

5G IMSおよびデータUPFノードでのハードウェアメンテナンスの実行

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[UPFとは](#)

[VPC-SIとは何ですか。](#)

[KVMハイパーバイザとは何ですか。](#)

[ICSRとは](#)

[問題](#)

[メンテナンスの手順](#)

概要

このドキュメントでは、IP Multimedia Subsystem(IMS)ノードおよびData User Plane Function(UPF)ノードでメンテナンスアクティビティを実行する手順について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- 5G-UPF
- Redundancy Configuration Manager(RCM)
- 仮想パケットコア(VPC) – シングルインスタンス(SI)
- カーネルベースの仮想マシン(KVM)ハイパーバイザ

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- 加入者マイクロサービスインフラストラクチャ(SMI)2020.02.2.35
- Star OS 21.22

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、初期(デフォルト)設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

UPFとは

ユーザプレーンインターフェイス(UPF)は、5Gコアネットワーク(5GC)のネットワーク機能(NF)の1つです。5Gアーキテクチャでは、パケットのルーティングと転送、パケット検査、QoSの処理、およびデータネットワーク(DN)を相互接続するための外部PDUセッションを担当します。

VPC-SIとは何ですか。

VPC-SIは、StarOSを実行する物理Cisco ASR 5500シャーシの動作を、市販のオフザシェルフ(COTS)サーバ上で実行できる単一の仮想マシン(VM)に統合します。各VPC-SI VMは独立したStarOSインスタンスとして動作し、物理シャーシの管理およびセッションプロセス機能を組み込みます。

KVMハイパーバイザとは何ですか。

カーネルベースの仮想マシン(KVM)は、Linuxに組み込まれているオープンソースの仮想化テクノロジーです。具体的には、KVMによってLinuxをハイパーバイザに変え、ホストマシンがゲストまたは仮想マシン(VM)と呼ばれる複数の独立した仮想環境を実行できるようにします。

ICSRとは

Interchassis Session Recovery(ICSR)は、別途ライセンスが必要なライセンス供与されたシスコの機能です。この機能は、加入者サービスを中断することなく、継続的なコールプロセスに対して可能な限り高い可用性を提供します。ICSRを使用すると、冗長性を確保するためにゲートウェイを設定できます。ゲートウェイに障害が発生した場合、ICSRではセッションを障害の周囲に透過的にルーティングできるため、ユーザエクスペリエンスが維持されます。ICSRでは、セッション情報と状態も保持されます。

問題

ハードウェア障害やソフトウェア/ファームウェアのアップグレードなどのハードウェアメンテナンスには、サーバのダウンタイムが必要です。この手順に従って、UPFベアメタルサーバでメンテナンスを実行し、サービスを適切に切り替えてUPFアプリケーションの不要なダウンタイムを回避する方法を実行する必要があります。

メンテナンスの手順

UPFノードは、KVMハイパーバイザでホストされるStarOS VMです。1台のKVMハイパーバイザが2つのVMインスタンスをホストします。IMS UPFには1:1の冗長性があり、すべてのアクティブインスタンスにスタンバイインスタンスがあります。冗長性を処理するために、Session Redundancy Protocol(SRP)とともにICSRを使用します。SRPは、ICSRシャーシ間でhelloメッセージを交換するために使用されます。また、アクティブ/スタンバイシャーシ(チェックポイントデータ)間でセッション状態情報を交換します。完全なサブスクライバセッション情報は、SRPリンクを介して、コールリカバリレコード(CRR)の形式でアクティブシャーシからスタンバ

イシャーシに送信されます。

KVMノードにログインし、KVM `virsh`コマンドを使用してVMインスタンスをリストします。

```
cloud-user@podname-upf-ims-kvmnode-1:~$ sudo virsh list --all
```

```
Id Name State
```

```
-----  
1 imsupf01 running
```

```
4 imsupf10 running
```

```
cloud-user@podname-upf-ims-kvmnode-1:~$
```

UPFインスタンスにログインし、シャーシのステータスを確認します。

```
[local]imsupf10# show srp info
```

```
Friday July 22 15:50:24 UTC 2022
```

```
Service Redundancy Protocol:
```

```
-----  
Context: srp
```

```
Local Address: 10.x.x.74
```

```
Chassis State: Standby
```

```
Chassis Mode: Backup
```

```
Chassis Priority: 2
```

```
Local Tiebreaker: 02-7E-35-53-F9-F1
```

```
Route-Modifier: 9
```

```
Peer Remote Address: 10.x.x.73
```

```
Peer State: Active
```

```
Peer Mode: Primary
```

```
Peer Priority: 1
```

```
Peer Tiebreaker: 02-11-59-73-87-35
```

```
Peer Route-Modifier: 8
```

```
Last Hello Message received: Fri Jul 22 15:50:21 2022 (3 seconds ago)
```

```
Peer Configuration Validation: Complete
```

```
Last Peer Configuration Error: None
```

```
Last Peer Configuration Event: Fri Jul 22 15:50:22 2022 (2 seconds ago)
```

```
Last Validate Switchover Status: None
```

```
Connection State: Connected
```

```
[local]imsupf01# show srp info
```

```
Friday July 22 15:31:20 UTC 2022
```

```
Service Redundancy Protocol:
```

```
-----  
Context: srp
```

```
Local Address: 10.x.x.66
```

```
Chassis State: Active
```

```
Chassis Mode: Backup
```

```
Chassis Priority: 2
```

```
Local Tiebreaker: 02-7C-1A-62-FA-3C
```

```
Route-Modifier: 5
```

```
Peer Remote Address: 10.x.x.65
```

```
Peer State: Standby
```

```
Peer Mode: Primary
```

```
Peer Priority: 1
```

```
Peer Tiebreaker: 02-87-33-98-6D-08
```

```
Peer Route-Modifier: 6
```

```
Last Hello Message received: Fri Jul 22 15:31:20 2022 (1 seconds ago)
```

```
Peer Configuration Validation: Complete
```

```
Last Peer Configuration Error: None
```

```
Last Peer Configuration Event: Fri Jul 22 15:20:13 2022 (668 seconds ago)
```

Last Validate Switchover Status: None
Connection State: **Connected**

IMS UPFのアクティブ/スタンバイICSRペアの回線数が同じかどうかを確認します。

```
Active node
# show configuration | grep -n -E "^end$"
Thursday July 21 07:30:17 UTC 2022
14960:end
```

```
Standby Node
# show configuration | grep -n -E "^end$"
Thursday July 21 07:31:02 UTC 2022
14959:end
```

アクティブUPFでのSRPスイッチオーバーの前に、SRP sessmgrがアクティブ接続状態であるかどうかを確認し、保留中のアクティブ状態がないことを確認します。

```
[local]imsupf01# show srp checkpoint statistics active
Thursday July 21 07:38:04 UTC 2022
Number of Sessmgrs: 20
Sessmgrs in Active-Connected state: 20
Sessmgrs in Standby-Connected state: 0
Sessmgrs in Pending-Active state: 0
```

スタンバイUPFでのSRPスイッチオーバーの前に、SRP sessmgrがアクティブ接続状態であるかどうかを確認し、pending-active状態がないことを確認します

```
[local]imsupf02# show srp checkpoint statistics active
Thursday July 21 07:40:03 UTC 2022
Number of Sessmgrs: 20
Sessmgrs in Active-Connected state: 0
Sessmgrs in Standby-Connected state: 20
Sessmgrs in Pending-Active state: 0
```

これら2つの状態のいずれかがActive状態の場合、切り替えの前に最初に次のタスクを実行する必要があります。

```
[upf-ims]# save config /flash/xxx_production.cfg. --> Replace xxx with the desired name of the config
```

```
[upf-ims]# srp validate-configuration
[upf-ims]# srp validate-switchover
```

VMをシャットダウンする前に、アクティブインスタンスがスタンバイに切り替わり、サブスライバが正常に切り替わることを確認する必要があります。インスタンスがすでにスタンバイ状態になっている場合は、アクションは必要ありません。インスタンスがアクティブな場合は、強調表示されている値を確認し、スタンバイがテイクオーバーできる状態であることを確認します。

アクティブなUPFインスタンスの現在のサブスライバを確認します。

```
[local]imsupf01# show subscribers data-rate summary
Friday July 22 16:01:37 UTC 2022
```

```
Total Subscribers : 175024
Active : 175024 Dormant : 0
```

アクティブ・ インスタンスをスタンバイに切り替えます。

```
[context-name]<hostname># srp initiate-switchover
```

スタンバイのステータスを確認します。スタンバイは現在アクティブになり、サブスクライバセッションも新しいアクティブインスタンスに移動します。両方のVMインスタンスがスタンバイ状態であるため、サーバメンテナンスのためにダウンすることをお勧めします。指定されたvirshコマンドを使用して、VMインスタンスをシャットダウンし、ステータスを確認します。

```
cloud-user@podname-upf-ims-kvmnode-1:~$ sudo virsh shutdown imsupf01
```

```
Domain imsupf01 is being shutdown
```

```
cloud-user@podname-upf-ims-kvmnode-1:~$ sudo virsh shutdown imsupf10
```

```
Domain imsupf10 is being shutdown
```

```
cloud-user@podname-upf-ims-kvmnode-1:~$ sudo virsh list --all
```

```
Id Name State
```

```
-----  
1 imsupf01 shut off
```

```
4 imsupf10 shut off
```

```
cloud-user@podname-upf-ims-kvmnode-1:~$
```

メンテナンス後にサーバが復帰すると、VMは自動的に起動します。UPFインスタンスはスタンバイ状態のままです。指定されたコマンドで確認します。

```
[local]imsupf10# show srp info
```

```
Friday July 22 15:50:24 UTC 2022
```

```
Service Redundancy Protocol:
```

```
-----  
Context: srp
```

```
Local Address: 10.x.x.74
```

```
Chassis State: Standby
```

```
Chassis Mode: Backup
```

```
Chassis Priority: 2
```

```
Local Tiebreaker: 02-7E-35-53-F9-F1
```

```
Route-Modifier: 9
```

```
Peer Remote Address: 10.x.x.73
```

```
Peer State: Active
```

```
Peer Mode: Primary
```

```
Peer Priority: 1
```

```
Peer Tiebreaker: 02-11-59-73-87-35
```

```
Peer Route-Modifier: 8
```

```
Last Hello Message received: Fri Jul 22 15:50:21 2022 (3 seconds ago)
```

```
Peer Configuration Validation: Complete
```

```
Last Peer Configuration Error: None
```

```
Last Peer Configuration Event: Fri Jul 22 15:50:22 2022 (2 seconds ago)
```

```
Last Validate Switchover Status: None
```

```
Connection State: Connected
```

データUPFは、N:Mの冗長性を持つRCMを使用します。NはアクティブUPFの数を表し、10未満で、Mは冗長グループ内のスタンバイUPFの数を表します。RCMは、StarOSベースのユーザプレーン機能(UPF)に冗長性を提供するシスコ独自のノードまたはネットワーク機能(NF s)です。すべてのアクティブUPFから必要なすべてのセッション情報を保存またはミラーリングします。スイッチオーバートリガーでは、共通の場所から適切なセッションデータを受信するために1つのスタンバイUPFが選択されます。RCMはVM上のK3クラスタで実行されます。オペレーションセンターがRCMノードを設定します。

データUPFノードもIMS UPFノードと同じです。唯一の違いは、RCM冗長管理です。

KVMノードのVMステータスをチェックします。

```
cloud-user@podname-upf-data-kvmnode-1:~$ sudo virsh list --all
```

```
Id Name State
```

```
-----  
1 dataupf20 running  
2 dataupf11 running
```

```
cloud-user@podname-upf-data-kvmnode-1:~$
```

UPFインスタンスにログインした後、RCMの冗長性ステータスを確認します。インスタンスがすでにスタンバイ状態になっている場合は、アクションは必要ありません。アクティブな場合は、正常にスタンバイに切り替える必要があります。

```
[local]dataupf11# show rcm info  
Friday July 22 17:23:17 UTC 2022  
Redundancy Configuration Module:
```

```
-----  
Context: rcm  
Bind Address: 10.x.x.75  
Chassis State: Active  
Session State: SockActive  
Route-Modifier: 26  
RCM Controller Address: 10.x.x.163  
RCM Controller Port: 9200  
RCM Controller Connection State: Connected  
Ready To Connect: Yes  
Management IP Address: 10.x.x.149  
Host ID: DATAUPF15  
SSH IP Address: 10.x.x.158 (Activated)  
SSH IP Installation: Enabled
```

```
[local]dataupf11#
```

すべてのsessmgrsがActive-connected状態であることを確認します。

```
local]dataupf11# show rcm checkpoint statistics active  
Thursday July 21 07:47:03 UTC 2022  
Number of Sessmgrs: 22  
Sessmgrs in Active-Connected state: 22  
Sessmgrs in Standby-Connected state: 0  
Sessmgrs in Pending-Active state: 0
```

顧客情報アンケート(CIQ)から対応するRCMノードを特定し、RCMのステータスを確認します。RCMの切り替えは、マスターノードからのみ実行できます。マスターRCMにログインしていることを確認します。

```
[podname-aio-1/dcrm01] rcm# rcm show-status  
message :
```

```
{"status": "MASTER"}
```

```
[podname-aio-1/dcrm01] rcm#
```

次のコマンドを使用して、アクティブおよびスタンバイUPFノードを検索します(出力を省略)。

```
[podname-aio-1/dcrm01] rcm# rcm show-statistics controller
```

```
message :
```

```
{  
"keepalive_version": "e7386cb81b1fefc3396dfd1d528e0d2a27de80d5de6a78364caf938a0d2149b6",  
"keepalive_timeout": "20s",  
"num_groups": 2,  
}
```

```

"groups": [
{
"groupid": 1,
"endpoints_configured": 7,
"standby_configured": 1,
"pause_switchover": false,
"active": 6,
"standby": 1,
"endpoints": [
{
"endpoint": "10.x.x.75",
"bfd_status": "STATE_UP",
"upf_registered": true,
"upf_connected": true,
"upf_state_received": "UpfMsgState_Active",
"bfd_state": "BFDDState_UP",
"upf_state": "UPFState_Active",
"route_modifier": 26,
"pool_received": true,
"echo_received": 142354,
"management_ip": "10.x.x.149",
"host_id": "DATAUPF15",
"ssh_ip": "10.x.x.158",
"force_nso_registration": false

....

....

{
"endpoint": "10.x.x.77",
"bfd_status": "STATE_UP",
"upf_registered": true,
"upf_connected": true,
"upf_state_received": "UpfMsgState_Standby",
"bfd_state": "BFDDState_UP",
"upf_state": "UPFState_Standby",
"route_modifier": 50,
"pool_received": false,
"echo_received": 3673,
"management_ip": "10.x.x.153",
"host_id": "",
"ssh_ip": "10.x.x.186",
"force_nso_registration": false
},

```

管理IPを使用してスタンバイUPFインスタンスにログインし、ステータスを確認します

```

[local]dataupf13# show rcm info
Friday July 22 17:36:04 UTC 2022
Redundancy Configuration Module:
-----
Context: rcm
Bind Address: 10.x.x.77
Chassis State: Standby
Session State: SockStandby
Route-Modifier: 50
RCM Controller Address: 10.x.x.163
RCM Controller Port: 9200
RCM Controller Connection State: Connected
Ready To Connect: Yes
Management IP Address: 10.x.x.153
Host ID:

```

```
SSH IP Address: 10.x.x.186 (Activated)
SSH IP Installation: Enabled
```

```
[local]dataupf13#
```

確認後、アクティブからスタンバイにグレースフルスイッチオーバーします。管理IPを使用していることを確認します。

```
[podname-aio-1/dcrm01] rcm# rcm switchover-mgmt-ip source 10.x.x.149 destination 10.x.x.153
```

注：スイッチオーバー後に新しいActive UP sessmgrがSERVER状態のままになる場合。シスコのテクニカルサポートに連絡してください。問題のあるインスタンスの場合は、sessmgrを強制終了する必要があるため、適切なCLIENTソケット状態でRCMに再接続し、回復します。すべてのsessmgrがCLIENT状態である必要があります。特定のコマンド(隠しモード)で確認します。

```
# show session subsystem facility sessmgr all debug-info | grep -E "SessMgr|Mode:"
```

```
Thursday July 21 07:56:26 UTC 2022
SessMgr: Instance 5000
Mode: UNKNOWN State: SRP_SESS_STATE SOCK_ACTIVE
SessMgr Activity Detected: FALSE
SessMgr: Instance 22
Mode: CLIENT State: SRP_SESS_STATE SOCK_ACTIVE
SessMgr Activity Detected: TRUE
SessMgr: Instance 21
Mode: CLIENT State: SRP_SESS_STATE SOCK_ACTIVE
SessMgr Activity Detected: TRUE
```

すべてのsessmgrがアクティブで準備完了状態であることを確認します。

```
# show rcm checkpoint statistics verbose
```

```
Thursday July 21 07:52:29 UTC 2022
smgr state peer recovery pre-alloc chk-point rcvd chk-point sent
inst conn records calls full micro full micro
```

```
-----
1 Actv Ready 0 0 1731 68120 3107912 409200665
2 Actv Ready 0 0 1794 70019 3060062 408647685
3 Actv Ready 0 0 1753 68793 3078531 406227415
4 Actv Ready 0 0 1744 67585 3080952 410218643
5 Actv Ready 0 0 1749 69155 3096067 404944553
6 Actv Ready 0 0 1741 68805 3067392 407133464
7 Actv Ready 0 0 1744 67963 3084023 406772101
8 Actv Ready 0 0 1748 68702 3009558 408073589
9 Actv Ready 0 0 1736 68169 3030624 405679108
10 Actv Ready 0 0 1707 67386 3071592 406000628
11 Actv Ready 0 0 1738 68086 3052899 407991476
12 Actv Ready 0 0 1720 68500 3102045 408803079
13 Actv Ready 0 0 1772 69683 3082235 406426650
14 Actv Ready 0 0 1727 66900 2873736 392352402
15 Actv Ready 0 0 1739 68465 3032395 409603844
16 Actv Ready 0 0 1756 69221 3063447 411445527
17 Actv Ready 0 0 1755 68708 3051573 406333047
18 Actv Ready 0 0 1698 66328 3066983 407320405
19 Actv Ready 0 0 1736 68030 3037073 408215965
20 Actv Ready 0 0 1733 67873 3069116 405634816
21 Actv Ready 0 0 1763 69259 3074942 409802455
22 Actv Ready 0 0 1748 68228 3051222 406470380
```

サブスクリバが新しいスタンバイに移動されることを確認します。

```
[local]dataupf11# show subscribers data-rate summary
```


Friday July 22 17:40:18 UTC 2022

Total Subscribers : 62259
Active : 62259 Dormant : 0

両方のインスタンスがスタンバイの場合、**virsh**コマンドを使用してVMをKVMからシャットダウンできます。

```
cloud-user@podname-upf-data-kvmnode-1:~$ sudo virsh shutdown dataupf20
```

```
Domain dataupf20 is being shutdown
```

```
cloud-user@podname-upf-data-kvmnode-1:~$ sudo virsh shutdown dataupf11
```

```
Domain dataupf11 is being shutdown
```

```
cloud-user@podname-upf-data-kvmnode-1:~$ sudo virsh list --all
```

```
Id Name State
```

```
-----  
1 dataupf20 shut off
```

```
4 dataupf11 shut off
```

```
cloud-user@podname-upf-data-kvmnode-1:~$
```

VMをシャットダウンすると、メンテナンスのためにKVMノード (物理サーバ) を停止できます。完了したら、サーバを起動します。VMは自動的に起動します。UPFインスタンスは自動的にスタンバイになります。所定のコマンドで同じことを確認します。

```
cloud-user@podname-upf-data-kvmnode-1:~$ sudo virsh list --all
```

```
Id Name State
```

```
-----  
1 dataupf20 running
```

```
2 dataupf11 running
```

```
cloud-user@podname-upf-data-kvmnode-1:~$
```

```
[local]dataupf11# show rcm info
```

```
Friday July 22 17:36:04 UTC 2022
```

```
Redundancy Configuration Module:
```

```
-----  
Context: rcm
```

```
Bind Address: 10.x.x.77
```

```
Chassis State: Standby
```

```
Session State: SockStandby
```

```
Route-Modifier: 50
```

```
RCM Controller Address: 10.x.x.163
```

```
RCM Controller Port: 9200
```

```
RCM Controller Connection State: Connected
```

```
Ready To Connect: Yes
```

```
Management IP Address: 10.x.x.153
```

```
Host ID:
```

```
SSH IP Address: 10.x.x.186 (Activated)
```

```
SSH IP Installation: Enabled
```

```
[local]dataupf13#
```

RCMノードでは、rcmコントローラはスタンバイUPFを「pending standby」として表示できます。スタンバイに変換するには、最大で15 ~ 20分かかります。次のコマンドで同じことを確認します (出力を省略)。

```
[podname-aio-1/dcrm01] rcm# rcm show-statistics controller
```

```
message :
```

```
{
```

```
"keepalive_version": "e7386cb81b1fefc3396dfd1d528e0d2a27de80d5de6a78364caf938a0d2149b6",
```

```
"keepalive_timeout": "20s",
"num_groups": 2,
"groups": [
{
"groupid": 1,
"endpoints_configured": 7,
"standby_configured": 1,
"pause_switchover": false,
"active": 6,
"standby": 1,
"endpoints": [
....
....
{
"endpoint": "10.x.x.77",
"bfd_status": "STATE_UP",
"upf_registered": true,
"upf_connected": true,
"upf_state_received": "UpfMsgState_Standby",
"bfd_state": "BFDState_UP",
"upf_state": "UPFState_Standby",
"route_modifier": 50,
"pool_received": false,
"echo_received": 3673,
"management_ip": "10.x.x.153",
"host_id": "",
"ssh_ip": "10.x.x.186",
"force_nso_registration": false
},
```