CX 10.8 Update
July/August 2021

aruba
a Hewlett Packard Enterprise company

UBT Enhancements & Device Fingerprinting

Justin Noonan
Technical Marketing Engineer

UBT Enhancements

Overview – User-Based Tunneling Enhancements

Role-Based Auto VLAN Creation

– Automates VLAN creation with user roles for VLAN Extend mode in UBT or locally-switched user roles (L2) or VSA assigned VLANs

– Eliminates the need for manually creating VLANs on every single switch at the access layer

– Simplifies switch configuration and deployment

– Only one command needed to enable

– Supported on 6200,6300,6400 platforms

Client IP Tracker - UBT Clients

– Tracks the IP addresses of client traffic connected to switch through user tunnels

– Assists operators with client visibility for tunneled clients

– Notes any IP address changes in the Client IP Table

– Sends an ARP/ND probe if no client traffic to validate existing address

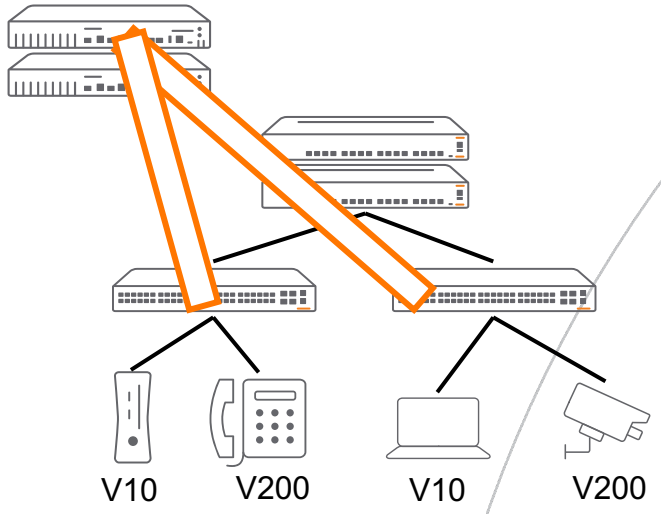
– Enabled at Switch, VLAN, or Port level

– Supported on 6200,6300,6400 platforms

Role-Based Auto VLAN

Role-Based Auto VLANs

- Scale based on maximum clients and VLANs available
  - Max Clients
    - 6400/6300 = 256 per port
    - 6200 = 32 per port
  - Max VLANs
    - 6400/6300 = 4094
    - 6200 = 2048
- Caveats
  - Mutually Exclusive with VSX, MVRP, and RPVST – must be disabled to enable auto VLANs
  - No SNMP Support
  - Not supported with the Reserved VLAN mode (local VLAN) of UBT



Role-Based Auto VLAN Configuration

– New command to allow Authentication-Based VLAN creation either by Role or VSA

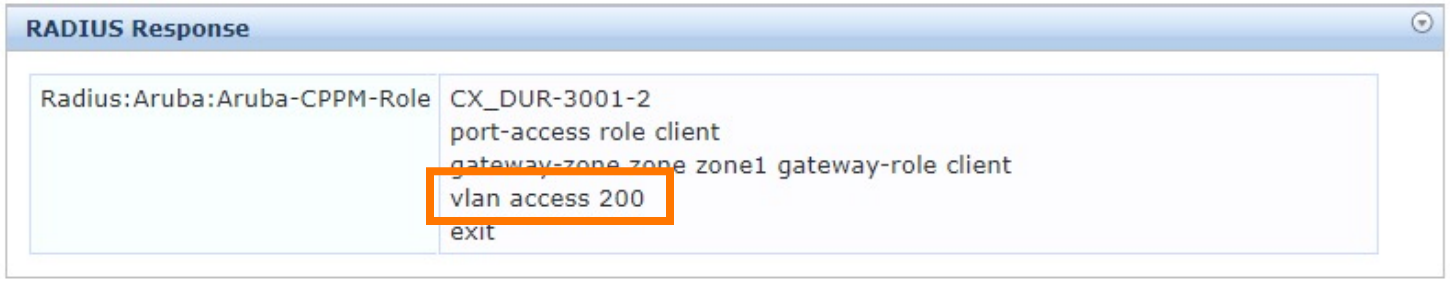
```
Switch(config)#port-access auto-vlan
```

– VLAN list shows VLANs assigned statically and by port-access

```
Switch(config)# show vlan

---------------------------------------------------------------------------------------------------------
VLAN  Name                                Status  Reason                Type         Interfaces
---------------------------------------------------------------------------------------------------------
1     DEFAULT_VLAN_1                      down    no_member_forwarding  default      1/1/2,1/1/4-1/1/28
100   VLAN100                             up      ok                    static       1/1/1
200   VLAN200                             up      ok                    port-access  1/1/3
```

– Configure the VLAN either in the downloadable role (ClearPass), Local Role, or VSA



*Note: UBT not required

Troubleshooting Role-Based Auto VLANs

– Ensure that the auto-vlan command is enabled

– debug portaccess will show any output dependent on the auto-vlan command

```
2021-07-30:09:44:00.604334|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Event handler of portclientauth with mac f4:30:b9:ce:7e:e6 on port '1/1/2' for event
'Auth-Profile Ready' in state 'FINAL AUTH SUCCESS' returned 'OK'
2021-07-30:09:44:00.604297|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handling event 'Auth-Profile Ready' for mac f4:30:b9:ce:7e:e6in port 1/1/2 in state
'FINAL AUTH SUCCESS'
2021-07-30:09:44:00.604255|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 portclientattr PUTN attribute status 0 for client'1/1/2, f4:30:b9:ce:7e:e6'
2021-07-30:09:44:00.604220|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 portclientattr VLAN attribute status 0 for client'1/1/2, f4:30:b9:ce:7e:e6'
2021-07-30:09:44:00.600586|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 3 to component Port Client Attribute
2021-07-30:09:44:00.600564|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Event handler of VLAN '200 for event 'Secure Client Add' in state 'AUTOACTIVE' returned
'Ok'
2021-07-30:09:44:00.600539|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handling event 'Secure Client Add' for VLAN '200 in state 'AUTOACTIVE'
2021-07-30:09:44:00.600505|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 3 to component Secure VLAN
2021-07-30:09:44:00.600406|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 0 to component Port Attribute
2021-07-30:09:44:00.600343|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 1 to component Port Client Attribute
2021-07-30:09:44:00.600270|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Event handler of VLAN '200 for event 'Secure Client Add' in state 'NULL' returned 'Ok'
2021-07-30:09:44:00.600217|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 securevlan SM State transition [NULL] -> [AUTOACTIVE] for object with key '200
2021-07-30:09:44:00.600157|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handling event 'Secure Client Add' for VLAN '200 in state 'NULL'
2021-07-30:09:44:00.600103|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 3 to component Secure VLAN
2021-07-30:09:44:00.600015|port-accessd|LOG_DEBUG|MSTR|1|PORTACCESS|PORTACCESS_SERVICES|logID=33151 Handing over the event 0 to component Port Client Attribute
```

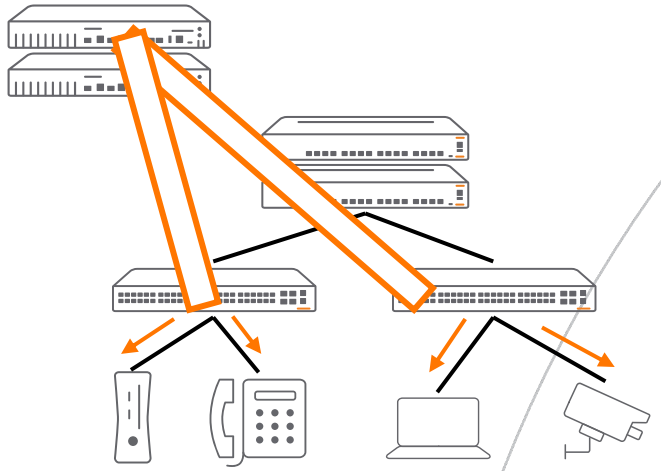
– Look for the “Secure client add” messages with the VLAN ID that is returned as “ok”

– Final Auth Success message indicates that the client was successfully placed into the automatically created VLAN

Client IP Tracker for User-Based Tunneling

UBT Client IP Tracker

- Scale based on maximum clients and VLANs available
  - Limited to Max UBT Clients
    - 6400/6300/6200/4100i = 1017 per switch/stack
- Key Considerations
  - Will work on an MC-LAG
  - Can be configured on LAG interfaces
    - Client MAC Address will appear as learned on a specific physical LAG port
    - Works the same as an access port
  - Not supported on a VSX ISL
  - Will not learn clients from a VSX peer – only local
  - Not supported on routed-only ports
  - Not supported on SVI ports



UBT Client IP Tracker - Configuration

– Configurable via Globally, Port, or VLAN

```
Switch(config)#client track ip
interface 1/1/2
    client track ip enable
    client track ip update-interval 60
```

– Default update interval is 1800 seconds – recommended setting
  – Can be lowered to a minimum of 60s (range is 60-28000)
  – Use caution as this will generate multiple probes and additional traffic

– “show client ip” lists the Client IP and MAC addresses that are being tracked

– When an IP is changed, the list will update, which could result in multiple entries for the same MAC

```
6300-UI(config-if)# show client ip

 MAC Address               Interface        VLAN             IP Address
 ------------------------------------------------------------------------------
 f4:30:b9:ce:7e:e6         1/1/2            200              10.5.6.60
 f4:30:b9:ce:7e:e6         1/1/2            200              10.5.6.65
```

UBT Client IP Tracker - Configuration

– At the port level, client IP tracker is set to auto, which is the default mode and should be best practice

```
Switch(config-if)# client track ip
  auto             Default mode. Track limited set of clients based on LLDP/CDP
                   signature and access/trunk configuration
  client-limit     Configure the maximum number of clients tracked on this port
  disable          Client IP addresses are not tracked on this port
  enable           Track all client IP addresses on this port
  update-interval  Configure the IP address update interval in seconds
  <cr>
```

– The feature can also be enabled and disabled, as well as set the client limit for the number of clients that will be tracked

– The update interval can also be adjusted in case of troubleshooting and need to probe clients more frequently

UBT Client IP Tracker - Configuration

– When the update interval timer is expired, the switch will send out an ARP or ND probe to probe the clients for any IP address changes

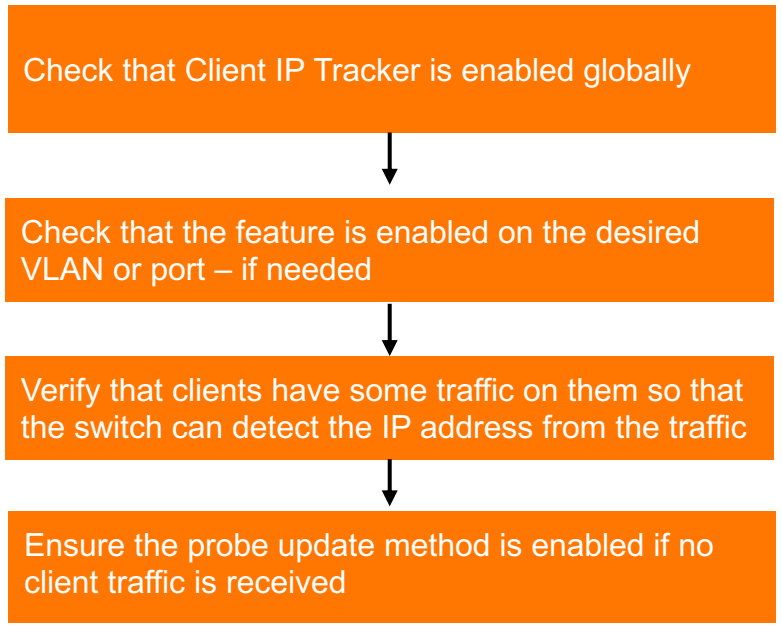
```
> Frame 211: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{1F18785D-D3C5-4C46-A983-B28BAA55AA87}, id 0
> Ethernet II, Src: ArubaaHe_92:d4:c0 (88:3a:30:92:d4:c0), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
v Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: ArubaaHe_92:d4:c0 (88:3a:30:92:d4:c0)
    Sender IP address: 0.0.0.0
    Target MAC address: Broadcast (ff:ff:ff:ff:ff:ff)
    Target IP address: 10.5.6.60
```

– The client will then respond back with the new address which will then be updated in the Client IP table

```
> Frame 261: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface \Device\NPF_{1F18785D-D3C5-4C46-A983-B28BAA55AA87}, id 0
> Ethernet II, Src: HewlettP_ce:7e:e6 (f4:30:b9:ce:7e:e6), Dst: ArubaaHe_92:d4:c0 (88:3a:30:92:d4:c0)
v Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: HewlettP_ce:7e:e6 (f4:30:b9:ce:7e:e6)
    Sender IP address: 10.5.6.65
    Target MAC address: ArubaaHe_92:d4:c0 (88:3a:30:92:d4:c0)
    Target IP address: 0.0.0.0
```

UBT Client IP Tracker - Troubleshooting

- Troubleshooting flow



Demos

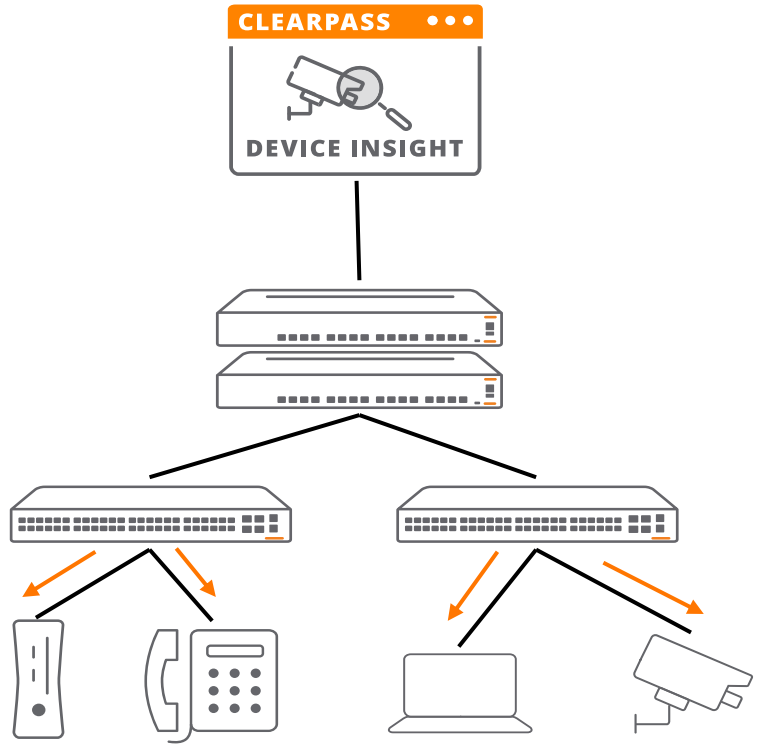
Device Fingerprinting

Device Fingerprinting

- Device Fingerprinting is achieved by configuring a switch to collect the traffic patterns and send only required information or attributes to an analyzer.
- The switch (Collector) collects the protocol data sent by the end clients and the Analyzer consumes this data to fingerprint the device.
- This solution consists of two parts:
  - **Collector:** Collects the information from the packets that are obtained from the clients. The information collected is based on the certain type of traffic from the clients. Aruba CX switches will act as collectors.
    - Current supported protocols are: DHCP, HTTP, LLDP and CDP.
  - **Analyzer :** Is an entity that processes the information collected from the collector and fingerprints the device with the details of the device such as type, host name, vendor type, etc. An example of an analyzer can be Central (using CPDI) or ClearPass.
- Previously, device fingerprinting could be found on the AOS-Switch product line

Device Fingerprinting

- Device Fingerprinting allows a network operator to have better visibility into what types of devices are plugging into the access layer
- By obtaining as much information as possible from a client, more granular security policies can be created and enforced to maintain client and network stability
- This is the first phase of introducing device fingerprinting in AOS-CX
  - Central support coming in 2.5.4 (client hostname visibility)
  - CPDI support coming in future
  - ClearPass support coming in future
- Supported on 6200,6300,6400 platforms
- Scale
  - 6300/6200 = 2000 clients
  - 6400 = 4000 clients
  - Max number of DFP profiles = 32 (all supported platforms)



Note: The only analyzer that will currently be supported is Central 2.5.4 which can receive the hostname attribute from the switch fingerprint attributes

Device Fingerprinting - Configuration

– Create Device Fingerprinting profile – Select Protocol – Apply to interface

```
Switch(config)#client device-fingerprint profile DFP
Switch(config-device-fingerprint)#
  cdp    Specify the CDP attributes for device fingerprinting
  dhcp   Specify the DHCP attributes for device fingerprinting
  end    End current mode and change to enable mode.
  exit   Exit current mode and change to previous mode
  http   Specify the HTTP attributes for device fingerprinting
  list   Print command list
  lldp   Specify the LLDP attributes for device fingerprinting
  no     Negate a command or set its defaults
  show   Show running system information
Switch(config)# client device-fingerprint profile DFP
    lldp tlv-num 5
    dhcp
    http user-agent
Switch(config-if)# client device-fingerprint
  apply-profile  Apply profile on a port/portlist
  client-limit   Specify client-limit on a port/portlist
```

– Protocols supported are:
  – CDP
  – LLDP
  – DHCP
  – HTTP

Device Fingerprinting - Configuration

Supported LLDP Parameters

- chassis-id (1)
- port-id (2)
- time-to-live (3)
- port-description (4)
- system-name (5)
- system-description (6)
- system-capabilities (7)
- management-address (8)

Supported CDP Parameters

- chassis-id (1)
- port-id (2)
- address (3)
- capabilities (4)
- version (5)
- platform (6)
- native-vlan (10)
- duplex (11)

Supported HTTP Parameters

– HTTP – User Agent

Supported DHCP Parameters

– Option 12 : Hostname - This provides the information about the name of the client.

– Option 55 : Parameter Requested List

– Option 60 : Vendor Class Identifier (VCI)

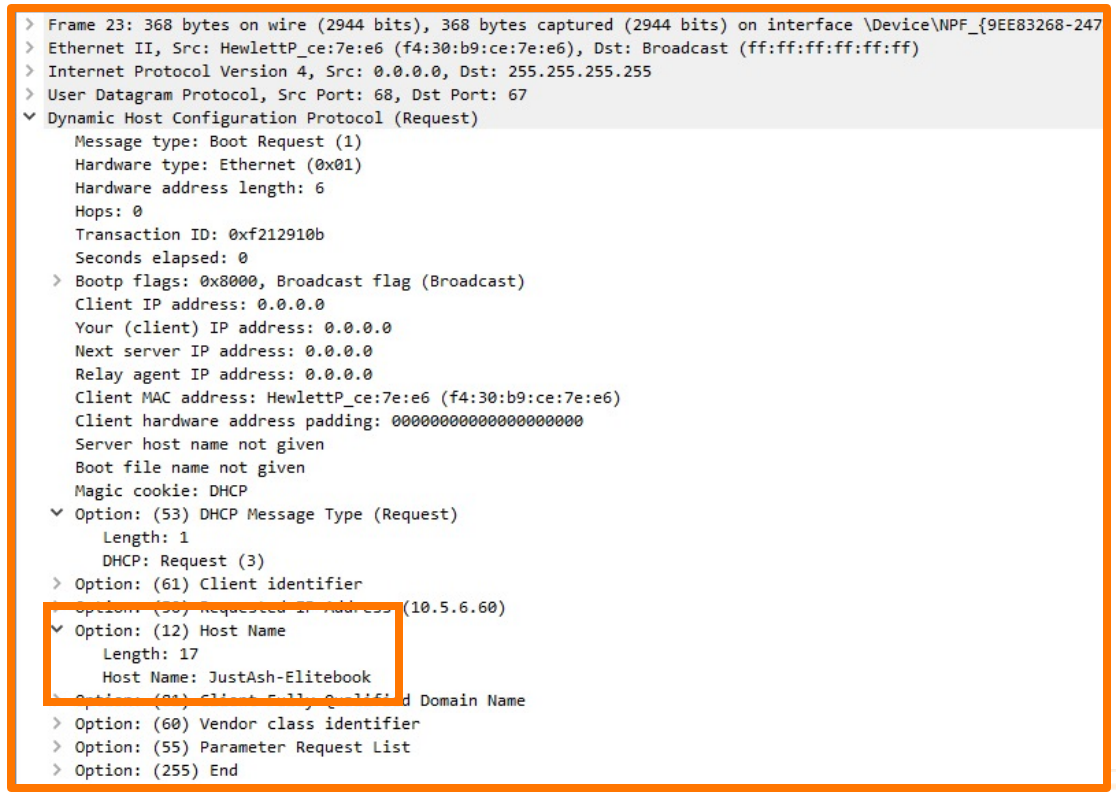
The combination of the option sequence in option 55 or vendor ID (60) is used to infer the OS and device type of the remote client

Device Fingerprinting - Validation

– Show client device-fingerprint

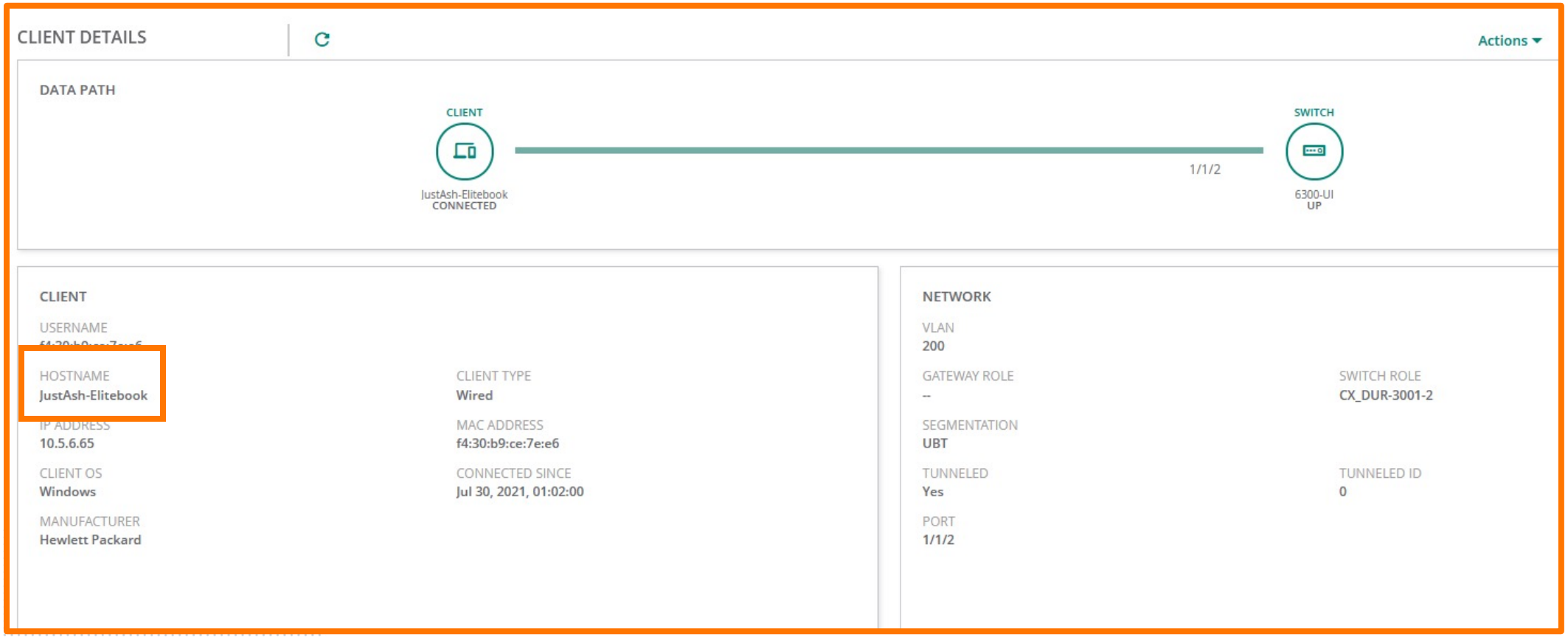
```
6300-UI(config)# show client device-fingerprint
Client MAC Address: f4:30:b9:ce:7e:e6
Port     : 1/1/2
VLAN     : 200
Protocol: DHCP
   Host name(12)                 :JustAsh-Elitebook
   Parameter Requested List(55)  :1,3,6,15,31,33,43,44,46,47,119,121,249,252
   Vendor Class Identifier(60)   :MSFT 5.0
Protocol: HTTP
   --
Protocol: LLDP
   System-Description(6)         :
   System-Name(5)                :
Protocol: CDP
   n/a

Client MAC Address: f8:60:f0:c8:e2:00
Port     : 1/1/3
VLAN     : 200
Protocol: DHCP
   Parameter Requested List(55)  :1,3,4,23,67,66,43,6,15,119,42,2,60,138
   Vendor Class Identifier(60)   :Aruba JL693A 2930F-12G-PoE+-2G-2SFP+ Switch dslforum.org
Protocol: HTTP
   --
Protocol: LLDP
   --
Protocol: CDP
   n/a
```



Device Fingerprinting - Validation

– Central 2.5.4 (~Q4CY21)



Device Fingerprinting Troubleshooting

– debug devicefingerprint

```
2021-07-30:11:23:50.378514|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT f4:30:b9:ce:7e:e6 - DHCP {60, MSFT 5.0}
2021-07-30:11:23:50.378494|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT f4:30:b9:ce:7e:e6 - DHCP {55,
1,3,6,15,31,33,43,44,46,47,119,121,249,252}
2021-07-30:11:23:50.378475|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT f4:30:b9:ce:7e:e6 - DHCP {12, JustAsh-Elitebook}
2021-07-30:11:23:50.378447|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT f4:30:b9:ce:7e:e6 - LLDP {ebf13c42-cc14-49ef-aec6-
80800fe629c6}
2021-07-30:11:23:50.378374|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_PACKET|DHCP f4:30:b9:ce:7e:e6 - 2 Option Num : 60 Data : MSFT 5.0
2021-07-30:11:23:50.378352|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_PACKET|DHCP f4:30:b9:ce:7e:e6 - 2 Option Num : 55 Data :
1,3,6,15,31,33,43,44,46,47,119,121,249,252
2021-07-30:11:23:50.378319|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_PACKET|DHCP f4:30:b9:ce:7e:e6 - 2 Option Num : 12 Data : JustAsh-Elitebook
2021-07-30:11:23:49.785173|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT f4:30:b9:ce:7e:e6 - LLDP {ebf13c42-cc14-49ef-aec6-
80800fe629c6}
2021-07-30:11:23:49.785146|dfpd|LOG_INFO|MSTR|1|DEVICEFINGERPRINT|DEVICEFINGERPRINT_CONFIG|OVSDB CLIENT {f4:30:b9:ce:7e:e6 - 2, 1/1/2} INSERTED
```

– Can see actual attributes gained from dhcp options

– Ensure profile is created and enabled

– Ensure DFP is enabled on the port

– Ensure the appropriate protocol is being transmitted by fingerprinted device (Ex. LLDP turned on, DHCP enabled, etc.)

Demo

aruba
a Hewlett Packard Enterprise company

Thank you

justin.noonan@hpe.com