Implementing Host Services and Applications

Cisco IOS XR software Host Services and Applications features on the router are used primarily for checking network connectivity and the route a packet follows to reach a destination, mapping a hostname to an IP address or an IP address to a hostname, and transferring files between routers and UNIX workstations.

Prerequisites for implementing Host Services and Applications

Ensure to install the relevant optional RPM package before using the host services or applications.

HTTP client application

A HTTP client application is a network application that

- allows files to be transferred from HTTP servers to other devices over a network,
- uses the HTTP protocol for communication, and
- provides configurable parameters for connection management and security.

TCP

TCP is a connection-oriented protocol that

- specifies the format of data and acknowledgments that two computer systems exchange to transfer data,
- defines the procedures computers use to ensure that data arrives correctly, and
- allows multiple applications on a system to communicate concurrently by handling all demultiplexing of incoming traffic among the application programs.

TCP dump file converters

TCP dump file converters are tools that

- convert IOS-XR dump files in binary format to user-friendly formats such as Packet Capture (PCAP) or text,

- provide access to packet traces stored during Non-Stop Routing (NSR) disablement or session flaps, and
- facilitate network troubleshooting by making packet trace data readable and analyzable.

When TCP dump files are created

When you disable NSR or experience a session flap on your system, the TCP process running on the NCS system automatically stores the latest 200 packet traces in binary format within a temporary folder.

TCP dump packet traces include:

- data about the configured routing protocols, and
- overall network traffic traversing your system.

This data provides the necessary insights to identify and resolve issues within your network infrastructure, facilitating proactive network troubleshooting.

Methods to view packet traces

You can view the packet traces binary files in user-readable format using these methods:

- Manual viewing: Use the `show tcp dump-file <binary filename>` command to view each binary file in text format manually. For details about viewing binary files manually, see View binary files in text format manually. This process is time-consuming, as you have to view each file manually one after another.
- Automated conversion: Convert all stored packet traces in binary files into PCAP, text, or both using the `tcp dump-file convert` command. For details about converting binary files, see Convert binary files to readable format using TCP dump file converter. This active approach greatly improves the efficiency and ease of packet analysis during network troubleshooting.

Caution: Observe limitations when using the TCP dump file converter

The system stores only the most recent 200 message exchanges that occur immediately before session termination, when NSR is disabled, or during a session flap.

You can view only one binary file in text format at a time using the `show tcp dump-file <binary filename>` command.

TCP dump files are generated by default for Border Gateway Protocol (BGP), Multicast Source Discovery Protocol (MSDP), Multiprotocol Label Switching Label Distribution Protocol (MPLS LDP), and SSH.

View binary files in text format manually

This procedure enables you to view the contents of packet trace binary files in text format individually, without converting them.

This method allows you to quickly view the contents of individual binary files stored in the tcpdump folder. However, viewing multiple files is time-consuming because you must view each file individually.

Procedure

Step 1 Run the `show tcp dump-file list all` command, to view the list of packet traces in binary files stored in the tcpdump folder.

Example:

```
RP/0/RP0/CPU0:ios# show tcp dump-file list all
total 1176
-rw-r--r-- 1 root root 5927 Nov 22 12:42 31_0_0_126.179.20966.cl.1700656933
-rw-r--r-- 1 root root 5892 Nov 22 12:42 31_0_0_127.179.35234.cl.1700656933
-rw-r--r-- 1 root root 6148 Nov 22 12:42 31_0_0_149.179.54939.cl.1700656933
-rw-r--r-- 1 root root 5894 Nov 22 12:42 31_0_0_155.179.18134.cl.1700656933
-rw-r--r-- 1 root root 6063 Nov 22 12:42 31_0_0_156.179.25445.cl.1700656933
-rw-r--r-- 1 root root 5860 Nov 22 12:42 31_0_0_161.179.30859.cl.1700656933
-rw-r--r-- 1 root root 5832 Nov 22 12:42 31_0_0_173.179.36935.cl.1700656933
-rw-r--r-- 1 root root 5906 Nov 22 12:42 31_0_0_190.179.25642.cl.1700656933
```

Step 2 Run the `show tcp dump-file <binary filename>` command to view the contents of a selected packet trace binary file in text format .

Example:

```
RP/0/RP0/CPU0:ios# show tcp dump-file 10_106_0_73.179.34849.cl.1707424077 location 0/RP0/CPU0
Filename: 10_106_0_73.179.34849.cl.1707424077

================================================================
Connection state is CLOSED, I/O status: 0, socket status: 103
PCB 0x00007f86bc05e3b8, SO 0x7f86bc05e648, TCPCB 0x7f86bc0c3718, vrfid 0x60000000,
Pak Prio: Medium, TOS: 192, TTL: 1, Hash index: 1593
Local host: 10.106.0.72, Local port: 179 (Local App PID: 11354)
Foreign host: 10.106.0.73, Foreign port: 34849
(Local App PID/instance/SPL_APP_ID: 11354/1/0)

Current send queue size in bytes: 0 (max 0)
Current receive queue size in bytes: 0 (max 0)  mis-ordered: 0 bytes
Current receive queue size in packets: 0 (max 0)

Timer          Starts    Wakeups         Next(msec)
Retrans        103448          8                0
SendWnd             0          0                0
TimeWait            1          0                0
AckHold        106815     106545                0
KeepAlive           1          0                0
PmtuAger            0          0                0
GiveUp              0          0                0
Throttle            0          0                0
FirstSyn            0          0                0

   iss: 161240548   snduna: 163206936   sndnxt: 163206936
sndmax: 163206936   sndwnd: 63104       sndcwnd: 18120
   irs: 3691232436  rcvnxt: 3693473072  rcvwnd: 26099   rcvadv: 3693499171
```

This sample displays only a part of the actual output; the actual output displays more details.

You can review the contents of each packet trace binary file in text format and analyze specific packet information as needed.

Convert binary files to readable format using TCP dump file converter

Use this procedure to convert all dump packet traces from binary files into PCAP and text formats to facilitate more effective and accessible packet analysis.

This automated approach allows you to convert all TCP dump binary files at once, significantly improving efficiency and ease of troubleshooting during network analysis.

Procedure

Step 1 Run the `tcp dump-file convert all-formats all` command to convert the dump packet traces in binary files into PCAP and text formats.

Example:

```
RP/0/RP0/CPU0:ios# tcp dump-file convert all-formats all
ascii file is saved at :
/harddisk:/decoded_dumpfiles/text_tcpdump_peer_all_node0_RP0_CPU0_2024_3_19_10_8_53.462070.txt
pcap file is saved at :
/harddisk:/decoded_dumpfiles/pcap_tcpdump_peer_all_node0_RP0_CPU0_2024_3_19_10_8_40.154838.pcap
[OK]
```

By default, the system stores the converted files in the "decoded_dumpfiles" folder on the hard disk.

Step 2 (Optional) Specify an alternate location and file name for the converted output using the `location node-id` and `file <file path>` keywords.

```
RP/0/RP0/CPU0:ios# tcp dump-file convert all-formats all location 0/RP0/CPU0 file /harddisk:/demo2
ascii file is saved at : /harddisk:/demo2.txt
pcap file is saved at : /harddisk:/demo2.pcap
[OK]
```

Step 3 Run the `run cat <text file path>` command to view the converted text file.

Example:

```
RP/0/RP0/CPU0:ios# run cat
/harddisk:/decoded_dumpfiles/text_tcpdump_peer_all_node0_RP0_CPU0_2024_3_19_10_8_53.462070.txt
Filename: 2024_3_19_10_8_53.462070

============================================================
Connection state is CLOSED, I/O status: 0, socket status: 103
PCB 0x0000000000f47a80, SO 0xf476d0, TCPCB 0xf6a370, vrfid 0x60000000,
Pak Prio: Medium, TOS: 192, TTL: 255, Hash index: 563
Local host: 14:11:11::1, Local port: 47743 (Local App PID: 19579)
Foreign host: 14:11:11::2, Foreign port: 179
(Local App PID/instance/SPL_APP_ID: 19579/1/0)

Current send queue size in bytes: 0 (max 0)
Current receive queue size in bytes: 0 (max 0)  mis-ordered: 0 bytes
Current receive queue size in packets: 0 (max 0)

Timer          Starts    Wakeups         Next(msec)
Retrans            70          2                  0
SendWnd             0          0                  0
TimeWait            2          0                  0
AckHold            66         61                  0
KeepAlive           1          0                  0
PmtuAger            0          0                  0
GiveUp              0          0                  0
Throttle            0          0                  0
```

```
FirstSyn            1          1               0

   iss: 3113104891  snduna: 3113106213  sndnxt: 3113106213
sndmax: 3113106213  sndwnd: 31523       sndcwnd: 2832
   irs: 4250126727  rcvnxt: 4250128049  rcvwnd: 31448   rcvadv: 4250159497
```

This sample shows only a part of the actual output. The real output contains more details.

Step 4 Copy the converted packet traces from the system to your local computer using the `scp` command and view the converted PCAP file.

The binary TCP dump packet traces are converted into readable PCAP and text files. These files are stored in your chosen directory for streamlined troubleshooting and analysis. You can now review and analyze network packet data.