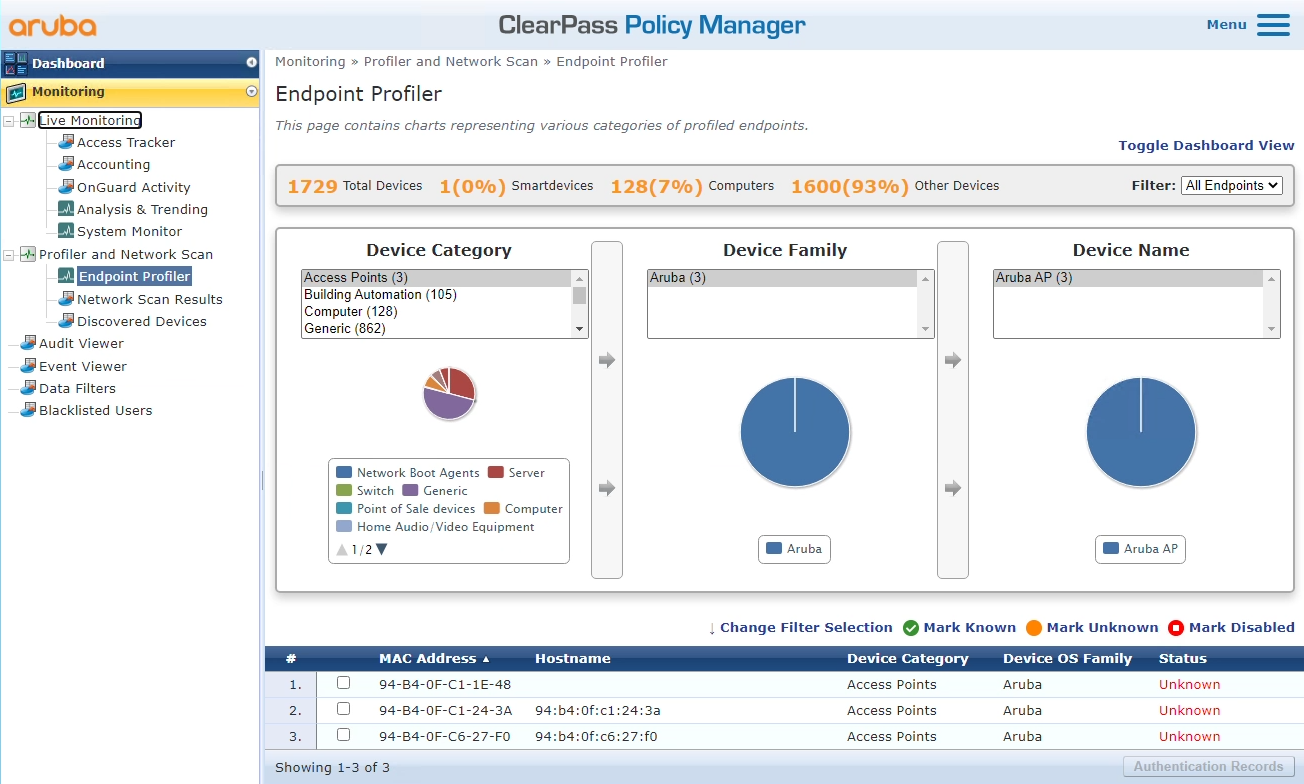
To see the NADs learnt:



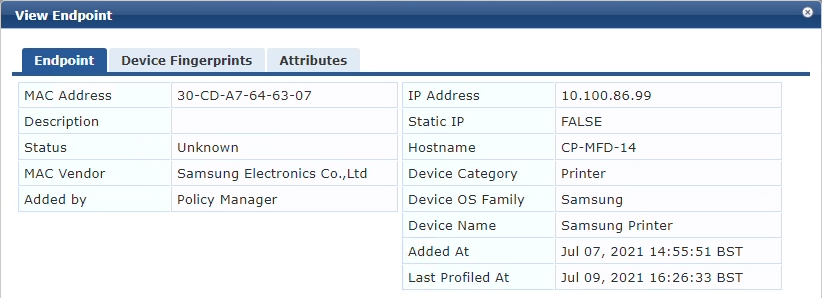
By selecting NAS you can quickly import as NADs – this is really convenient to populate the NADs for a new site. It allows you to configure the RADIUS and TACACS details directly:

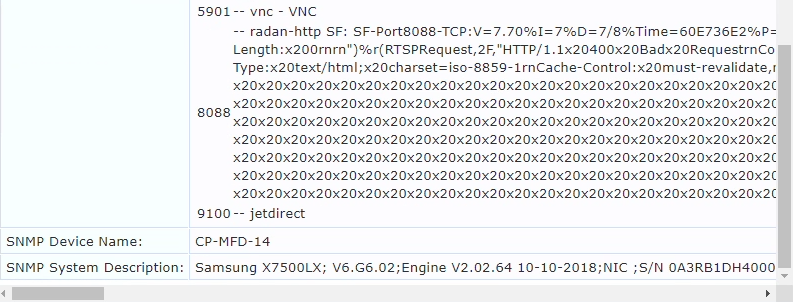
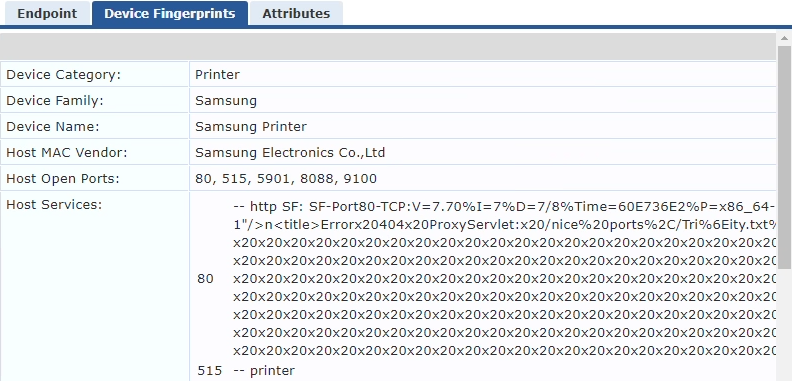
Alas, other details, like SNMP, have to be configured afterwards.

Learnt Endpoints:

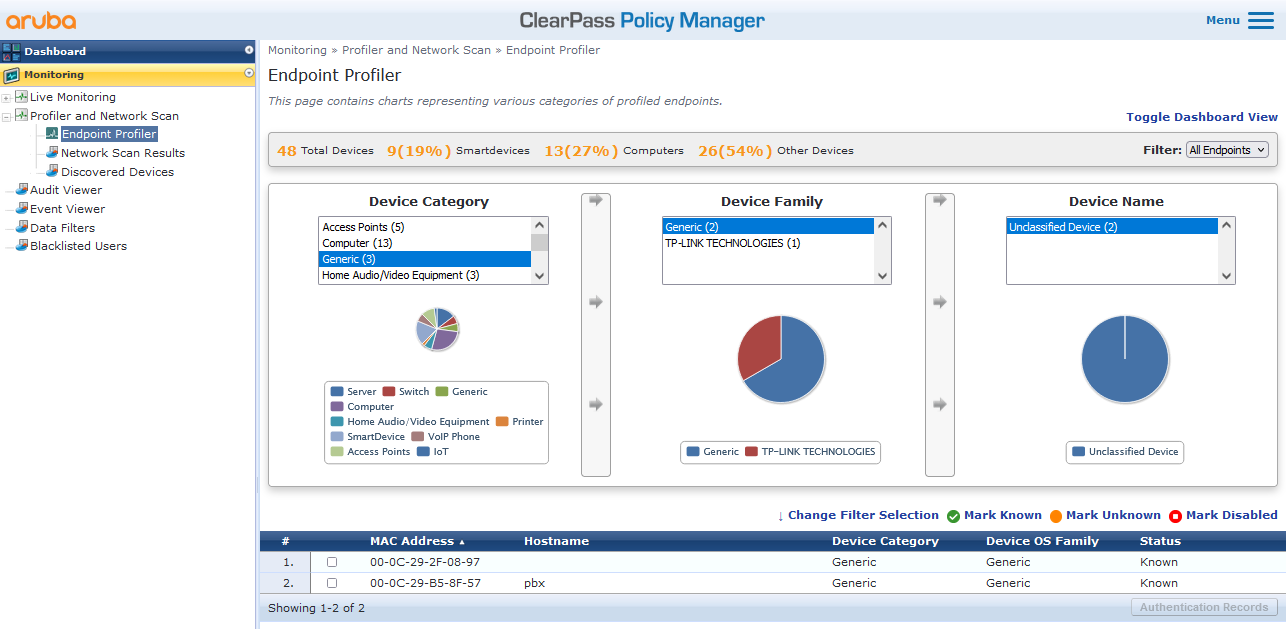


If you select a MAC address it will show you the Device Fingerprints tab:





This is fine when the device has been fingerprinted correctly.

Often, particularly with NMAP, the accuracy is questionable. There are not many built-in NMAP Profiles. Devices that do not match a fingerprint rule are placed onto the Generic category:

Note: Device Family will automatically report the manufacturer’s name (based on the OUI) – if it is available.

# Identifying Common Fingerprints

There are few products in the market place that use sensors to monitor traffic and machine learning to categorize products. This is typically based on Jaccard Similarity and uses group theory to determine a good match. Logically:

Jaccard Similarity = (number of observations in both sets)/(number in either set)

Or, written in notation form:

**J(A,B)** = |**A**ꓵ**B**|/|**A**ꓴ**B**|

This will be explored later.

Humans are fairly good at interpreting this type of data.

The challenge with ClearPass is extracting the information. Alas the actual endpoint’s fingerprint is not exposed in the RESTful API. Instead you have to use the PostGRESQL interface. With the appropriate SQL you can extract all the endpoints and their associated fingerprint components into a CSV file.

I export it into columns:

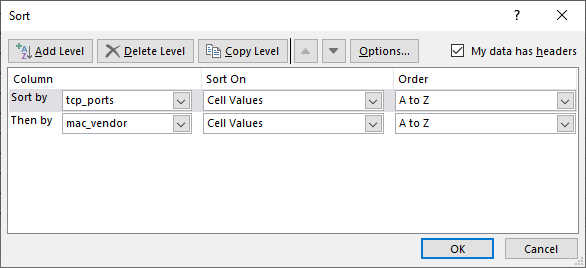
mac, mac\_vendor, ip, static\_ip, hostname, device\_category, device\_family, device\_name, dhcp, tcp\_ports, services, snmp.

NMAP profiling alone is only reliable if it is very distinctive. For example, an Aruba Controller’s AOS8 open ports are usually: 17, 21, 22, 80, 443, 1723, 4343, 8080, 8081, 8082, 8088 (Mobility Controller also has 9071), with common services ["17:tcpwrapped", "21:ftp - Aruba router ftpd", "80:http - Apache httpd", "443:http - Apache httpd", "1723:pptp", "4343:http - Apache httpd", "8080:http - nginx", "8081:http - nginx", "8082:http - Apache httpd", "8088:http - nginx]

The SSH (22) varies!

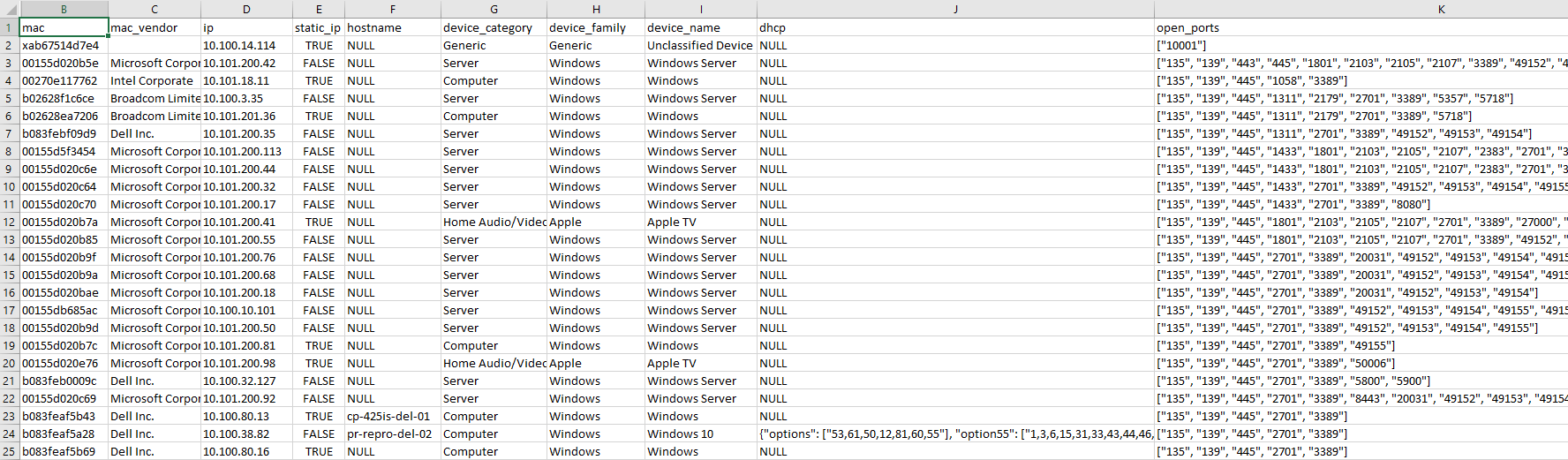
0lder 6.x is missing 8082. The service details are significantly different.

IAP: 22, 80, 443, 4343, 8080 with services ["22:ssh - OpenSSH Version: 7.1", "80:http - mini\_httpd", "443:https", "4343:unicall", "8080:http-proxy - tinyproxy Version: 1.8.2"]

In Excel order the file based on tcp\_ports and mac\_vendor:

This will highlight common tcp open ports:

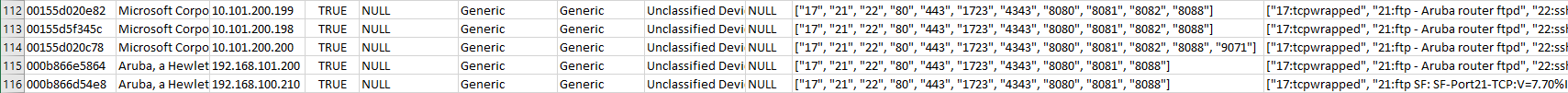
Phantom MAC address



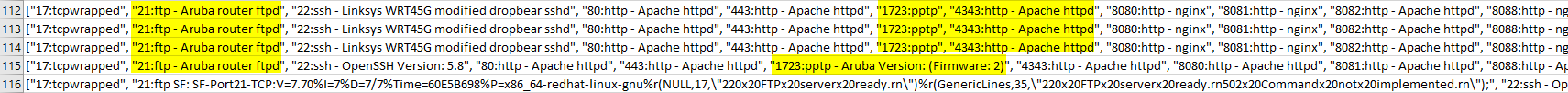
From here it is an easy exercise to see devices with common open ports, and more detailed services. Problem with the services is they often change and may indicate version number, though they can be really useful in identifying the device. A human validation of the device maybe required.

Irrespective, the CSV can have a status column added and then it can become a network audit and re-imported back into ClearPass – see <https://ase.arubanetworks.com/solutions/id/91>

## Custom Fingerprint

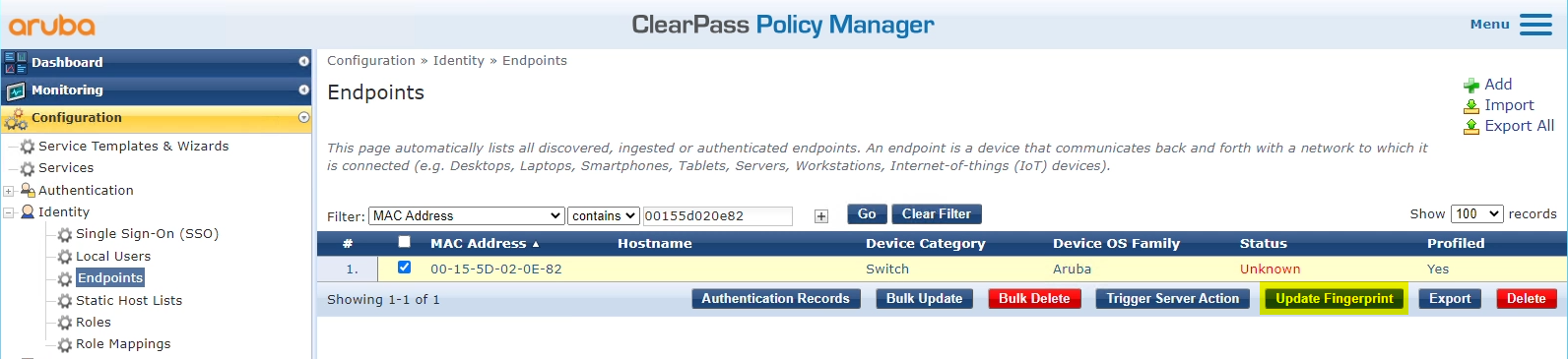
Often devices that are profiled with NMAP do not match a built-in filter. By looking at the open ports/services it can sometimes be identified. For example here are the customer’s Aruba Controllers:

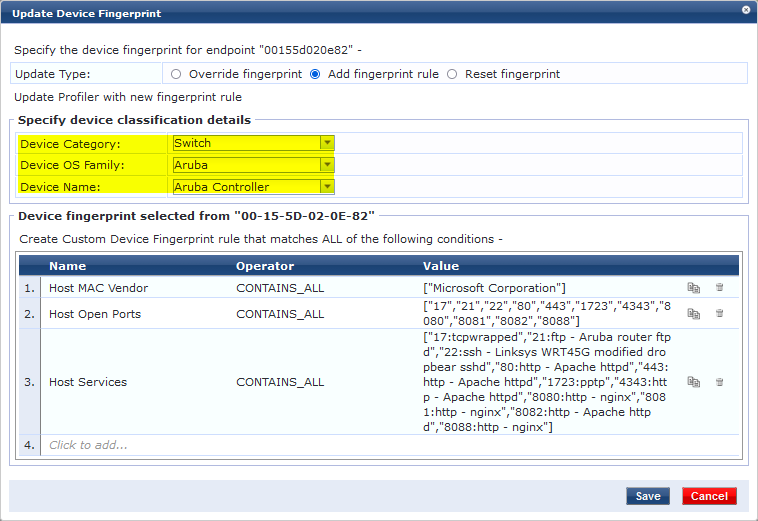
10.101.200.200 is the VIP, though not sure what 192.168.100.210 is, but it is definitely Aruba.



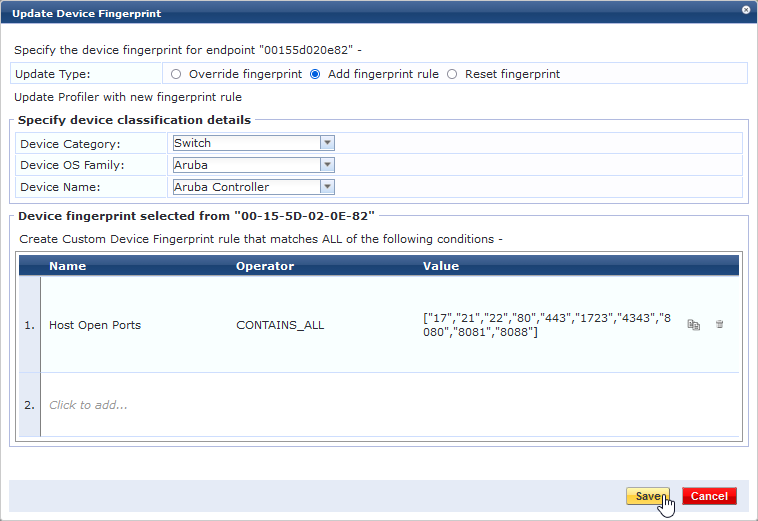
The common open TCP ports are 17, 21, 22, 80, 443, 1723, 8080, 8081, 8088.

Aruba Controllers already has a classification (Category/Family/Name) of Switch/Aruba/Controller. Hence, we can create a rule that any device connects that has these open ports will be associated with the Aruba Controller fingerprint.

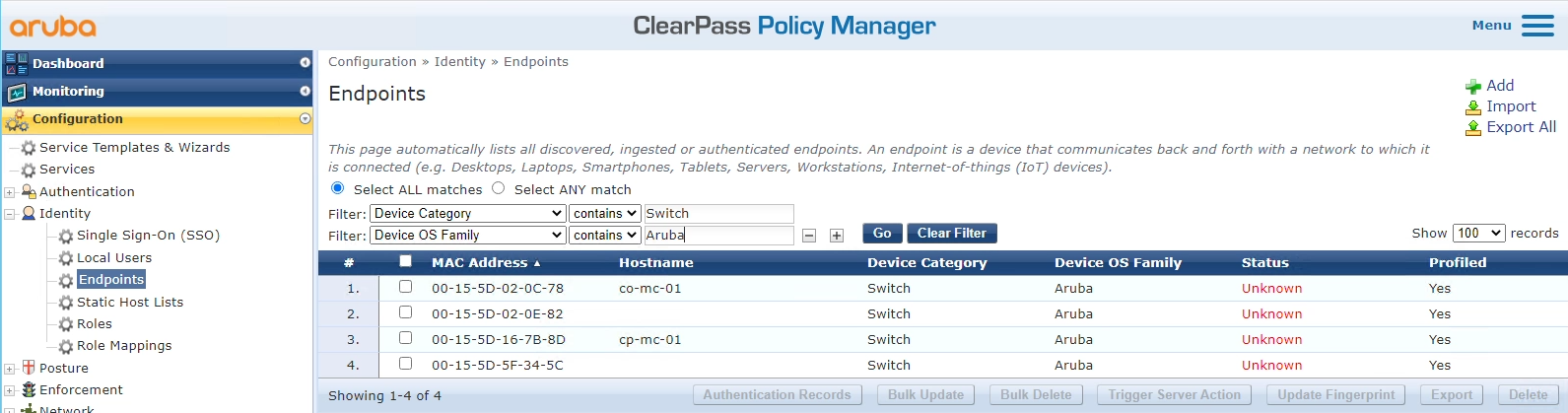
To add a new fingerprint: Search for the MAC address of the device, and update its fingerprint:



Trim the fingerprint rules down:

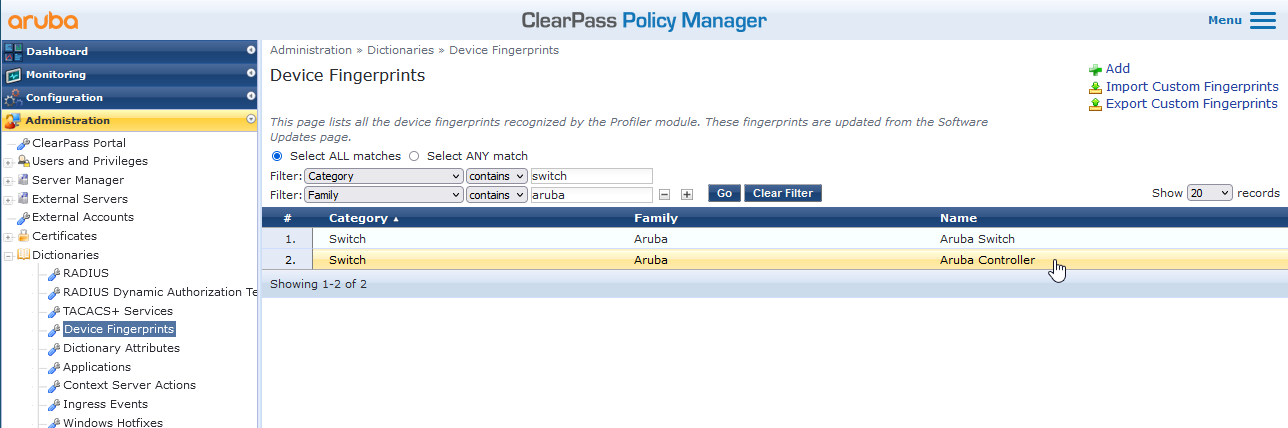


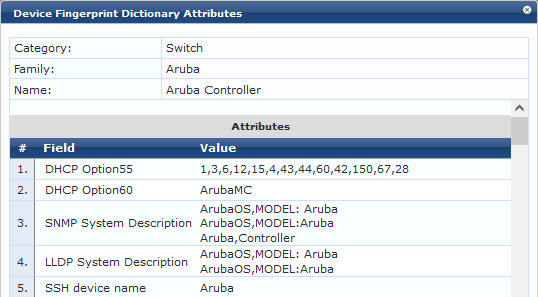
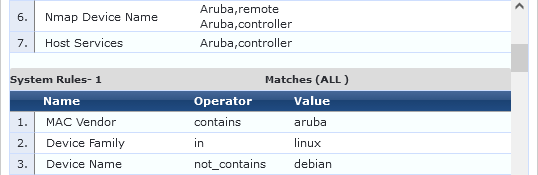
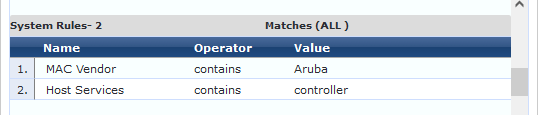
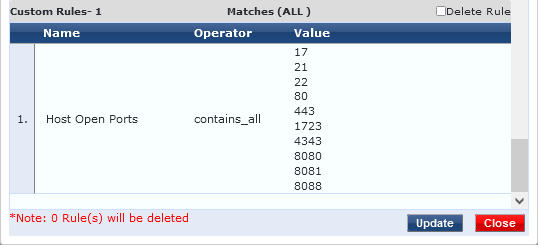
Once saved all matching devices will be inherit this fingerprint:



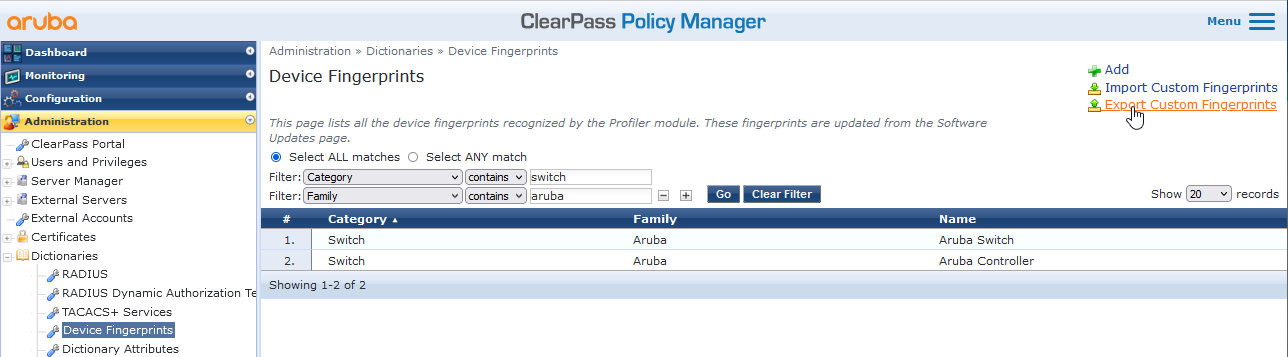
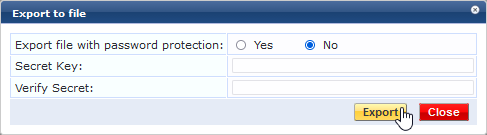
WARNING: You can add multiple fingerprint rules. But be careful, I believe if searches down these rules and finds the first match. Hence, you need to order these most specific first. See below.

## Edit/Delete Existing Fingerprint



You can’t directly edit anything from here. You have to export to XML:



There are no secrets in here so export without a secret.

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>

<TipsContents xmlns="http://www.avendasys.com/tipsapiDefs/1.0">

<TipsHeader exportTime="Tue Jul 13 12:08:33 BST 2021" version="6.9"/>

<DeviceFingerprints>

<DeviceFingerprint category="Server" family="VMWare" name="ESXi650">

<FingerprintRules>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="snmp.sys\_descr" operator="contains" value="VMware ESXi 6.5.0 build-13932383 VMware, Inc. x86\_64"/>

</FingerprintRule>

</FingerprintRules>

</DeviceFingerprint>

<DeviceFingerprint category="Switch" family="Aruba" name="Aruba Controller">

<FingerprintRules>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="host.ports" operator="contains\_all">

<valueList>17</valueList>

<valueList>21</valueList>

<valueList>22</valueList>

<valueList>80</valueList>

<valueList>443</valueList>

<valueList>1723</valueList>

Added in

<valueList>8080</valueList>

<valueList>8081</valueList>

<valueList>8082</valueList>

<valueList>8088</valueList>

</RuleCondition>

</FingerprintRule>

</FingerprintRules>

</DeviceFingerprint>

<DeviceFingerprint category="Home Audio/Video Equipment" family="Apple" name="Apple TV">

<FingerprintRules>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="host.ports" operator="contains\_all">

<valueList>3689</valueList>

<valueList>5000</valueList>

<valueList>7000</valueList>

<valueList>7100</valueList>

<valueList>62078</valueList>

</RuleCondition>

</FingerprintRule>

</FingerprintRules>

</DeviceFingerprint>

<DeviceFingerprint category="Home Audio/Video Equipment" family="TP-Link" name="TP-Link Webcam">

<FingerprintRules>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="dhcp.option60" operator="contains">

<valueList>udhcp 1.12.1</valueList>

</RuleCondition>

<RuleCondition name="dhcp.options" operator="contains">

<valueList>53,61,12,60,50,54,55</valueList>

</RuleCondition>

</FingerprintRule>

</FingerprintRules>

</DeviceFingerprint>

<DeviceFingerprint category="Computer" family="Raspberry Pi" name="Raspberry Pi">

<FingerprintRules>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="snmp.name" operator="contains\_all" value="raspberrypi"/>

<RuleCondition name="snmp.sys\_descr" operator="contains\_all" value="Linux raspberrypi 4.19.102-v7+ #1295 SMP Thu Feb 6 15:43:59 GMT 2020 armv7l"/>

</FingerprintRule>

<FingerprintRule match-conditions="ALL">

<RuleCondition name="dhcp.option55" operator="contains">

<valueList>1,121,33,3,6,12,15,26,28,42,51,54,58,59,119</valueList>

</RuleCondition>

<RuleCondition name="dhcp.option60" operator="contains">

<valueList>dhcpcd-6.11.5:Linux-4.19.102-v7+:armv7l:BCM2835</valueList>

</RuleCondition>

<RuleCondition name="dhcp.options" operator="contains">

<valueList>53,61,50,57,60,12,-111,55</valueList>

</RuleCondition>

<RuleCondition name="host.mac\_vendor" operator="contains">

<valueList>Raspberry Pi Foundation</valueList>

</RuleCondition>

</FingerprintRule>

</FingerprintRules>

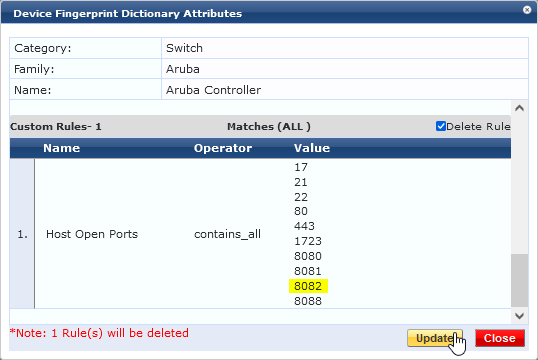
</DeviceFingerprint>

</DeviceFingerprints>

</TipsContents>

This can then be edited and re-imported back in.

Alternatively, you can delete the rule:

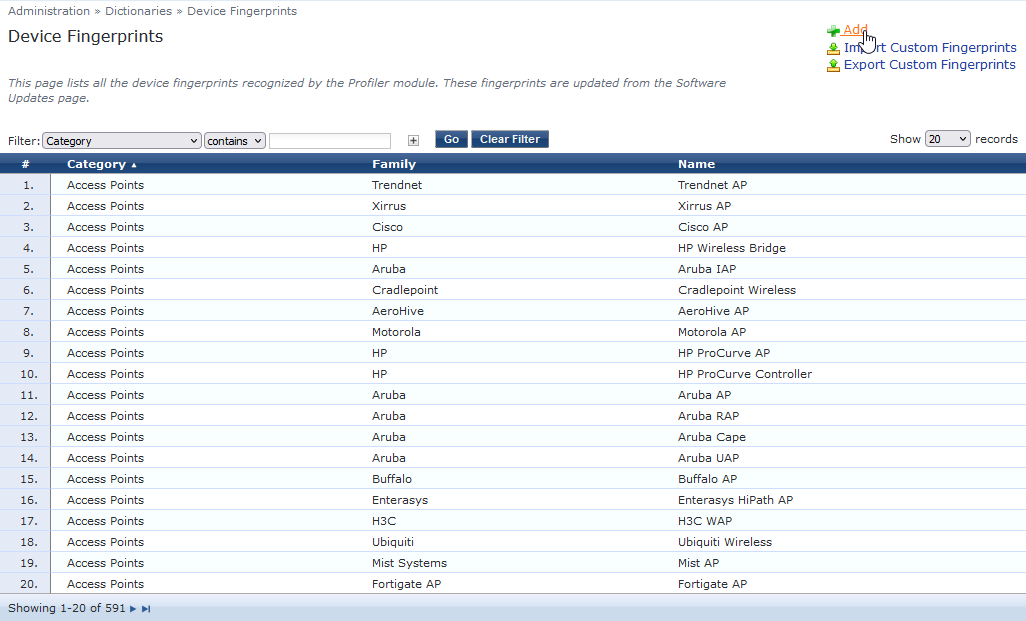
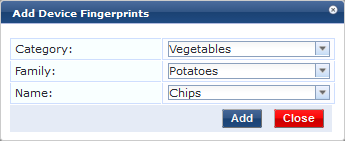


At this point the fingerprint is removed from the associated endpoints. You can then delete a custom fingerprint.

Irrespective, at this point you can re-add the rule…

## New Classification

Occasionally you need to define a new classification



Once the new classification has been added it can then be associated with an fingerprint.

# Automate

Please refer to https://www.statology.org/jaccard-similarity-python/