cisco Live!

Let's go

The bridge to possible

Troubleshooting Expressway via command line interface

Jamie Wang -Technical Consulting Engineer
Michael Zheng Huang - Customer Delivery Engineering Technical Leader

BRKCOL-3013

Agenda

- Background
- Basic admin CLI command
- Advanced root CLI command
- Troubleshooting web access loss
- Documentation as reference

Background

Background

- To ensure efficient troubleshooting of Expressway issues, it is important to have the ability to troubleshoot via CLI in addition to the web interface.
- This is particularly useful when customers are more familiar with CLI or if the web interface is down. Without this knowledge, resolution time may be prolonged.
- In this session, we will cover some verification and debug commands and general troubleshooting steps. And I will cover some troubleshooting steps on web access loss.

Basic admin CLI commands

Basic admin CLI commands

These are the top-level commands over Admin CLI.

about - Displays the system version information

configlog - Displays lines from log files

eventlog - Displays lines from log files

networklog - Displays lines from log files

help - Displays help for the top-level commands

license - Lists and displays third party software licenses

relkey - Gets and sets the system release key

Based on Version 14.x.

xcommand - <type "xcommand help" for more details>

xconfiguration - <type "xconfiguration help" for more details>

xfeedback - Provide information about events as they happen

xgetxml - Displays an XML description of some configuration

xhistory - <type "xhistory help" for more details>

xstatus - <type "xstatus help" for more details>

bye - Exits the shell

Basic admin CLI commands examples

xstatus //systemunit hardware #check system hardware and serial number

*s SystemUnit: /

Hardware:

SerialNumber: "0A60F0AF"

Version: "VMware"

xstatus // SystemUnit software version #check software version

*s SystemUnit: /

Software:

Version: "X14.0.7"

xstatus // SystemUnit Uptime # check system Uptime

*s SystemUnit: /

Uptime: "687940"

xstatus //sys sys #check system time

*s SystemUnit: /

SystemTime: "2023-03-25 04:59:36"

xstatus //systemunit time #check timezone

*s SystemUnit: /

TimeZone: "Asia/Shanghai"

xstatus // sys local #check local time

*s SystemUnit: /

LocalTime: "2023-03-25 13:01:18"

Basic admin CLI commands examples

xstatus // Applications //Cluster #check cluster status

```
*s Applications: /
  External:
    Status:
      ClusterStatus:
        ClusterLastSyncDate: "2023-03-25 13:07:12"
        ClusterLastSyncResult: "SUCCEEDED"
        ClusterNextSyncDate: "2023-03-25 13:08:12"
        ClusterState: "Enabled"
```

xstatus // Options #list option key added

```
*s Options: /
  Option:
    1:
      Description: "5 Rich Media Sessions"
      Key: "116XXXXX-X-XXXXXXX"
```

xcommand OptionKeyAdd 116XXXXX-X-XXXXXXX #add option key

```
*r Result (status=OK)
  ID: 1
*r/end
```

xcommand OptionKeyDelete 116XXXXX-X-XXXXXXX # del option key

```
*r Result (status=OK): /
*r/end
```

Basic admin CLI commands examples

xstatus // Warnings #check system warnings

217:

ID: "b62c4b06-84f1-49da-a8d0-0d6571f32194"

Reason: "Peer not responding - A peer address for the toGS neighbour zone is down or unreachable"

State: "Unacknowledged”

xcommand WarningAcknowledge warningID:b62c4b06-84f1-49da-a8d0-0d6571f32194 #acknowledge warning

xstatus // Warnings

217:

ID: "b62c4b06-84f1-49da-a8d0-0d6571f32194"

Reason: "Peer not responding - A peer address for the toGS neighbour zone is down or unreachable"

State: "Acknowledged"

Basic admin CLI commands examples

xcommand Networkinterface ? #To enable, disable and verify Dual interface (Dual int requires option key)

```
xCommand Networkinterface
  "Enable/disable network interfaces"
    DedicatedManagementInterface: <not_set/enable/disable/status>
    DualInterfaces: <not_set/enable/disable/status>
```

xCommand Networkinterface DualInterfaces: status

```
time_emitted | result
-------------+-------
0 ms         | true
```

xcommand boot #reboot this server

xcoomand restart #restart this server

Basic admin CLI commands examples

xcommand ping 8.8.8.8

```
time_emitted | host    | ttl | time_ms
-------------+---------+-----+--------
1220 ms      | 8.8.8.8 | 116 | 2.17
```

xcommand traceroute X.X.X.X

xcommand dnslookup dltaclab.com

```
time_emitted | question_type | answer_name | answer_ttl | answer_class | answer_type | answer_data
-------------+---------------+-------------+------------+--------------+-------------+------------
2670 ms      | SRV           | _sips._tcp.dltaclab.com. | 0          | IN           | SRV         | 10 10 5061 expe.dltaclab.com.
2680 ms      | SRV           | _sip._tcp.dltaclab.com. | 0          | IN           | SRV         | 10 10 5060 expe.dltaclab.com.
2680 ms      | SRV           | _collab-edge._tls.dltaclab.com. | 0          | IN           | SRV         | 10 10 8443 expe.dltaclab.com.
```

Basic admin CLI commands examples

xconfiguration // IP v4 #to find out the Ethernet IP settings

*c xConfiguration Ethernet 1 IP V4 Address: "10.10.X.X"

*c xConfiguration Ethernet 1 IP V4 StaticNAT Address: "64.104.X.X"

*c xConfiguration Ethernet 1 IP V4 StaticNAT Mode: "On"

*c xConfiguration Ethernet 1 IP V4 SubnetMask: "255.255.255.0"

*c xConfiguration Ethernet 2 IP V4 Address: "64.104.X.X"

*c xConfiguration Ethernet 2 IP V4 StaticNAT Address: "127.0.0.1"

*c xConfiguration Ethernet 2 IP V4 StaticNAT Mode: "Off"

*c xConfiguration Ethernet 2 IP V4 SubnetMask: "255.255.X.X"

*c xConfiguration Ethernet 3 IP V4 Address: "192.168.0.100"

*c xConfiguration Ethernet 3 IP V4 StaticNAT Address: "127.0.0.1"

*c xConfiguration Ethernet 3 IP V4 StaticNAT Mode: "Off"

*c xConfiguration Ethernet 3 IP V4 SubnetMask: "255.255.255.0".

xconfiguration IP Route #check static route settings

*c xConfiguration IP Route 1 Address: "10.70.X.X"

*c xConfiguration IP Route 1 Gateway: "64.104.X.X"

*c xConfiguration IP Route 1 Interface: "LAN2"

*c xConfiguration IP Route 1 PrefixLength: "26”

xconfiguration IP Gateway #check IPv4 gateway

*c xConfiguration IP Gateway: "10.10.X.X”

xconfiguration IP DNS # check DNS hostname and domain settings

*c xConfiguration IP DNS Domain Name: "dltaclab.com"

*c xConfiguration IP DNS Hostname: "expe”

xconfiguration DNS #check DNS server settings

*c xConfiguration DNS Server 1 Address: "64.104.X.X"

Basic admin CLI commands examples

xconfiguration Administration #check administration settings

*c xConfiguration Administration HTTP Mode: Off

*c xConfiguration Administration HTTPS Mode: On

*c xConfiguration Administration DeviceProvisioning Mode: Off

*c xConfiguration Administration SSH Mode: On

*c xConfiguration Administration SerialConsole Mode: On

xConfiguration SIP Advanced SipTlsVersions: **TLSv1.2**
#disable other SIP TLS versions

xconfiguration SystemUnit Maintenance #check if system is in MW or not

*c xConfiguration SystemUnit Maintenance Mode: "Off"

configlog [n|all|clear]

n – number of lines from end of log to dump

all – dump whole log

clear – delete this log

#configuration log provides a list of all changes to the expressway configuration

networklog [n|all|clear]

#network log provides a list of the call signaling messages that have been logged on this expressway

eventlog [n|all|clear]

#event log provides a list of the events that have occurred on your system since the last upgrade

Advanced root CLI Commands

Advanced root CLI commands

Logging in first as admin will not allow any switch to root, therefore you can log in as root first and switch to Tanberg and back to root anytime if needed

Log in as root and then type "tsh" to move to admin/Tandberg, then type "bye" to go back to root if needed.

~ # tsh

TANDBERG Video Communication Server X14.0.7

SW Release date: 2022-05-19 12:14, build

OK

bye

Bye!

~ #

Advanced root CLI commands->Verification commands

These are the steps to follow in order to verify status, configurations, statistics and other important facts that can be useful on troubleshooting.

Verify basic network connectivity, DNS, routes, etc.

- **ping** <IP, domain>
- **traceroute** <IP, domain>
- **nslookup** <domain>
- **dig** <domain>
- **routel**, **ip route**
- **ifconfig**
- **ethtool [eth0|eth1]**

Advanced root CLI commands->Verification commands

Verify CPU usage in real time, memory, other processes, etc.

The most common command used for verifying CPU and memory usage in real time is top, however we can also take advantage of the command **htop** which provides a more graphical output, see example below, Unlike to top command **htop** gets stopped with “**F10**” keywor

```
1  [|                                0.7%]   Tasks: 105, 237 thr; 1 running
2  [||                               3.9%]   Load average: 0.07 0.06 0.06
Mem[||||||||||||||||||||||||| 1.97G/5.81G]   Uptime: 4 days, 04:24:49
Swp[                            0K/10.0G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
 8221 root       20   0  428M  112M 10580 S  1.3  1.9 57:18.25 /lib64/erlang/erts-8.3.
17464 root       20   0  279M  161M 30884 S  1.3  2.7 59:51.08 /tandberg/images/ivy -f
21008 root       20   0  5012  3424  2880 R  0.7  0.1  0:00.21 htop
11575 root       20   0  653M  398M 46296 S  0.7  6.7 50:58.14 /tandberg/images/app
 8259 root       20   0  428M  112M 10580 S  0.7  1.9 27:26.11 /lib64/erlang/erts-8.3.
11749 root      -51   0  653M  398M 46296 S  0.7  6.7 13:53.04 /tandberg/images/app
10500 root       20   0  653M 87728 20076 S  0.7  1.4  4:51.68 /bin/python /share/pyth
 8768 root       20   0  653M 87728 20076 S  0.0  1.4  1h25:57 /bin/python /share/pyth
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

Advanced root CLI commands->Verification commands

Verify System TLS version based on cipher.

~ # **openssl ciphers -v**

ECDHE-RSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=RSA Enc=AESGCM(256) Mac=AEAD

ECDHE-ECDSA-AES256-GCM-SHA384 TLSv1.2 Kx=ECDH Au=ECDSA Enc=AESGCM(256) Mac=AEAD

ECDHE-RSA-AES256-SHA SSLv3 Kx=ECDH Au=RSA Enc=AES(256) Mac=SHA1

ECDHE-ECDSA-AES256-SHA SSLv3 Kx=ECDH Au=ECDSA Enc=AES(256) Mac=SHA1

~ # **openssl ciphers -v | awk '{print $2}' | sort | uniq**

SSLv3

TLSv1.2

Specifies the supported SIP TLS protocol versions. Default: TLSv1:TLSv1.1:TLSv1.2

xConfiguration SIP Advanced SipTlsVersions: **TLSv1.2** **<<<<disable other SIP TLS versions.**

Advanced root CLI commands->Verification commands

Verify active connections/ports on server

netstat -an | grep PORTNUMBER

```
~ # netstat -an | grep 5060
tcp        0      0 127.0.0.1:5060          0.0.0.0:*               LISTEN
tcp        0      0 10.10.12.2:5060         0.0.0.0:*               LISTEN
tcp        0      0 10.1.1.151:5060         0.0.0.0:*               LISTEN
tcp        0      0 10.1.1.151:26471        10.1.2.137:5060         ESTABLISHED
tcp        0      0 ::1:5060                :::*                    LISTEN
udp        0      0 127.0.0.1:5060          0.0.0.0:*
udp        0      0 ::1:5060
```

lsof -i and/or **netstat -tanp** can also be issued to verify all ports status.

Advanced root CLI commands->Verification commands

Verify Round Trip Delay Between Expressway Cluster Nodes. Expressway supports a round trip delay of up to 80ms. This means that each Expressway in the cluster must be within a 40ms hop of all other peers in the cluster.

~# ping -i 0.03 -s 4000 10.124.42.74

Let the ping run for one to two minutes.

Press **Ctrl + C** in order to stop the ping after one to two minutes.

A summary with the average RTT displays at the end of the output:

--- 10.124.42.74 ping statistics ---

1226 packets transmitted, 1226 received, 0% packet loss, time 37980ms

rtt min/avg/max/mdev = 0.102/0.255/0.646/0.070 ms

Advanced root CLI commands->Debug commands

This section focuses on the "**tcpdump**" feature, which is available via root and can be customized to display only relevant information using various combinations.

General debug on all ports, to stop the output, press Ctrl +C

~ # **tcpdump**

14:55:57.611868 IP expe.ssh > 10.140.249.142.50507: Flags [P.], seq 3096222186:3096222426, ack 1166705145, win 501, options [nop,nop,TS val 1243279615 ecr 254629597], length 240

14:55:57.667994 IP 10.140.249.142.50507 > expe.ssh: Flags [.], ack 0, win 2104, options [nop,nop,TS val 254629678 ecr 124

3279593], length 0

^C

103 packets captured

105 packets received by filter

0 packets dropped by kernel

Advanced root CLI commands->Debug commands

Debug Interfaces.

All Interfaces:

```
tcpdump -i any
```

Only that interface:

```
tcpdump -i eth0
```

To a specific IP address, .73 on the example:

```
tcpdump host 10.124.42.73
```

Advanced root CLI commands->Debug commands

Prior to X14, the tcpdump feature on the expressway web interface had a 50 MB packet file size limit that could result in incomplete packet captures. However, later versions increased this limit by allowing up to 20 pcap files per LAN with each file being limited to 20MB in size.

Start tcpdump:

mkdir /mnt/harddisk/traces

Enter command, **tcpdump -w /mnt/harddisk/traces/trace-eth0.pcap -s 0 -C 40 -W 100 -i eth0**

For dual network system, open another terminal and SSH into Exp-E as root again

Enter command,**tcpdump -w /mnt/harddisk/traces/trace-eth1.pcap -s 0 -C 40 -W 100 -i eth1**

Stop and log collection:

Press **Ctrl+C** to stop tcpdump

/mnt/harddisk/traces/

Name	Size	Changed	Rights	Owner
..		3/26/2023 3:26:12 PM	rwxr-xr-x	root
trace-eth1.pcap00	1,270 KB	3/26/2023 3:26:51 PM	rw-r--r--	root

Download tcpdump files under **/mnt/harddisk/traces** directory via any SFTP server.

Advanced root CLI commands->Debug commands

Debug HTTP/HTTPS request.

```
~ # tcpdump port 443 or 80
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
15:52:54.463110 IP 10.140.249.142.52606 > expresswayc1hq.443: Flags [S], seq 1782977, win 65535, options [mss 1250,nop,wscale 6,nop,nop,TS val 3652393708 ecr 0,sackOK,eol], length 0
15:52:54.463144 IP expresswayc1hq.443 > 10.140.249.142.52606: Flags [S.], seq 850957238, ack 1782978, win 65160, options [mss 1460,sackOK,TS val 639368885 ecr 3652393708,nop,wscale 7], length 0
```

The server should receive and respond to HTTP/S requests.

If no such requests are seen after running this command is either because:

1. Ports are not opened on network. (443, 80, etc)
2. HTTP/S requests are getting sent from remote device to access the web but getting dropped before hitting the server (network issue).
3. A possible issue on browser.

Advanced root CLI commands->Debug commands

Debug DNS request.

~ # **tcpdump udp port 53**

16:05:38.822814 IP expe.35965 > dns-tokyo.dltaclab.com.domain: 19010+ NAPTR? webex.com. (27)

16:05:38.834587 IP dns-tokyo.dltaclab.com.domain > expe.35965: 19010 0/1/0 (109)

16:05:38.835114 IP expe.34679 > dns-tokyo.dltaclab.com.domain: 23852+ SRV? _sips._tcp.webex.com. (38)

16:05:38.843410 IP dns-tokyo.dltaclab.com.domain > expe.34679: 23852 2/0/0 SRV geo-sec-1.cmr.webex.com.:5061 40 100, SRV geo-pri-1.cmr.webex.com.:5061 20 100 (124)

The server should send and process DNS requests and receive replies from the DNS.

Running this command will display any DNS requests that have been answered or show the output of whatever answer was received.

In some DNS troubleshooting scenarios you will need to clear DNS cache, run the following command:

/etc/init.d/dnsmasq restart

Advanced root CLI commands->Debug commands

Debug Media ports(range basis)

~ # tcpdump -an udp portrange 36002-59999

16:13:14.270156 IP 10.10.12.2.36234 > 10.72.133.237.51718: UDP, length 81

16:13:14.270201 IP 10.10.12.2.36237 > 10.72.133.237.57089: UDP, length 28

16:13:14.286501 IP 10.72.133.237.51718 > 10.10.12.2.36234: UDP, length 184

16:13:14.286522 IP 10.72.133.237.57088 > 10.10.12.2.36236: UDP, length 33

Other tcpdump combinations can be used in order to customize your filters depending on src/dst IP, such:

tcpdump dst 192.168.10.82 and -an portrange 36002-59999

tcpdump host 192.168.10.82 and -an portrange 36002-59999

This can simplify and expedite the diagnosis of server issues related to media traversal.

To identify a particular media type passing through the server, check the diagnostic log for SDP's media ports and find them in the output.

Advanced root CLI commands->System commands

Get a backup, snapshot file.

~ # **/sbin/backup.sh** >>>creates a tar.gz.enc backup file as done through Maintenance > Backup & restore

Password to encrypt backup with:

Starting backup...

Backup complete:
/mnt/harddisk/backuprestore/system_backup/expe.dltaclab.com_X14.0.7_XXXXXXXX_2023_03_26__16_29_12_backup.tar.gz.enc

~ # **snapshot.sh**

/mnt/harddisk/snapshot/XXXXXXXX_2023_03_26__16_38_46_full_sysdump.tar.gz

Wait for the snapshot to be generated, it will take some time - file name will be reported as the command completes.

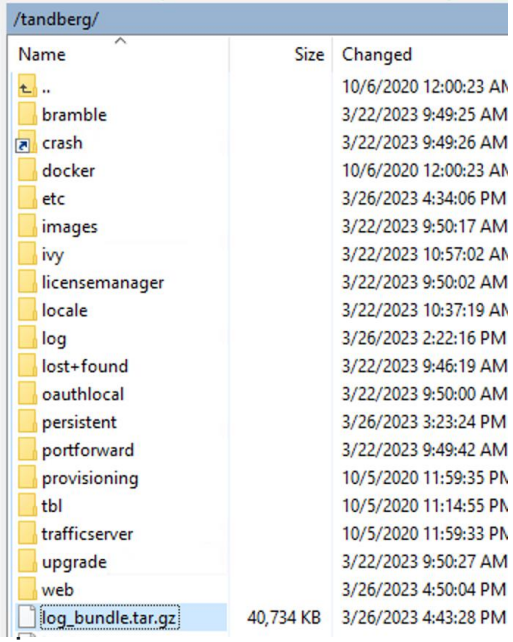
Snapshot can be found in **/mnt/harddisk/snapshot** as a (.tar.gz) file.

Once found it can be moved to PC side and save it for further research.

Advanced root CLI commands->System commands

Get the log files that you can normally find within the **harddisk** log folder in the Expressway snapshot.

~ # **tar -czvf log_bundle.tar.gz /mnt/harddisk/log/**
tar: Removing leading `/' from member names
/mnt/harddisk/log/
/mnt/harddisk/log/network_log.12
/mnt/harddisk/log/smartlicensedaemon_log.4
/mnt/harddisk/log/smartlicenseagent_log.1
/mnt/harddisk/log/sensors.1
/mnt/harddisk/log/critical
/mnt/harddisk/log/smartlicensedaemon_log.10
/mnt/harddisk/log/fail2ban.log
/mnt/harddisk/log/packagesd.log



Wait until the output stops by itself .
Open Winscp and navigate to /tandberg and locate the folder.
Drag and drop to your pc.
Once done, delete the folder with right-click and delete to avoid space issues.
Or remove it from root: **rm log_bundle.tar.gz**

Advanced root CLI commands->System commands

Extract certificates. The certificates was under the path **/tandberg/persistent/certs/**

```
~ # ls /tandberg/persistent/certs/
ca.pem        client-ca.crl.default  generated_csr  multidomaincerts    policy-services.crl.default  saml        server-ssh.pem
client-ca.crl  crl-update.conf       mtls_ca.pem    policy-services.crl  privkey.pem                  server.pem
```

~ # **cat /tandberg/persistent/certs/ca.pem** <it will list all the CA certificates on the trusted CA store.

O=QuoVadis Limited, CN=QuoVadis Root CA 2


```
-----BEGIN CERTIFICATE-----
MIIFtzCCA5+gAwIBAgICBQkwDQYJKoZIhvcNAQEFBQAwRTELMAkGA1UEBhMCQk0x
.....
8eOx79+Rj1QqCyXBJhnEUhAFZdWCEOrCMc0u
-----END CERTIFICATE-----
```


Once this output shows up copy and paste it a notepad, save it as *<file name>.cer* in order to have it checked if needed.
Another method is to use winscp to drag the certificate to your PC directly.

Troubleshooting web access loss

Basic Initial steps

You can initially try the following basic tests:

- Try a different browser. Check for the **Supported browsers** section on Expressway Admin guide.
- Try to access the server from a different network to isolate network-related issues. Admin can also try with a different pc.
- Try to access all hosted IPs on the server, especially if it hosts multiple IPs like Expressway Es setups
- Try IP instead of hostname to isolate DNS issues.
- If the current browser tab does not work open a new one.
- Ensure you enter **https://** instead of **http://** , enter admin port if you use ports such as 445, 7443, among others.

Web access disabled in the server

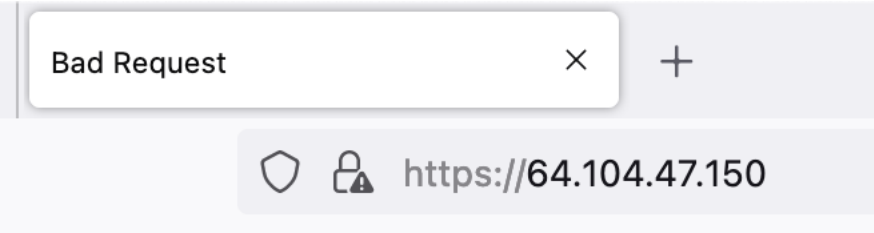
```
xconfig admin
*c xConfiguration Administration LCDPanel Mode: On
*c xConfiguration Administration IntrusionProtection Mode: On
*c xConfiguration Administration HTTP Mode: Off
*c xConfiguration Administration HTTPS Mode: Off
```

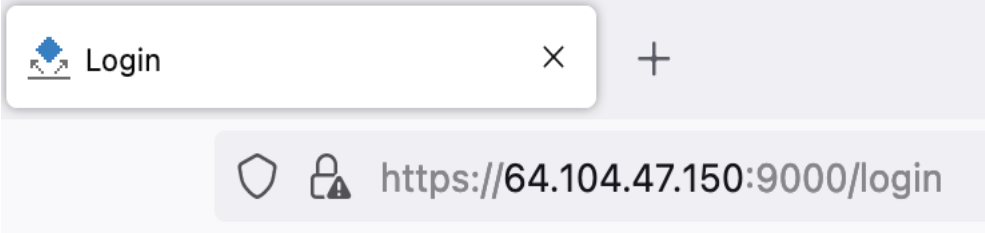
If HTTPS is disabled on the server, the server-side output will display a [R] flag (reset), as shown in the example below:

```
~ # tcpdump port 443
17:17:50.154696 IP 10.140.249.142.53971 > expresswayc1hq.443: Flags [R], seq 1527735919, win 0, length 0
17:17:50.154720 IP 10.140.249.142.53971 > expresswayc1hq.443: Flags [R], seq 1527735919, win 0, length 0
```

Web portal port-related

By default, when HTTPS is attempted, the browser uses 443, sometimes the server has a different port configured for web portal and thus the web page does not completely come up and rather shows "*Bad Request*", see example below.






```
xconfig manage
*c xConfiguration Management Interface Http HstsMode: On
*c xConfiguration Management Interface WebAdministration Port: 9000
*c xConfiguration Management Session InactivityTimeout: 30
*c xConfiguration Management Session MaxConcurrentSessionsTotal: 0
*c xConfiguration Management Session MaxConcurrentSessionsUser: 0
```

Network configuration in the server

If there are recent, new, or incorrect IP modifications, web access may be lost and SSH access may not be possible. In such cases, it is important to ensure that console access via vSphere is granted.

- Log in as root.
- Switch to Tandberg (Admin) with "**tsh**" command
- **xconfig ethernet** to check the IP configuration
- **xconfig IP gateway** to verify what the server gateway is
- **xconfig IP external** to verify the external interface is correct in case the server is an Expressway E.
- **xconfig IP route** to verify that static routes are properly configured

Ensure the IP, subnet, and gateway belong to the same and correct subnet and verify they represent the correct values that this network portion should have. Depending on the network design, some subnets can only be reached through static routes in the E server.

Network configuration in the server

Duplicated IPs. In order to isolate this scenario, follow the steps below,

- Log in as root within the vSphere in order to track the traffic exchanged.
- Debug ICMP and port 443 with **tcpdump icmp or port 443** root command.
- Open another CLI from another platform such as pc, server, etc.
- Ping the server in question, run **ping <server IP>**
- Send packets towards server IP with port 443 with the root command **wget <server IP>:443**
- Check if sent packets are shown in the server console output.

```
~ # tcpdump icmp or port 443

^C
0 packets captured
0 packets received by filter
0 packets dropped by kernel
~ #
```

```
~ # ping 192.168.1.7 -c 2
PING 192.168.1.7 (192.168.1.7) 56(84) bytes of data.
64 bytes from 192.168.1.7: icmp_seq=1 ttl=46 time=80.4 ms
64 bytes from 192.168.1.7: icmp_seq=2 ttl=46 time=78.4 ms
--- 192.168.1.7 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 78.431/79.295/80.398/0.820 ms
~ # wget 192.168.1.7:443
--2023-03-26 17:44:34--  http://192.168.1.7:443/
Connecting to 192.168.1.7:443... connected.
HTTP request sent, awaiting response... No data received.
Retrying.
```

TCP Port related

When network ports are not opened or routing for the web port is not properly set up web portal connections fail.

- Log in as root.
- Run **tcpdump -n port 443 -tttt**
- Try to access the web and verify if the packets are received.
- The expected sequence is the three-way handshake (SYN, SYNACK, ACK).
- If no incoming packets are received ensure the network is accepting traffic over port 443 or the access port in use (445, among others).
- The messages show as follows:
- [S] SYN
- [S.] SYNACK
- [.] ACK

TCP Port related

Successful TCP connection:

~ # **tcpdump -n port 9000 -tttt**

2023-03-26 18:02:11.430487 IP **10.140.249.142.54688** > **10.10.12.2.9000**: Flags [S], seq 1551636100, win 65535, options [mss 1250,nop,wscale 6,nop,nop,TS val 2193885641 ecr 0,sackOK,eol], length 0

2023-03-26 18:02:11.430551 IP **10.10.12.2.9000** > **10.140.249.142.54688**: Flags [S.], seq 3388609280, ack 1551636101, win 65160, options [mss 1460,sackOK,TS val 1254453434 ecr 2193885641,nop,wscale 7], length 0

•2023-03-26 18:02:11.505257 IP **10.140.249.142.54688 > 10.10.12.2.9000**: Flags [.], ack 1, win 2050, options [nop,nop,TS val 2193885718 ecr 1254453434], length 0

Invalid certificate issue

This issue commonly happens when certificates were signed using an unsupported signature algorithm (e.g., RSASSA-PSS). This algorithm has not been officially tested nor included in Expressway code as a result server cannot properly parse the file preventing services from properly starting up after restarting the server.

In order to resolve this issue the newly uploaded certificate needs to be removed from the server, after the file is removed and server restarted the services should start properly again.

Prior x12.5.x, the newly uploaded invalid certificate could be overwritten with a default certificate using the **copy** function.

- **Step 1.** Remove the certificate

```
~ # cp /tandberg/persistent/certs/server.pem.default /tandberg/persistent/certs/server.pem
~ # cp /tandberg/persistent/certs/privkey.pem.default /tandberg/persistent/certs/privkey.pem
~ # cp /tandberg/persistent/certs/ca.pem.default /tandberg/persistent/certs/ca.pem
```

- **Step 2.** Server reboot

- After certification removal the server requires a reboot to apply changes, this can be done via reboot function.

```
~ # reboot
```


Invalid certificate issue

x12.5.x or later

- In newer versions, the default certificate is no longer listed in the certificate path, so it needs to be manually removed using **remove** function.
- **Step 1.** Remove the certificate

~ # rm persistent/certs/server.pem
~ # rm persistent/certs/privkey.pem
~ # rm persistent/certs/ca.pem

- **Step 2.** Server reboot
- After certification removal the server requires a reboot to apply changes, this can be done via reboot function:

~ # reboot Once server boots up again, webUI will respond.

Documentation as reference

References

Expressway administrator guide:
https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/expressway/admin_guide/X14-0/exwy_b_cisco-expressway-administrator-guide/exwy_m_reference-material.html

Redhat Command-Line interface Reference:

https://access.redhat.com/documentation/zh-cn/red_hat_enterprise_linux_openstack_platform/7/html-single/command-line_interface_reference_guide/index

Tcpdump reference:

http://www.tcpdump.org/

Thank you

cisco Live!

Let's go