

Procédure pour dépister des pulsations entre CVP et UCCE

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Informations générales](#)

[Procédure](#)

Introduction

Ce document décrit comment dépister des messages de pulsation entre le Customer Voice Portal (CVP) Callserver et le gestionnaire d'interface périphérique d'unité de réponse vocale (VRU PIM) utilisant VRU PIM se connecte et capture Wireshark.

Conditions préalables

Exigences

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Serveur CVP
- Intelligent Contact Management de Cisco Unified (missile aux performances améliorées), déploiements du Cisco Unified Contact Center Enterprise (UCCE)
- Callserver et gestionnaire d'interface périphérique d'unité de réponse vocale (VRU PIM)

[Composants utilisés](#)

Les informations contenues dans ce document sont basées sur les versions de logiciel suivantes :

- Serveur 9.0 CVP et en haut
- UCCE 9.0 et en haut
- Wireshark

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

[Informations générales](#)

Des messages de pulsation sont permutés entre le serveur et le VRU PIM d'appel CVP utilisant le port configuré sur le sous-système missile aux performances améliorées du serveur d'appel CVP.

Le numéro de port par défaut est 5000.

Procédure

Afin de dépister des messages de pulsation entre CVP et VRU PIM, il y a plusieurs étapes qui doivent être exécutées.

Step1. Du côté VRU PIM, augmentez le niveau des suivis. Le niveau de suivi par défaut n'affiche aucun message de pulsation.

Si vous activez seulement des suivis de heartbeat (*heartbeat * /on), les logs n'afficheront pas l'intercation de heartbeat entre CVP et VRU PIM, vous voient seulement ceci :

suivi pg2A-pim1 de 18:58:00:552 : PIMActiveHeartbeat

suivi pg2A-pim1 de 18:58:05:536 : PIMActiveHeartbeat

suivi pg2A-pim1 de 18:58:10:536 : PIMActiveHeartbeat

suivi pg2A-pim1 de 18:58:15:537 : PIMActiveHeartbeat

suivi pg2A-pim1 de 18:58:20:537 : PIMActiveHeartbeat

Mais si vous avez le *heartbeat de suivi * /on et *session de suivi * /on il vous affiche réellement que les messages dans le VRU PIM se connecte avec les numéros de séquence :

Voici l'exemple sur la façon dont activer les suivis VRU PIM :

```
C:\icn\ins98\PG1A\logfiles>procmon ins98 pg1a pin2
18:35:56 Trace: EMI Creating Mutex Global\IMIConnect_DisconnectLock
>>>>trace *session* /on
>>>>trace *heartbeat* /on
>>>>
```

Avertissement : Augmentez le niveau des suivis dans un envorment de production peut degradade les performances du système.

Il y a 3 messages qui un échange complet de pulsation.

```
18:59:05:538 pg2A-pim1 Trace: PG->VRU: Heartbeat Req (= Message Type 5); Message Length 4 bytes
Invoke ID: (2857109196) aa4c06cc
```

```
18:59:05:538 pg2A-pim1 Trace: VRU->PG: Heartbeat Conf (= Message Type 6); Message Length 4 bytes
Invoke ID: (2857109196) aa4c06cc
```

```
18:59:05:538 pg2A-pim1 Trace: PIMActiveHeartbeat
```

Et 5 secondes plus tard :

18:59:10:538 pg2A-pim1 Trace: PG->VRU: **Heartbeat Req** (= Message Type 5); Message Length 4 bytes
Invoke ID: (2857109197) aa4c06cd

18:59:10:538 pg2A-pim1 Trace: VRU->PG: **Heartbeat Conf** (= Message Type 6); Message Length 4 bytes
Invoke ID: (2857109197) aa4c06cd

18:59:10:538 pg2A-pim1 Trace: **PIMActiveHeartbeat**

Et toutes les 5 secondes ensuite cela

18:59:15:538 pg2A-pim1 Trace: PG->VRU: **Heartbeat Req** (= Message Type 5); Message Length 4 bytes
Invoke ID: (2857109198) aa4c06ce

18:59:15:538 pg2A-pim1 Trace: VRU->PG: **Heartbeat Conf** (= Message Type 6); Message Length 4 bytes
Invoke ID: (2857109198) aa4c06ce

18:59:15:538 pg2A-pim1 Trace: **PIMActiveHeartbeat**

18:59:20:538 pg2A-pim1 Trace: PG->VRU: **Heartbeat Req** (= Message Type 5); Message Length 4 bytes
Invoke ID: (2857109199) aa4c06cf

18:59:20:538 pg2A-pim1 Trace: VRU->PG: **Heartbeat Conf** (= Message Type 6); Message Length 4 bytes
Invoke ID: (2857109199) aa4c06cf

18:59:20:538 pg2A-pim1 Trace: **PIMActiveHeartbeat**

Step2. Suivis de Wireshark de capture.

Les messages de log VRU PIM sont assez typiquement de voir l'interaction de heartbeat entre les deux composants. Cependant, dans quelques scénarios les captures de Wireshark sont nécessaires.

Voici un extrait des suivis d'un Wireshark pris du CVP Callserver.

Puisque les deux périphériques parlent sur le port 5000, les suivis sont filtrés par **tcp.port==5000**

Les paquets de pulsation dans le Wireshark sont de 66 octets pour (pousser, ACK) et puis de 54 à 60 octets pour l'ACK.

Les données réelles de keepalive dans les messages de pousser qui sont envoyés de CVP Callserver à VRU PIM et sont vice-versa seulement 12 octets. Suivant les indications de l'image

No.	Time	Source	Destination	Protocol	Length	Info
34	2012-12-01 18:59:05.536897	192.168.0.25	192.168.0.27	TCP	66	opencore > complex-main [PSH, ACK] Seq=3382468379 Ack=1858767277 win=65079 Len=12
35	2012-12-01 18:59:05.536897	192.168.0.27	192.168.0.25	TCP	66	complex-main > opencore [PSH, ACK] Seq=1858767277 Ack=3382468391 win=64887 Len=12
38	2012-12-01 18:59:07.645295	192.168.0.25	192.168.0.27	TCP	54	opencore > complex-main [ACK] Seq=3382468391 Ack=1858767289 win=65067 Len=0
83	2012-12-01 18:59:10.517089	192.168.0.25	192.168.0.27	TCP	66	opencore > complex-main [PSH, ACK] Seq=3382468391 Ack=1858767289 win=65067 Len=12
84	2012-12-01 18:59:10.517089	192.168.0.27	192.168.0.25	TCP	66	complex-main > opencore [PSH, ACK] Seq=1858767289 Ack=3382468403 win=64875 Len=12
87	2012-12-01 18:59:10.5174784	192.168.0.25	192.168.0.27	TCP	54	opencore > complex-main [ACK] Seq=3382468403 Ack=1858767301 win=65055 Len=0
114	2012-12-01 18:59:14.208964	192.168.0.27	192.168.0.25	TCP	150	complex-main > opencore [PSH, ACK] Seq=1858767301 Ack=3382468403 win=64875 Len=12
117	2012-12-01 18:59:14.396464	192.168.0.25	192.168.0.27	TCP	54	opencore > complex-main [ACK] Seq=3382468403 Ack=1858767397 win=64959 Len=0
148	2012-12-01 18:59:15.537089	192.168.0.25	192.168.0.27	TCP	66	opencore > complex-main [PSH, ACK] Seq=3382468403 Ack=1858767397 win=64959 Len=12
149	2012-12-01 18:59:15.537089	192.168.0.27	192.168.0.25	TCP	66	complex-main > opencore [PSH, ACK] Seq=1858767397 Ack=3382468415 win=64863 Len=12
152	2012-12-01 18:59:15.724081	192.168.0.25	192.168.0.27	TCP	54	opencore > complex-main [ACK] Seq=3382468415 Ack=1858767409 win=64947 Len=0
199	2012-12-01 18:59:20.537281	192.168.0.25	192.168.0.27	TCP	66	opencore > complex-main [PSH, ACK] Seq=3382468415 Ack=1858767409 win=64947 Len=12
200	2012-12-01 18:59:20.537281	192.168.0.27	192.168.0.25	TCP	66	complex-main > opencore [PSH, ACK] Seq=1858767409 Ack=3382468427 win=64855 Len=12
201	2012-12-01 18:59:20.731570	192.168.0.25	192.168.0.27	TCP	54	opencore > complex-main [ACK] Seq=3382468427 Ack=1858767421 win=64935 Len=0

Prenez un paquet. comme exemple, choisissez le paquet 34. Les matches de ce paquet avec le timestamp(18:59:05) des premiers messages de pulsation vus dans le VRU PIM se connecte plus tôt.

Développez juste la partie données de la trame :

```

# Frame 34: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
# Ethernet II, Src: Vmware_96:10:ea (00:50:56:96:10:ea), Dst: Vmware_96:6c:e1 (00:50:56:96:6c:e1)
# Internet Protocol Version 4, Src: 192.168.0.25 (192.168.0.25), Dst: 192.168.0.27 (192.168.0.27)
# Transmission Control Protocol, Src Port: opencore (4089), Dst Port: complex-main (5000), Seq: 3382468379, Ack: 1858767277, Len: 12
# Data (12 bytes)
  Data: 0000000400000005aa4c06cc
    [Length: 12]

```

Vous pouvez voir les correspondances des données **aa4c06cc** avec l'invocation numéroté dans les suivis VRU PIM.

Les paquets de pulsation qui suivent auraient alors le nombre **aa4c06cd**, **aa4c06ce** et ainsi de suite.