Cambium Networks™

Troubleshooting Tools in cnPilot E-Series APs

Cambium Networks

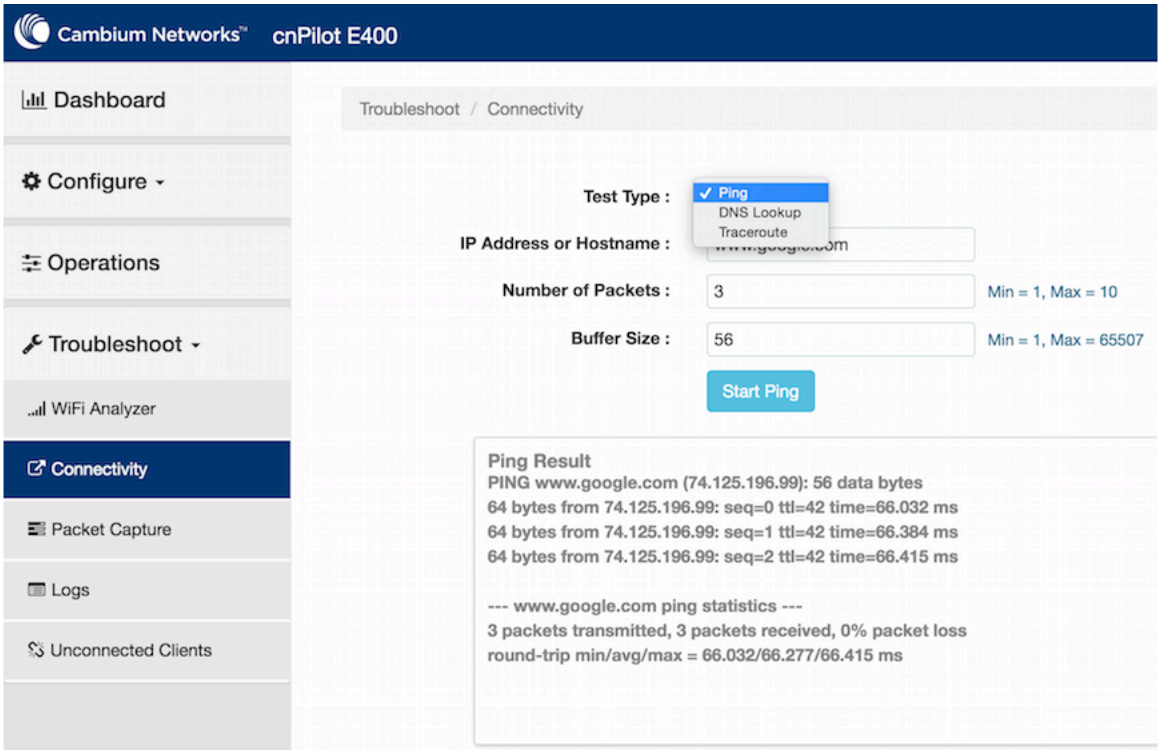
Table of Contents

1. Introduction

The cnPilot E-series WiFi Access Points from Cambium Networks include several troubleshooting tools for administrators to gain more detailed insights into the behavior of the AP as well as the clients and the network it is connected to. This document describes these tools and their usage.

2. Network Connectivity Tools

cnPilot APs include ping, traceroute and DNS lookup utilities, which can be invoked from the device GUI or the CLI. In the GUI the tools are under '***Troubleshooting->Connectivity***':



Test Type : Ping

IP Address or Hostname : www.google.com

Number of Packets : 3 Min = 1, Max = 10

Buffer Size : 56 Min = 1, Max = 65507

```
Ping Result
PING www.google.com (74.125.196.99): 56 data bytes
64 bytes from 74.125.196.99: seq=0 ttl=42 time=66.032 ms
64 bytes from 74.125.196.99: seq=1 ttl=42 time=66.384 ms
64 bytes from 74.125.196.99: seq=2 ttl=42 time=66.415 ms

--- www.google.com ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 66.032/66.277/66.415 ms
```

These are also accessible from the CLI of the device:

```
cnWest-E400(config)# ping www.google.com
PING www.google.com (74.125.196.106): 56 data bytes
64 bytes from 74.125.196.106: seq=0 ttl=42 time=69.369 ms
64 bytes from 74.125.196.106: seq=1 ttl=42 time=72.623 ms
^C
--- www.google.com ping statistics ---
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max = 69.369/70.996/72.623 ms
cnWest-E400(config)#
cnWest-E400(config)#
cnWest-E400(config)# traceroute www.google.com
```

```
traceroute to www.google.com (74.125.196.147), 30 hops max, 38 byte packets
 1  10.140.134.254 (10.140.134.254)  0.696 ms  0.734 ms  1.384 ms
 2  *  *  *
 3  *  *  *
```

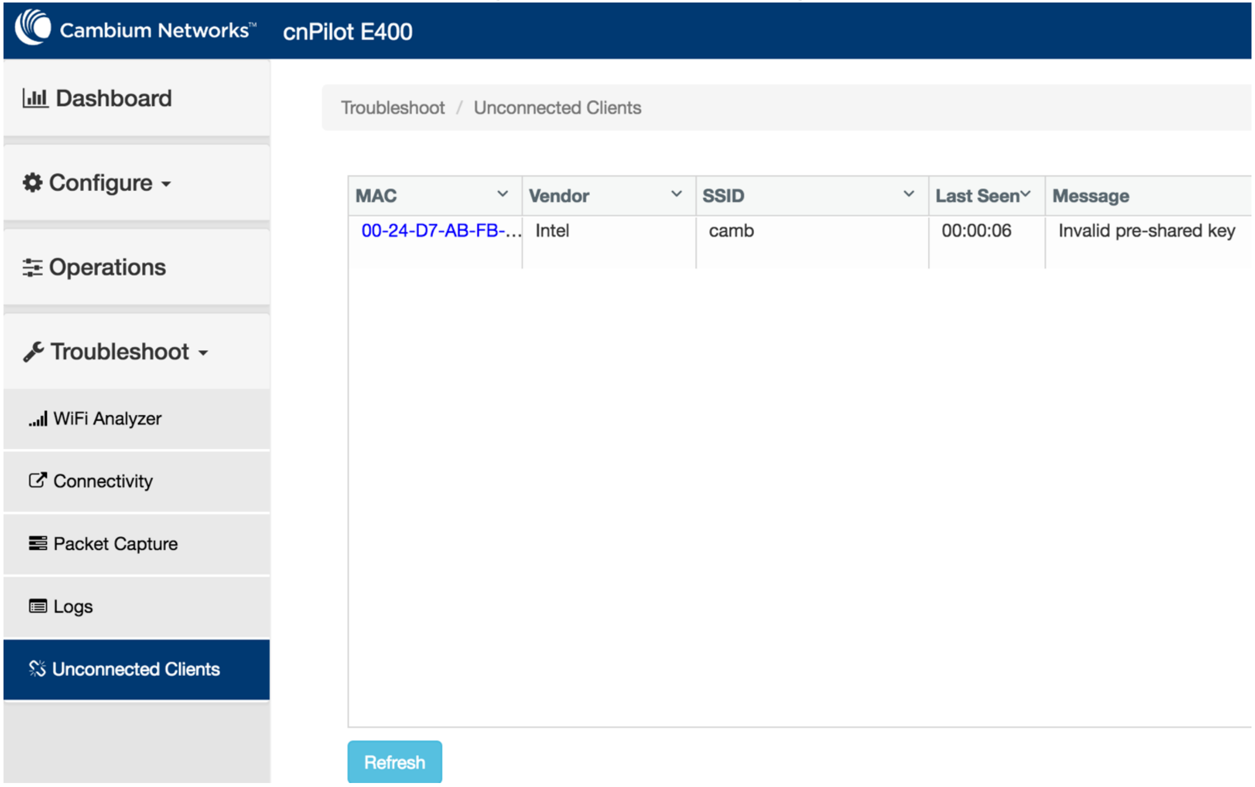
3. Client Connectivity

The AP dashboard displays a list of currently connected clients as well as their key parameters such as the IP address, the radio and WLAN they are on, the amount of traffic from and to the client. This information is also available in the CLI using the **show wireless client** and associated commands.

However one problem that still remains is clients that tried to connect, but were unsuccessful for some reason. These clients will not show up in the client table and it may not be obvious to the administrator why they are not. Cambiums cnPilot Enterprise Access Points solve this by also tracking such clients and showing them in a separate Unconnected Clients table. Clients might fail to connect due to various reasons such as:

- A MAC ACL denying access
- Failing authentication on a WPA2-Enterprise wireless LAN
- Mis-Matched passphrase on a WPA2-Personal wireless LAN

To view these clients, on the GUI navigate to **Troubleshooting->Unconnected Clients**:



MAC	Vendor	SSID	Last Seen	Message
00-24-D7-AB-FB-...	Intel	camb	00:00:06	Invalid pre-shared key

This information is also available in the CLI as **show wireless clients unconnected** as well as through cnMaestro's troubleshooting page for WiFi APs.

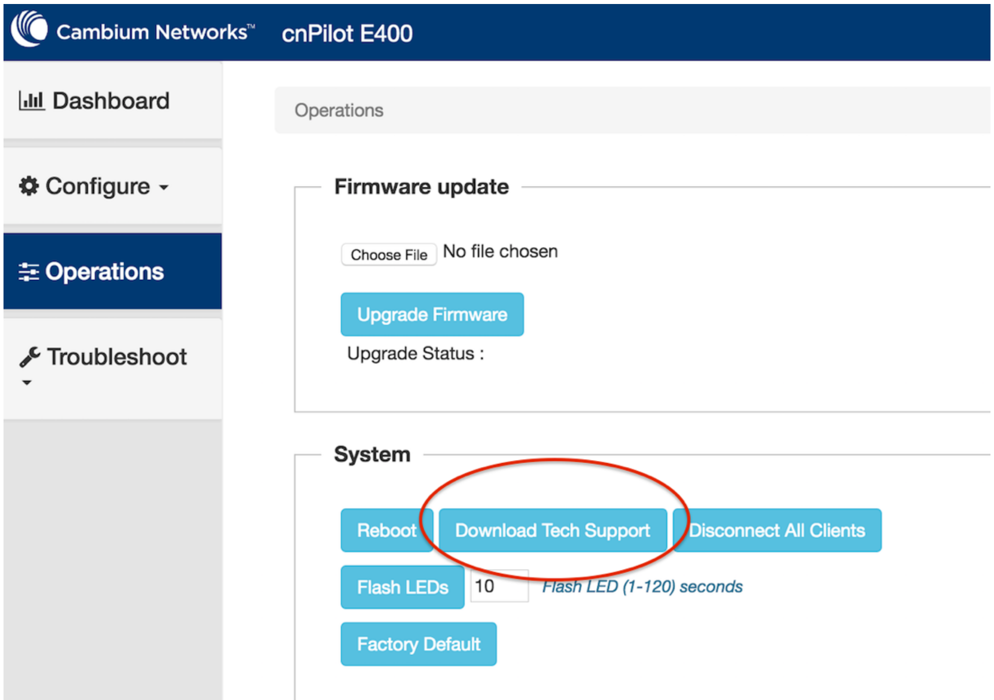
4. Tech-Support

All the debugging information needed from an Access Point to troubleshoot something is bundled together in a single file by the AP, and accessible through a single click in the GUI. This avoids having to do multiple commands one by one to collect all the information we might need to troubleshoot an issue.

The tech-support file is a single zipped up file which contains:

- The system configuration file
- Debug logfiles of all the daemons/processes
- Contents of /proc related to the radio drivers datapath
- Output of a number of CLI commands ('show wireless client', 'show version' etc)
- Output of a number of Linux shell commands ('ifconfig', 'free', 'top', 'ps')
- System startup logs as reported by 'dmesg'.
- Any process core-files and crashlogs

All of this information can be downloaded at one go using the tech-support:



Techsupport can also be downloaded from the CLI directly onto a TFTP or FTP server:

```
cnWest-E400(config)# export tech-support tftp://1.2.3.4/tech.tar.gz
cnWest-E400(config)#
```

5. Packet Capture Options

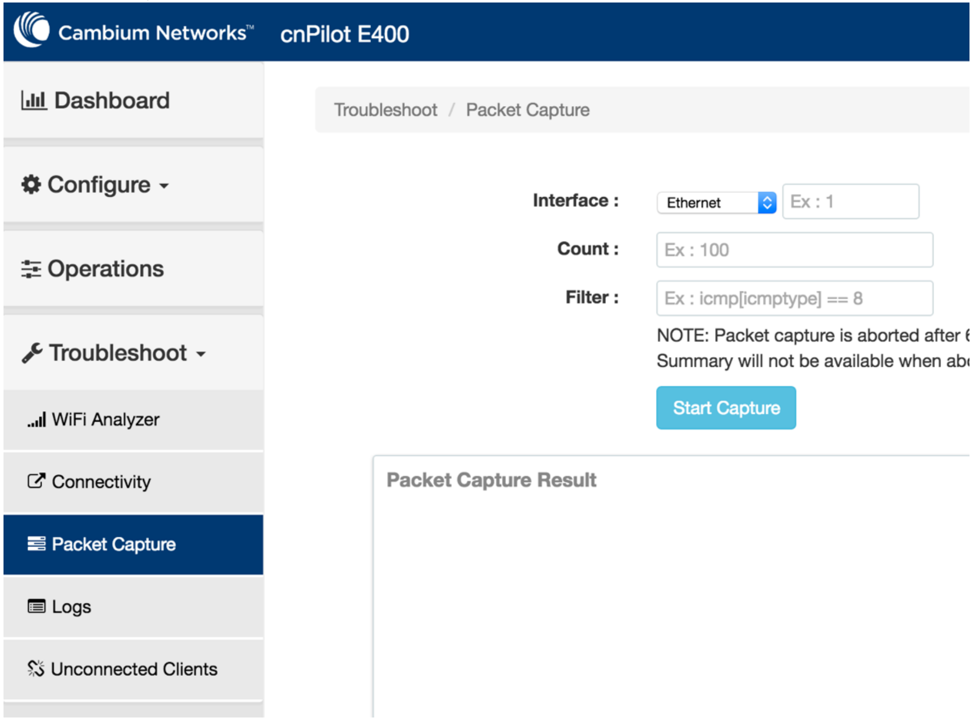
cnPilot APs include a facility to capture packets for analysis. In the current software release this analysis is done inline, going forward options will be added to save packets and export them for offline analysis using tools such as Wireshark.

Packet capture can be invoked from the CLI or the GUI. There are three ‘capture-points’:

- Eth – captures packets on the Ethernet port of the AP
- Vlan – captures packets on a specific SVI (L3 vlan interface) of the AP
- Wlan – capture packets on a specific BSSID/WLAN.

Captures at **eth** are useful to see every packet ingressing or egressing the AP. Captures on **VLAN** are useful for debugging issues with say DHCP when the onboard DHCP server is used. Captures at the **WLAN** level are useful to view every data packet that is ingressing or egressing the radio on that BSSID. Note that the WLAN data capture only intercepts data packets, so the packet format is 802.3 not 802.11. Also, any management and control frames will not be seen as they are processed at the radio itself.

Packet capture can be invoked from the GUI or the CLI. In the GUI it is under *Troubleshoot->Packet Capture:*



Interface : Ethernet Ex : 1

Count : Ex : 100

Filter : Ex : icmp[icmptype] == 8

NOTE: Packet capture is aborted after 6
Summary will not be available when abo

In addition to specifying the interface, the administrator can also specify a packet count for the capture, as well as a filter to limit the capture to only the interesting packets.

The capture filter options are same as what *tcpdump* supports, since the packet capture commands use tcpdump on their backend. An example of all the filter options available can be seen in the CLI:

```
cnWest-E400(config)# packet-capture vlan 1

  Specify vlan (1-4094) followed optionally by a filter. e.g.:
     Format: <protocol>   <direction>   <type>
                          src or dst    [host], net, port
     arp
     icmp
     ether dst  <aa:bb:cc:dd:ee:ff>  # Matches packet's destination MAC address
     ether src  <aa:bb:cc:dd:ee:ff>  # Matches packet's source MAC address
     [ip] dst  <a.b.c.d>  # Matches packet's destination IP address
     [ip] src  <a.b.c.d>  # Matches packet's src IP address

     tcp dst port X  # Matches packet's tcp destination port
     tcp src port X  # Matches packet's tcp source port

     udp dst port X  # Matches packet's udp destination port
     udp src port X  # Matches packet's udp source port

     dst port X # Matches packet's tcp or udp destination port
     src port X # Matches packet's tcp or udp source port


cnWest-E400(config)# packet-capture wlan 1 tcp
    1  00:00:00.000000 00:26:ca:d6:82:4a > 00:00:00:00:00:00, IPv4, length 133: (tos
0x20, ttl 38, id 17496, offset 0, flags [DF], proto TCP (6), length 119)
    54.86.61.141.443 > 10.140.134.5.54165: tcp 67
    2  00:00:01.247395 00:26:ca:d6:82:4a > 00:00:00:00:00:00, IPv4, length 121: (tos
0x20, ttl 38, id 12961, offset 0, flags [DF], proto TCP (6), length 107)
    54.86.61.141.443 > 10.140.134.5.40563: tcp 67
^C
2 packets captured
2 packets received by filter
0 packets dropped by kernel
cnWest-E400(config)#
```

So for instance to capture all RADIUS packets, which would usually go on the standard port 1812 the filter used would be **dst port 1812**. Filters can also be combined into one using **and** and **or**. For instance '**dst port 1812 or src port 1812'**. Filters can also be negated with a **not** prefix.

Firmware version 2.5.1 includes a new option called *binary* which makes the packet-capture command on the CLI dump out the contents in binary format. This is useful if the user wishes to redirect the output to say Wireshark for offline analysis. After SSH'ing into the AP, packet-capture

can be invoked with the binary option, and all the output redirected to a wireshark instance (configured to read off STDIN) on a PC.
For example:

```
sshpass -p admin ssh –o StrictHostKeyChecking=no admin@192.168.0.80 'packet-
capture binary eth 1’ | wireshark -k -i -
```

6. View System Events

There are two levels of logs maintained on the AP:

- Events
- Debug-Logs

An **event** that captures any significant occurance on the AP such as a client associating, or the system being rebooted etc. Events are processed in the following manner:

- The last 256 events are maintained in the APs buffer and accessible using **show events**
- Every event is forwarded up to cnMaestro for display in its Event/Alarm table
- Every event is translated into an SNMP trap for any configured trap receivers.
- Every event is forwarded as a Syslog to any configured Syslog receiver.

Example output of **show events:**

```
May 23 21:15:20 WIFI-6-CLIENT-CONNECTED Client [00-24-D7-AB-FB-3C] connected to wireless lan [camb]
May 23 21:15:20 WIFI-6-CLIENT-DISCONNECT-INFO Client [00-24-D7-AB-FB-3C] disconnect-info [ap-sent-
disassoc-with-code-2]
May 23 21:15:20 WIFI-6-CLIENT-DISCONNECTED Client [00-24-D7-AB-FB-3C] disconnected from wireless lan
[camb]
May 23 21:15:19 WIFI-6-CLIENT-WPA2-INVALID-PSK Client [00-24-D7-AB-FB-3C] failed handshake on wireless lan
[camb] due to invalid pre-shared key
```

Events follow the syslog formatting model of a timestamp prefix, followed by a mnemonic (capitalized) which includes a short-anem of the event, the module that this event refers to, and a severity level from 1..7, following standard Syslog concents (1 being highest severity, 7 being informational/debug messages).

7. View Process Debug log messages

Debug logs are meant primarily for troubleshooting and are free-form logs (not formatted the events are) from different modules in the system. Each module maintains its own debug logs, the primary modules on the AP include:

- Scmd – responsible for stats and configuration
- Infrad – responsible for infrastructure components like DHCP server, Ethernet port config
- Wifid – responsible for radio-configuration, client-association and handshake handling
- Sysmod – responsible for pinging all Cambium processes that have requested monitoring and restart any that appear down.

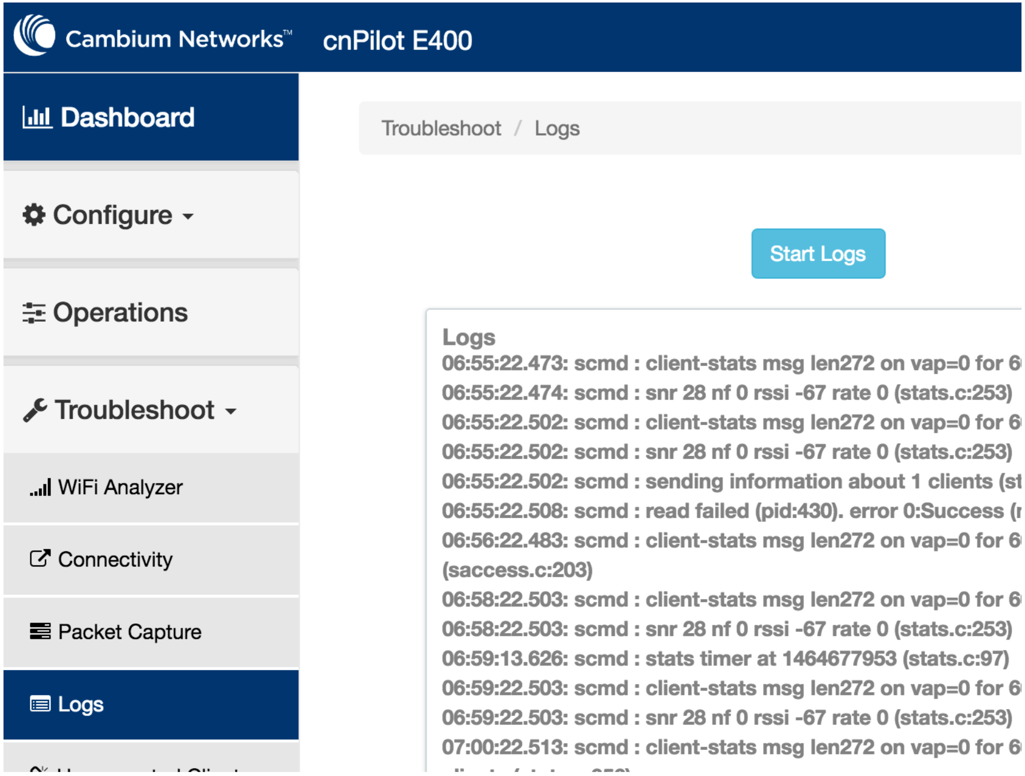
All logs go under /var/log/DAEMON-NAME.log and are rotated out once they are 100KB large. To view the logs from the CLI:

```
cnWest-E400(config)# service show debug-logs wifid|more
07:05:42.346: GA update rcvd for client [00-24-D7-71-19-94] on ssid Hotel WLAN2, vlan 1,
session time 600 (cache.c:386)
07:05:42.346: notify msg type CMB_NOTIFY_MSG_TYPE_GA_UPDATE[2] received (cache.c:544)
07:05:42.345: tx sync message type 0 of 348 bytes (main.c:136)
07:04:42.161: client[00-24-D7-71-19-94] not added, ssid [deepguest] not on this device
(cache.c:568)
```

Logs can be very large, and as shown above CLI output qualifiers can be used to trim down to what the administrator might be interested in. When piped through **more** (like the example above) the command output will be paginated for easy reading. When piped through **grep** the command output will be filtered out by whatever regular expression or string was passed to it. For example:

```
cnWest-E400(config)# service show debug-logs wifid | grep 00-24-D7-71-19-94
07:05:42.346: GA update rcvd for client [00-24-D7-71-19-94] on ssid Hotel WLAN2, vlan 1,
session time 600 (cache.c:386)
07:04:42.161: client[00-24-D7-71-19-94] not added, ssid [deepguest] not on this device
(cache.c:568)
07:04:38.035:  client[00-24-D7-71-19-94]  not  added,  ssid  [deepguest]  not  on  this  device
(cache.c:568)
```

Debug logs such as these can also be seen from cnMaestro or from the device's GUI interface:



```
Logs
06:55:22.473: scmd : client-stats msg len272 on vap=0 for 6
06:55:22.474: scmd : snr 28 nf 0 rssi -67 rate 0 (stats.c:253)
06:55:22.502: scmd : client-stats msg len272 on vap=0 for 6
06:55:22.502: scmd : snr 28 nf 0 rssi -67 rate 0 (stats.c:253)
06:55:22.502: scmd : sending information about 1 clients (st
06:55:22.508: scmd : read failed (pid:430). error 0:Success (r
06:56:22.483: scmd : client-stats msg len272 on vap=0 for 6
(saccess.c:203)
06:58:22.503: scmd : client-stats msg len272 on vap=0 for 6
06:58:22.503: scmd : snr 28 nf 0 rssi -67 rate 0 (stats.c:253)
06:59:13.626: scmd : stats timer at 1464677953 (stats.c:97)
06:59:22.503: scmd : client-stats msg len272 on vap=0 for 6
06:59:22.503: scmd : snr 28 nf 0 rssi -67 rate 0 (stats.c:253)
07:00:22.513: scmd : client-stats msg len272 on vap=0 for 6
```

8. View AP Operating System level status

There are also several commands on the CLI under the **service** prefix which can be used to view the state of the system by invoking differnet operation system level commands.

Service show memory : to view the state of allocated and available memory on the AP. For example:

```
cnWest-E400(config)# service show memory
             total         used         free       shared      buffers
Mem:        126208        71176        55032            0            0
-/+ buffers:              71176        55032
Swap:            0            0            0
cnWest-E400(config)#
```

Service show top : to view the state of every running process and a snapshot of the amount of CPU and memory it is using:

```
CPU:   9% usr   0% sys   0% nic  90% idle   0% io   0% irq   0% sirq
Load average: 0.00 0.01 0.05 1/39 5550
  PID  PPID USER     STAT   VSZ %VSZ %CPU COMMAND
 5550  5517 root     R     1504   1%   9% top -n1
  984   962 root     S     5580   4%   0% /usr/sbin/lighttpd -D -f /etc/lighttpd/lighttpd.conf
  985   962 root     S     5444   4%   0% /usr/sbin/lighttpd -D -f /etc/lighttpd/lighttpd.conf
  433     1 root     S     4604   4%   0% /usr/sbin/snmpd -f -c /var/run/etc/snmp/snmpd.conf -
p /tmp/snmpd.pid
  432     1 root     S     4588   4%   0% /usr/sbin/wifid -d
29396     1 root     S     4456   4%   0% /usr/sbin/device-agent -d2 -
shttps://cloud.cambiumnetworks.com
  430     1 root     S     4404   3%   0% /usr/sbin/scmd -d
  962     1 root     S     3904   3%   0% /usr/sbin/lighttpd -D -f /etc/lighttpd/lighttpd.conf
 5517   441 root     S     3876   3%   0% -clish
 1140     1 root     S     3464   3%   0% /usr/bin/stunnel /var/run/etc/stunnel/stunnel.conf
  431     1 root     S     1620   1%   0% /usr/sbin/infrad -d
  441     1 root     S     1508   1%   0% /usr/sbin/telnetd -F
    1     0 root     S     1504   1%   0% init
  515     1 root     S     1504   1%   0% /usr/sbin/ntpd -n -p pool.ntp.org
  436     1 root     S     1504   1%   0% /sbin/getty -L ttyS0 115200 vt100
  320     1 root     S     1500   1%   0% /sbin/klogd -n
```

service show ps : to view the PIDs of every running process on the AP

```
cnWest-E400(config)# service show ps
PID   USER     COMMAND
    1 root     init
    2 root     [kthreadd]
    3 root     [ksoftirqd/0]
    5 root     [kworker/0:0H]
    7 root     [watchdog/0]
    8 root     [khelper]
    9 root     [kdevtmpfs]
   10 root     [kworker/u2:1]
   93 root     [writeback]
   96 root     [bioset]
   98 root     [kblockd]
  134 root     [kswapd0]
  179 root     [fsnotify_mark]
  190 root     [ath79-spi]
  286 root     [kpsmoused]
  301 root     [deferwq]
  309 root     [yaffs-bg-1]
  320 root     /sbin/klogd -n
  321 root     /sbin/syslogd -n
  355 root     /usr/sbin/rngd -r /dev/urandom
  429 root     /usr/sbin/sysmond -d
  430 root     /usr/sbin/scmd -d
  431 root     /usr/sbin/infrad -d
```

service show route : to view the route table on the AP. This will include routes learnt over DHCP (Eg: a default route learnt from DHCP options) as well as routes configured by the user.

```
cnWest-E400(config)# service show route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         10.140.134.254  0.0.0.0         UG    0      0        0 br0.1
10.140.134.0    0.0.0.0         255.255.255.0   U     0      0        0 br0.1
169.254.0.0     0.0.0.0         255.255.0.0     U     0      0        0 br0.1
cnWest-E400(config)#
```

service show netstat : information on every socket open on the AP. This includes connections out to other servers from the AP, as well as all the ports on which the AP itself is either listening for connections or has a client now connected.

```
cnWest-E400(config)# service show netstat
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
tcp        0      0 127.0.1.1:8888          0.0.0.0:*               LISTEN
```

service show ifconfig : information about all the networking interfaces on the AP, their state, their MAC address, the number of bytes tx and rx from each

```
cnWest-E400(config)# service show ifconfig
br0       Link encap:Ethernet  HWaddr 00:04:56:F8:33:90
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5799458 errors:0 dropped:44 overruns:0 frame:0
          TX packets:474383 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:447833423 (427.0 MiB)  TX bytes:88194328 (84.1 MiB)

br0.1     Link encap:Ethernet  HWaddr 00:04:56:F8:33:90
          inet addr:10.140.134.154  Bcast:10.140.134.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:5799414 errors:0 dropped:3688 overruns:0 frame:0
          TX packets:474377 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:425410367 (405.7 MiB)  TX bytes:88193860 (84.1 MiB)
```

The *br* interfaces in ifconfig are bridged interfaces. *Br0* is the default bridge and for every L3 vlan interface X a new interface called *br0.X* is created. In the example above br0.1 is the L3 vlan SVI for VLAN1 and the IP address learnt over that VLAN was assigned to it here.

The *eth* interfaces refer to the wired etherent ports of the AP. Depending on the AP model there might be one or more of these.

The *wlan* interface refer to the BSSIDs of the AP. Wlan0-wlan15 refer to all the BSSIDs on radio1. Wlan16-wlan31 refers to all the BSSIDs on radio2.

Service show iwconfig: shows the status of the radio interfaces (detailed information on each BSSID of the AP):

```
cnWest-E400(config)# service show iwconfig
wlan25    IEEE 802.11ac  ESSID:""
          Mode:Master  Frequency:5.745 GHz  Access Point: Not-Associated
          Bit Rate:0 kb/s   Tx-Power=23 dBm
          RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=0/94  Signal level=-95 dBm  Noise level=-95 dBm
          Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
          Tx excessive retries:0  Invalid misc:0   Missed beacon:0
```

Service show dmesg : shows all the system startup messages, as well as radio-driver kernel logs. The radio driver logs are very verbose, and also the driver is common to a number of chipsets so a warning or even an error reported in the log might just be informational, for the radio-logs it is recommended to run things past Cambium support or Cambium engineering.

```
cnWest-E400(config)# service show dmesg|more
[4045418.180000] ar9300_handle_radar_bb_panic: BB status=0x04008009 rifs=0 - disable
[4045418.190000] ar9300_abort_tx_dma[938]: ar9300_stop_dma_receive failed
[4045418.190000] ar9300_reset[5836]: ar9300_stop_dma_receive failed
[4049876.910000] wmi_unified_event_rx : no registered event handler : event id 0x901b
[4049877.490000] [radio1] FWLOG: [120452248] WAL_DBGID_SECURITY_UCAST_KEY_SET ( 0x0 )
[4049877.500000] [radio1] FWLOG: [120452266] RATE: ChainMask 1, phymode 10, ni_flags
0x06013006, vht_mcs_set 0xfffe, ht_mcs_set 0x00ff, legacy_rate_set 0x72df4bc
[4049877.510000] [radio1] FWLOG: [120452284] WAL_DBGID_SECURITY_UCAST_KEY_SET ( 0x0 )
[4049877.520000] [radio1] FWLOG: [120452284] WAL_DBGID_SECURITY_ENCR_EN (  )
[4049877.530000] [radio1] FWLOG: [120452284] WAL_DBGID_SECURITY_ALLOW_DATA ( 0x436ee8 )
[4049877.530000] [radio1] FWLOG: [120452349] WAL_DBGID_TX_BA_SETUP ( 0x436ee8, 0x6, 0x2, 0x20,
0x1 )
[4049877.540000] [radio1] FWLOG: [120452350] RATE: ChainMask 1, phymode 10, ni_flags 0x06013006,
vht_mcs_set 0xfffe, ht_mcs_set 0x00ff, legacy_rate_set 0x72df609
```

Service system-trace : this command can be used to invoke the *strace* program which displays all system calls made by the process. This is useful to help troubleshooting a process that might be spinning (100% CPU utilization) or appears stuck.

```
cnWest-E400(config)# service system-trace 432
Process 432 attached
clock_gettime(CLOCK_MONOTONIC, {4522705, 765175967}) = 0
gettimeofday({1464680180, 268531}, NULL) = 0
clock_gettime(CLOCK_MONOTONIC, {4522705, 766201541}) = 0
epoll_wait(4, {}, 130, 118)             = 0
clock_gettime(CLOCK_MONOTONIC, {4522705, 885457006}) = 0
clock_gettime(CLOCK_MONOTONIC, {4522705, 885954506}) = 0
```