Paketerfassung für Connected Mobile Experience (CMX)

Inhalt

Einführung

In diesem Dokument wird beschrieben, wie Paketerfassungen vom CMX-Server (CLI of Connected Mobile Experience) 10.x erfasst werden. Diese Paketerfassungen können bei der Fehlerbehebung in mehreren Szenarien helfen (z. B.: NMSP-Kommunikation zwischen Wireless LAN Controller (WLC) und CMX-Server) zur Validierung des Kommunikationsflusses.

Anforderungen

- CLI-Zugriff (Command Line Interface) auf den CMX-Server.
- Computer mit installiertem Wireshark, um die Aufzeichnungen detailliert zu lesen.

Verwenden von TCPDUMP für die Erfassung

TCPDUMP ist ein Paketanalysator, der die übertragenen und empfangenen Pakete auf dem CMX-Server anzeigt. Sie dient als Analyse- und Fehlerbehebungstool für Netzwerk-/Systemadministratoren. Das Paket ist in den CMX-Server integriert, wo die Rohdaten der Pakete eingesehen werden können.

Das Ausführen von tcpdump als 'cmxadmin'-Benutzer schlägt fehl, und es wird folgender Fehler ausgegeben: (Root-Zugriff erforderlich)

```
In this example, tcpdump is attempted to be run as a 'cmxadmin' user.

[cmxadmin@laughter ~]$ tcpdump -i eth0 port 16113
tcpdump: eth0: You don't have permission to capture on that device
(socket: Operation not permitted)
```

Wechseln Sie nach der Anmeldung als 'cmxadmin'-Benutzer zur CLI über SSH oder Konsole.

```
[cmxadmin@laughter ~]$ su - root
Password:
```

```
[root@laughter ~]#
```

Verwenden der richtigen Schnittstelle

Notieren Sie sich die Schnittstelle, über die die Pakete erfasst werden. Sie kann mit dem Befehl 'ifconfig -a' abgerufen werden.

```
In this example, 10.10.10.25 is the IP address of CMX server and 'eth0' is the interface it's
tied to on the server.
[cmxadmin@laughter ~]$ ifconfig -a eth0      Link encap:Ethernet  HWaddr 00:50:56:A1:38:BB
          inet addr:10.10.10.25  Bcast:10.10.10.255  Mask:255.255.255.0
          inet6 addr: 2003:a04::250:56ff:fea1:38bb/64 Scope:Global
          inet6 addr: fe80::250:56ff:fea1:38bb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:32593118 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3907086 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3423603633 (3.1 GiB)  TX bytes:603320575 (575.3 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:1136948442 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1136948442 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:246702302162 (229.7 GiB)  TX bytes:246702302162 (229.7 GiB)

[cmxadmin@laughter ~]$
```

Erfassen von Paketen

```
This example captures and displays all packets that are sourced from port - 16113 and enter the
CMX server on the eth0 interface.

[root@laughter ~]# tcpdump -i eth0 src port 16113 tcpdump: verbose output suppressed, use -v or
-vv for full protocol decode listening on eth0, link-type EN10MB (Ethernet), capture size 65535
bytes 09:50:29.530824 IP 172.18.254.249.16113 > laughter.cisco.com.40020: Flags [P.], seq
983381312:983382645, ack 2483597279, win 191, options [nop,nop,TS val 1792647414 ecr
1148435777], length 1333 09:50:31.507118 IP 172.18.254.249.16113 > laughter.cisco.com.40020:
Flags [.], seq 1333:2715, ack 1, win 191, options [nop,nop,TS val 1792647908 ecr 1148437650],
length 1382 09:50:31.507186 IP 172.18.254.249.16113 > laughter.cisco.com.40020: Flags [P.], seq
2715:2890, ack 1, win 191, options [nop,nop,TS val 1792647908 ecr 1148437650], length 175
09:50:33.483166 IP 172.18.254.249.16113 > laughter.cisco.com.40020: Flags [P.], seq 2890:4239,
ack 1, win 191, options [nop,nop,TS val 1792648402 ecr 1148439626], length 1349 09:50:35.459584
IP 172.18.254.249.16113 > laughter.cisco.com.40020: Flags [P.], seq 4239:5396, ack 1, win 191,
options [nop,nop,TS val 1792648896 ecr 1148441603], length 1157 ^C 5 packets captured 5 packets
received by filter 0 packets dropped by kernel [root@laughter ~]#
```

So schreiben Sie die Ausgabe in eine Datei

```
In this example, tcpdump would capture packets that are from 10.10.20.5 received on it's eth0
interface and write it to a file named TEST_NMSP_WLC.pcap.

[root@laughter cmxadmin]# tcpdump -i eth0 src 10.10.20.5 -w TEST_NMSP_WLC.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
^C7 packets captured
7 packets received by filter
```

```
0 packets dropped by kernel
[root@laughter cmxadmin]#
```

Sobald die Datei fertig ist, müssen Sie die .pcap-Datei aus dem CMX auf Ihren Computer extrahieren, um sie in einem komfortableren Tool wie Wireshark zu analysieren. Hierfür können Sie jede SCP-Anwendung verwenden. In Windows ermöglicht die WinSCP-Anwendung beispielsweise die Verbindung mit dem CMX mithilfe der SSH-Anmeldeinformationen. Anschließend können Sie das Dateisystem durchsuchen und die soeben erstellte .pcap-Datei suchen. Um den aktuellen Pfad zu finden, geben Sie nach der Ausführung des tcpdump "pwd" ein, um zu erfahren, wo die Datei gespeichert wurde.

So erfassen Sie eine bestimmte Anzahl von Paketen

Wenn eine bestimmte Anzahl von Paketen gewünscht wird, werden die -c-Optionsfilter genau für diese Anzahl verwendet.

```
[root@laughter ~]# tcpdump -Z root -i eth0 -c 5 src 10.10.20.5 -w CMX_WLC_Capture.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
5 packets captured
6 packets received by filter
0 packets dropped by kernel
[root@laughter ~]#
```

Andere Filteroptionen

```
[root@laughter cmxadmin]# tcpdump -i eth0 dst 10.10.20.5 (filtered based on destination IP
address)
[root@laughter cmxadmin]# tcpdump -i eth0 src 10.10.20.4 (filtered based on Source IP address)

[root@laughter cmxadmin]# tcpdump -i eth0 port 80  (filtered for packets on port 80 in both
directions)
[root@laughter cmxadmin]# tcpdump -i eth0 port 443  (filtered for packets on port 443 in both
directions)
```

Die in Dateien geschriebenen Captures werden im aktuellen Verzeichnis auf dem Server gespeichert und können mithilfe von Wireshark zur detaillierten Überprüfung kopiert werden.