Captures de paquets sur une expérience mobile connectée (CMX)

Contenu

Introduction

Ce document décrit comment collecter des captures de paquets à partir de l'interface de ligne de commande du serveur CMX (Connected Mobile Experience) 10.x. Ces captures de paquets peuvent aider à dépanner plusieurs scénarios (par exemple : Communication NMSP entre le contrôleur de réseau local sans fil (WLC) et le serveur CMX pour valider le flux de communication.

Conditions requises

- Accès CLI (Command Line Interface) au serveur CMX.
- Ordinateur avec Wireshark installé pour lire les captures en détail.

Utilisation de TCPDUMP pour les captures

TCPDUMP est un analyseur de paquets qui affiche les paquets transmis et reçus sur le serveur CMX. Il sert d'outil d'analyse et de dépannage pour les administrateurs réseau/système. Le paquet est intégré au serveur CMX où les données brutes des paquets peuvent être examinées.

L'exécution de tcpdump en tant qu'utilisateur 'cmxadmin' échouerait avec l'erreur suivante : ('accès racine requis)

```
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)
```

Basculez vers l'utilisateur 'root' après vous être connecté en tant qu'utilisateur 'cmxadmin' à l'interface de ligne de commande via SSH ou la console.

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

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

Utilisation de l'interface de droite

Notez l'interface dans laquelle les paquets sont capturés. Il peut être obtenu à l'aide de la commande ifconfig -a

```
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 ~]$
```

Capture de paquets

```
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 ~]#
```

Pour écrire la sortie dans un fichier

```
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]#
```

Une fois le fichier prêt, vous devez extraire le fichier .pcap de la CMX vers votre ordinateur pour l'analyser dans un outil plus confortable tel que wireshark. Vous pouvez utiliser n'importe quelle application SCP pour cela. Par exemple, sous Windows, l'application WinSCP vous permet de vous connecter à CMX à l'aide des informations d'identification SSH. Vous pouvez ensuite parcourir le système de fichiers et rechercher le fichier .pcap que vous venez de créer. Pour trouver le chemin d'accès actuel, tapez « pwd » après l'exécution de tcpdump pour savoir où le fichier a été enregistré.

Pour capturer un nombre spécifique de paquets

Si un nombre spécifique de paquets est souhaité, l'option -c filtre exactement ce nombre.

```
[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 ~]#
```

Autres options de filtrage

```
[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)
```

Les captures écrites dans les fichiers sont enregistrées dans le répertoire actif du serveur et peuvent être copiées pour examen détaillé à l'aide de Wireshark.