



Cisco Catalyst 9800 シリーズ ワイヤレスコントローラ (Cisco IOS XE Bengaluru 17.6) 高可用性 SSO 導入ガイド

初版発行日：2021 年 7 月 31 日

はじめに

目次

はじめに.....	5
概要.....	5
機能の説明と機能の動作.....	5
サポートされるプラットフォーム.....	6
SSO の前提条件.....	6
Cisco Catalyst C9800-40-K9 および C9800-80-K9 ワイヤレスコントローラの SSO.....	6
C9800-L、C9800-40 および C9800-80 ワイヤレスコントローラ HA SSO の物理的な接続性.....	8
SSO に RJ-45 RP ポートを使用する C9800-L ワイヤレスコントローラの接続.....	8
SSO に RJ-45 RP ポートを使用する C9800-40 および 9800-80 ワイヤレスコントローラの 接続.....	8
SSO に SFP ギガビット RP ポートを使用する C9800-40 および 9800-80 ワイヤレスコントローラの接続.....	9
上流側スイッチへの C9800 ワイヤレスコントローラ HA ペアの接続.....	9
オプション 1: バックツーバック接続 RP を使用する単一の VSS スイッチ (またはスタック、VSL ペア、モジュラスイッチ).....	10
オプション 2: アップストリーム経由の RP を使用する単一の VSS スイッチ (またはスタック、VSL ペア、モジュラスイッチ).....	10
オプション 3: HSRP を使用するデュアル分散スイッチ.....	11
上流側スイッチへの C9800 ワイヤレスコントローラ HA ペアの接続 (リリース 17.1 以降).....	11
ESXi、KVM、Hyper-V で実行する Cisco Catalyst C9800-CL での SSO.....	12
GUI を使用した高可用性 SSO の設定.....	13
CLI を使用した高可用性 SSO の設定.....	14
モビリティ MAC.....	14
アクティブおよびスタンバイの選出プロセス.....	15
HA SSO ペア構成の状態遷移.....	16
HA ペアのモニタリング.....	18
CLI からの HA ペアのモニタリング.....	20
冗長状態の確認.....	21
スタンバイ ワイヤレスコントローラ コンソールへのアクセス.....	23
スイッチオーバー機能.....	24

はじめに

プロセス障害によるスイッチオーバー.....	24
電源障害によるスイッチオーバー.....	24
手動スイッチオーバー.....	25
フェールオーバー プロセス.....	25
アクティブ ワイヤレスコントローラ.....	25
スタンバイ ワイヤレスコントローラ.....	25
AP とクライアント SSO の状態同期の確認.....	26
SSO フェールオーバー時間のメトリック.....	27
リダンダンシー マネジメント インターフェイス.....	27
WebUI を使用したリダンダンシー マネジメント インターフェイスの構成.....	27
プログラムによる RMI IP の設定.....	28
RMI IPv4 を使用したデュアルスタックのサポート.....	29
RMI IPv6 を使用したデュアルスタックのサポート.....	29
ピアタイムアウトの設定.....	29
CLI を使用したリダンダンシー マネジメント インターフェイスの構成.....	30
RMI と RP 構成の確認.....	31
RMI と RP ペアリングの組み合わせ.....	31
アップグレードと以前の HA 構成を使用しない HA ペアリング.....	31
すでにペアリングされているコントローラのアップグレード.....	32
ダウングレード.....	32
デフォルト ゲートウェイ チェック.....	33
ゲートウェイ障害検出間隔の設定.....	34
デフォルト ゲートウェイ チェックの WebUI 設定.....	36
デフォルト ゲートウェイ チェックの CLI 設定.....	36
システムおよびネットワークの障害処理.....	37
HA ペアリング解除動作.....	42
SSO ペアでの LACP、PAGP サポート.....	44
サポートされる LACP、PAGP トポロジ.....	44
マルチシャーシリンク集約グループ.....	45
サポートされるマルチシャーシ LAG トポロジ.....	45
サポート対象プラットフォーム :	46
サポートされる LAG ポートのグループ化.....	46

はじめに

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例	47
ワイヤレスコントローラの設定.....	47
VSS の設定	51
HA セットアップでのコントローラの置き換え	54
SSO ハイブリッド導入による N+1	54
RMI を使用したスタンバイのモニタリング	55
スタンバイモニタリング CLI.....	55
スタンバイ モニタリング プログラム インターフェイス.....	57
RMI IPv4 への SSH 接続を使用してスタンバイコントローラをモニタする手順	57
RESTCONF を使用するスタンバイモニタリングのコマンド	59
リリース 17.5 のスタンバイモニタリング	59
SNMP を使用したアクティブを通したスタンバイコントローラのインターフェイス ステータス	68

はじめに

はじめに

高可用性は、稼働中のネットワークのダウンタイムを最小限に抑えるためのワイヤレスコントローラの要件でした。このドキュメントでは、アクセス ポイントおよびクライアントのステートフル スイッチオーバー (AP およびクライアント SSO) のサポートに関連する Catalyst 9800 ワイヤレスコントローラの操作と設定の理論について説明します。Catalyst 9800 ワイヤレスコントローラは小規模から大規模まで、さまざまな拡張性の目標を持つ複数のプラットフォームで稼働可能な次世代のワイヤレスコントローラです。Catalyst 9800 ワイヤレスコントローラ (C9800-L、C9800-40、C9800-80 および C9800-CL) の物理アプライアンスおよび仮想クラウドプラットフォームで AP とクライアント SSO がサポートされています。基盤となっている SSO 機能は、セットアップ プロセスでの違いはいくつかあるものの、すべてのプラットフォームで同じです。

概要

ワイヤレスコントローラでの高可用性 SSO 機能は、アクセス ポイントがアクティブ ワイヤレスコントローラとの CAPWAP トンネルを確立し、アクティブ ワイヤレスコントローラが AP のミラー コピーとクライアント データベースをスタンバイ ワイヤレスコントローラと共有できるようにします。AP が [Discovery] 状態になることはなく、クライアントはアクティブ ワイヤレスコントローラに障害が発生した場合も切断されず、スタンバイ コントローラがアクティブ ワイヤレスコントローラとしてネットワークを引き継ぎます。AP とアクティブ状態のワイヤレスコントローラ間では一度に 1 つの CAPWAP トンネルのみが維持されます。

リリース 16.10 では、フル アクセス ポイントとクライアントのステートフル スイッチ オーバーがサポートされています。クライアント SSO がサポートされるのは、すでに認証および DHCP フェーズが完了し、トラフィックの送信を始めたクライアントです。SSO クライアントによって、ワイヤレスコントローラにクライアントが関連付けられたときか、またはクライアントのパラメータが変更されたときに、クライアント情報はスタンバイ ワイヤレスコントローラに同期します。完全に認証済みのクライアント (実行状態のクライアントなど) はスタンバイへと同期され、スイッチオーバー時のクライアントの再関連付けが回避されます。これによりクライアントと AP のフェールオーバーはシームレスになり、クライアント サービスのダウンタイム ゼロと SSID の停止ゼロが実現します。Catalyst 9800 ワイヤレスコントローラへの AP およびクライアントの SSO サポートの追加は、プライマリ サイトでのボックス フェールオーバー、ネットワーク フェールオーバー、または停電により発生する可能性がある障害状態によってワイヤレス ネットワークの大規模なダウンタイムを削減することが全体目標となっています。

機能の説明と機能の動作

コントロールプレーン アクティビティはすべて、アクティブ ユニットとスタンバイ ユニット間で一元管理され、同期されます。アクティブ コントローラが、すべての制御と管理通信を一元管理します。ネットワーク制御データのトラフィックは、スタンバイ ユニットからアクティブ ユニットに透過的に切り替えられ、一元処理されます。

一括設定や増分設定は 2 つのコントローラ間で実行時に同期され、両方のコントローラが管理インターフェイス上で同じ IP アドレスを共有します。実行状態のアクセス ポイントの CAPWAP 状態も、アクティブ ワイヤレスコントローラからホットスタンバイ ワイヤレスコントローラに同期されるため、アクティブ ワイヤレスコントローラに障害が発生したときにアクセス ポイントをステートフルに切り替えることができます。アクティブ ワイヤレスコントローラに障害が発生した場合に AP が [Discovery] 状態になることはなく、スタンバイ ワイヤレスコントローラがアクティブ ワイヤレスコントローラを引き継いでネットワークにサービスを提供します。

サポートされるプラットフォーム

2 つのユニットは、専用の RP ポート（物理的な銅線ポートまたは光ファイバポート）または VM の仮想インターフェイスを介してピア接続を形成します。アクティブ/スタンバイはブート時に、最も高い優先順位（優先順位の範囲は 1 ~ 2）または優先順位が同じ場合、最も小さい MAC に基づいて選択されます。デフォルトでは、C9800 の プライオリティは 1 です。HA ペアが形成されると、すべての構成と AP およびクライアントのデータベースがアクティブとスタンバイの間で同期されます。アクティブの構成は、自動的にスタンバイに同期されます。スタンバイは、RP リンク上のキープアライブを介してアクティブを継続的にモニタしています。アクティブが使用できなくなると、スタンバイがアクティブのロールを担います。これは、スタンバイがワイヤレス管理 IP アドレスを取得することを伝える Gratuitous ARP メッセージアドバタイジングをネットワークに送信することによって実行されます。すべての構成とデータベースがすでに同期されているため、サービスを中断することなくスタンバイが引き継ぐことができます。

SSO にはプリエンプト機能はありません。つまり、以前アクティブだったワイヤレスコントローラが動作を再開しても、アクティブ ワイヤレスコントローラのロールを取り戻すことはありません。ただし、その状態を現在アクティブ ワイヤレスコントローラとネゴシエートし、ホットスタンバイ状態に移行します。

サポートされるプラットフォーム

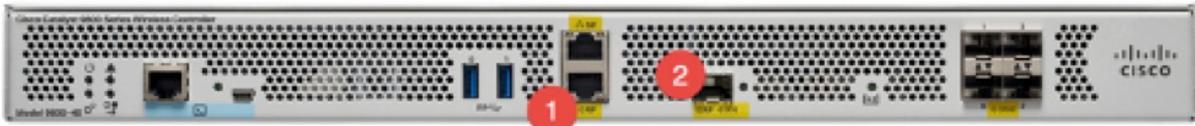
- Cisco Catalyst C9800-40 ワイヤレスコントローラ
- Cisco Catalyst C9800-80 ワイヤレスコントローラ
- Cisco Catalyst C9800-CL ワイヤレスコントローラ
- Cisco Catalyst C9800-L ワイヤレスコントローラ

SSO の前提条件

- HA ペアは同じフォームファクタの 2 つのワイヤレスコントローラ間でのみ形成できます。
- 9800-LC と 9800-LF の間の HA は形成できません。
- 銅線 RP とファイバ RP 間の HA は形成できません。
- HA ペアを形成するには、両方のコントローラが同じソフトウェアバージョンを実行している必要があります。
- 最大 RP リンク遅延 = 80 ms RTT、最小帯域幅 = 60 Mbps、最小 MTU = 1500

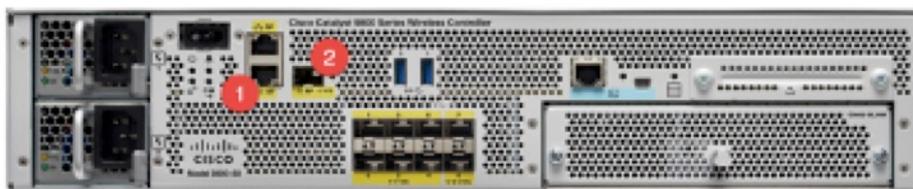
Cisco Catalyst C9800-40-K9 および C9800-80-K9 ワイヤレスコントローラの SSO

Cisco C9800-40-K9 ワイヤレスコントローラは拡張可能な高性能のワイヤレスコントローラであり、最大 2,000 アクセス ポイント、32,000 クライアントまで拡張できます。コントローラには 4 つの 10G データポートがあり、スループットは 40G です。



1	RP-RJ-45 1G 冗長イーサネットポート。	2	ギガビット SFP RP ポート
---	--------------------------	---	------------------

Cisco C9800-80-K9 ワイヤレスコントローラは、ラックユニットの空間を 2 つ占有し、着脱可能なモジュールスロットと 8 つの組み込み 10GE/1GE インターフェイスをサポートする 100G ワイヤレスコントローラです。



1	RP-RJ-45 1G 冗長イーサネットポート。	2	ギガビット SFP RP ポート
---	--------------------------	---	------------------

上の図に示すように、C9800-40-K9 ワイヤレスコントローラと C9800-80-K9 ワイヤレスコントローラの両方に 2 つの RP ポートが備わっています。

■ RJ-45 イーサネット冗長ポート

■ SFP ギガビット冗長ポート

両方の冗長ポートが接続されている場合は、次のようになります。

■ 両方の冗長ポートが同時に接続されている場合は、SFP ギガビット イーサネット ポートが優先されます。

■ RJ-45 ポートと SFP ギガビット RP ポート間では HA はサポートされません。

■ RP ポートでは、シスコのサポート対象の SFP (GLC-LH-SMD と GLC-SX-MMD) のみがサポートされています。

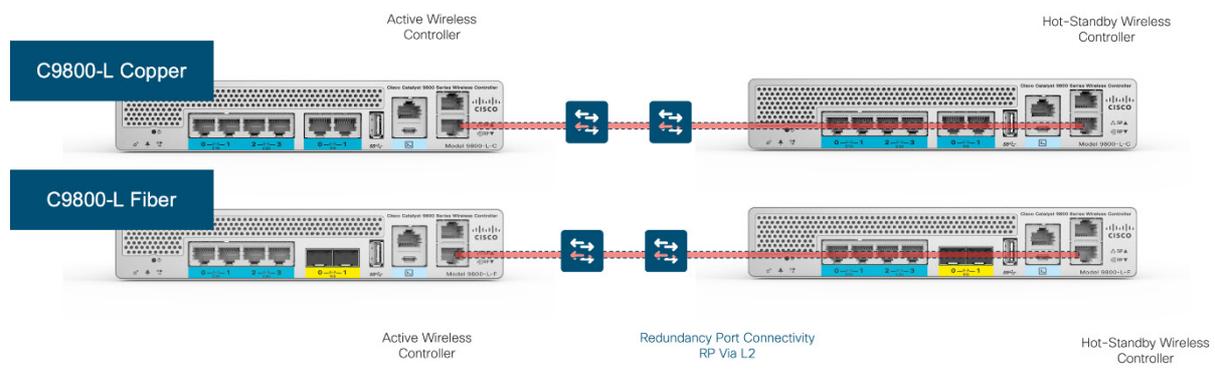
■ 10G SFP-10G-SR は RP ポートではサポートされません。

■ RJ-45 経由で HA リンクが動作している場合、HA ポート上の SFP 間にリンクがない場合でも、それらを挿入しないでください。物理レベルの検出であるため SFP が優先され、HA がダウンする原因となります。

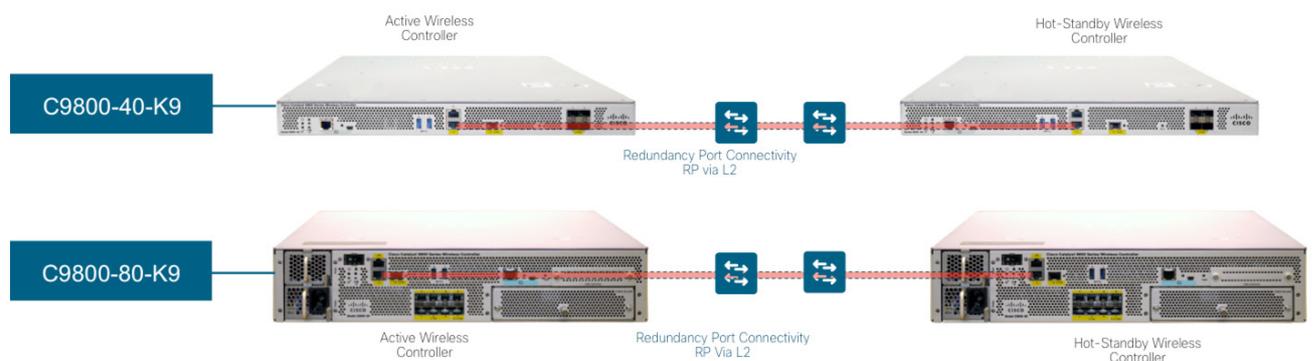
C9800-L、C9800-40 および C9800-80 ワイヤレスコントローラ HA SSO の物理的な接続性

HA ペアには常にアクティブ コントローラ 1 台と、スタンバイ コントローラ 1 台が必要です。アクティブ コントローラが使用できなくなると、スタンバイがアクティブのロールを担います。アクティブ ワイヤレスコントローラはワイヤレス情報のすべてを作成、更新し、その情報を常にスタンバイ コントローラと同期します。アクティブ ワイヤレスコントローラに障害が発生すると、スタンバイ ワイヤレスコントローラがアクティブ ワイヤレスコントローラのロールを担い、HA ペアの動作を維持します。アクセス ポイントとクライアントは、アクティブからスタンバイへのスイッチオーバーのときも接続を維持し続けます。

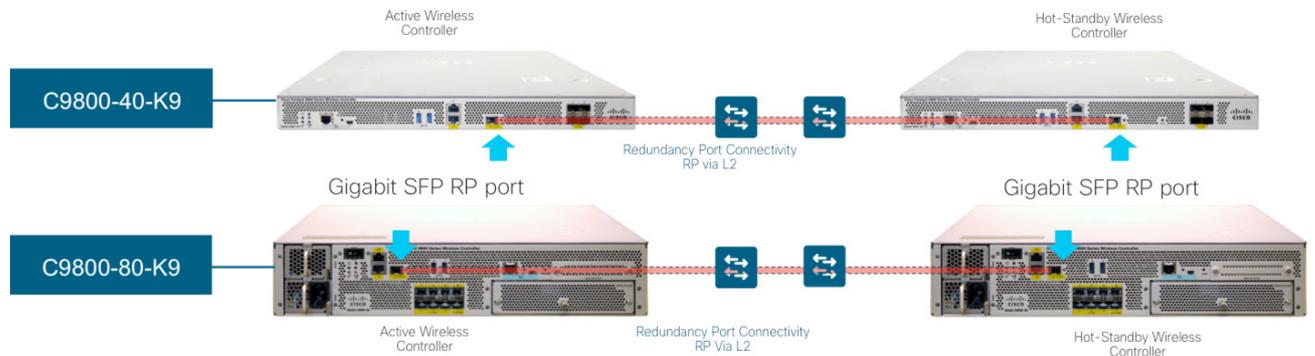
SSO に RJ-45 RP ポートを使用する C9800-L ワイヤレスコントローラの接続



SSO に RJ-45 RP ポートを使用する C9800-40 および 9800-80 ワイヤレスコントローラの接続



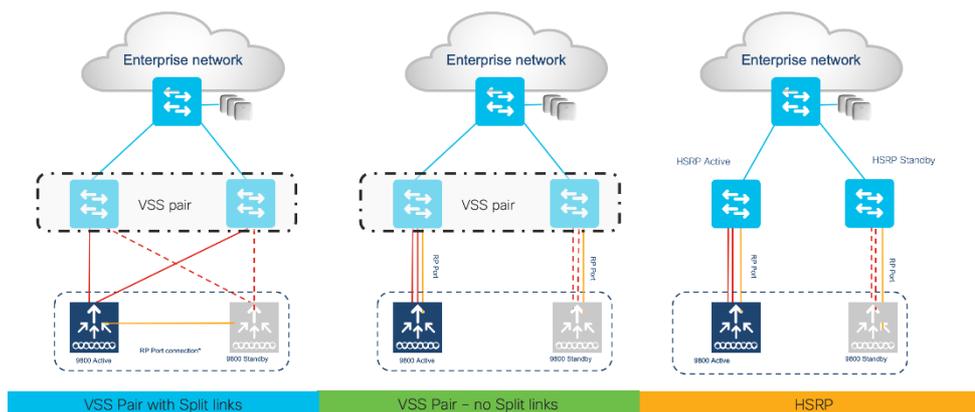
SSO に SFP ギガビット RP ポートを使用する C9800-40 および 9800-80 ワイヤレスコントローラの接続



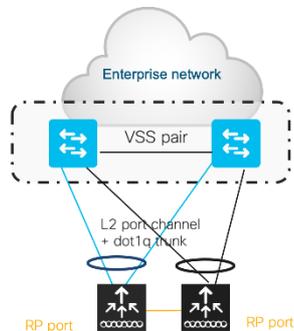
上流側スイッチへの C9800 ワイヤレスコントローラ HA ペアの接続

17.1 より前は、ネットワークへのアップストリーム接続に関して次のトポロジがサポートされていました。

1. 分割リンクとバックツーバック接続の RP を使用してアップストリーム VSS ペアに接続された SSO ペア。
2. ゲートウェイダウンのシナリオを検出するために、スイッチのアップストリームセット経由接続の RP を使用してアップストリーム VSS ペアに接続された SSO ペア。
3. ゲートウェイダウンのシナリオを検出するために、アップストリーム HSRP アクティブ/スタンバイおよびスイッチのアップストリームセット経由接続の RP に接続された SSO ペア。



オプション 1：バックツーバック接続 RP を使用する単一の VSS スイッチ（またはスタック、VSL ペア、モジュラスイッチ）

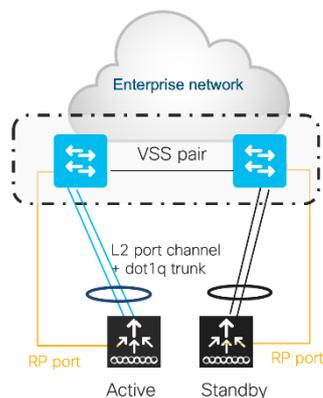


各ボックスに単一の L2 ポートチャンネルが作成され、これにより dot1q で複数の VLAN を伝送できるようになります。VSS ペア全体に HA ペアのアップリンクを分散し、RP をバックツーバックで接続します（間に L2 ネットワークはありません）。スイッチが ARP および MAC テーブルエントリに関して拡張できることを確認します。

このトポロジをお勧めします。

注：HA SSO トポロジでは、モードが ON の LAG のみがサポートされます。

オプション 2：アップストリーム経由の RP を使用する単一の VSS スイッチ（またはスタック、VSL ペア、モジュラスイッチ）

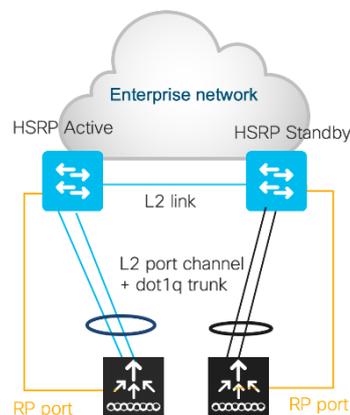


このトポロジでは、各ボックスに単一の L2 ポートチャンネルが作成されます。dot1q で複数の VLAN が伝送でき、同じ方法でスタンバイを接続できます。スイッチが ARP および MAC テーブルエントリに関して拡張できることを確認します。

重要：このトポロジでは、リンクは VSS スタック全体に分散されません。RP ポートは、バックツーバックではなく、アップリンクと同じ VSS またはスタックメンバに接続します。

注：HA SSO トポロジでは、モードが ON の LAG のみがサポートされます。

オプション 3 : HSRP を使用するデュアル分散スイッチ



このトポロジでは、各ボックスに単一の L2 ポートチャネルが作成されます。dot1q で複数の VLAN が伝送でき、同じ方法でスタンバイを接続できます。スイッチが ARP および MAC テーブルエントリに関して拡張できることを確認します。

重要 : RP ポートは、バックツーバックではなく、アップリンクと同じディストリビューション スイッチに接続します。

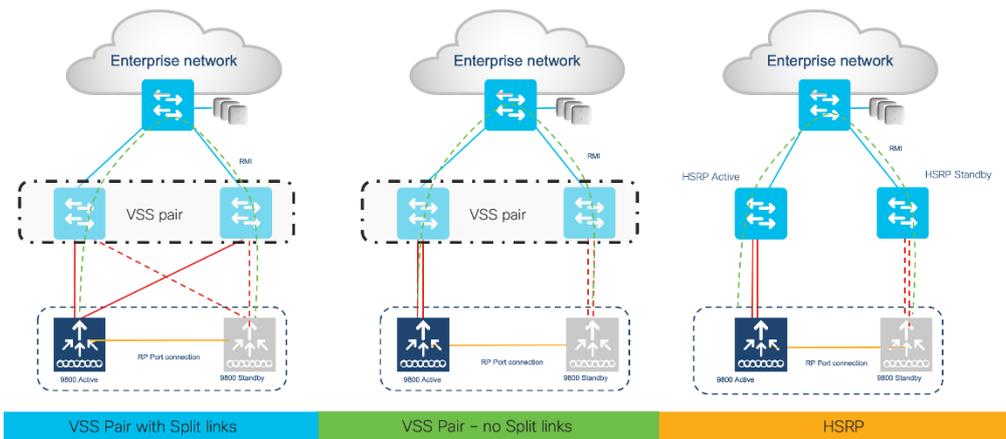
注 : リリース 17.1 より前の HA SSO トポロジでは、モードが ON の LAG のみがサポートされます。17.1 では、LACP と PAGP もサポートされます。詳細については、「SSO ペアでの LACP、PAGP サポート」セクションを参照してください。

上流側スイッチへの C9800 ワイヤレスコントローラ HA ペアの接続 (リリース 17.1 以降)

リリース 17.1 で使用できる RMI およびデフォルト ゲートウェイ チェック機能のオプションについては、次のトポロジがサポートされ、推奨されるようになりました。

1. 分割リンクとバックツーバック接続の RP を使用してアップストリーム VSS ペアに接続された SSO ペア。
2. アップストリーム VSS ペアとバックツーバック接続の RP に接続された SSO ペア。
3. アップストリーム HSRP アクティブ/スタンバイおよびバックツーバック接続の RP に接続された SSO ペア。

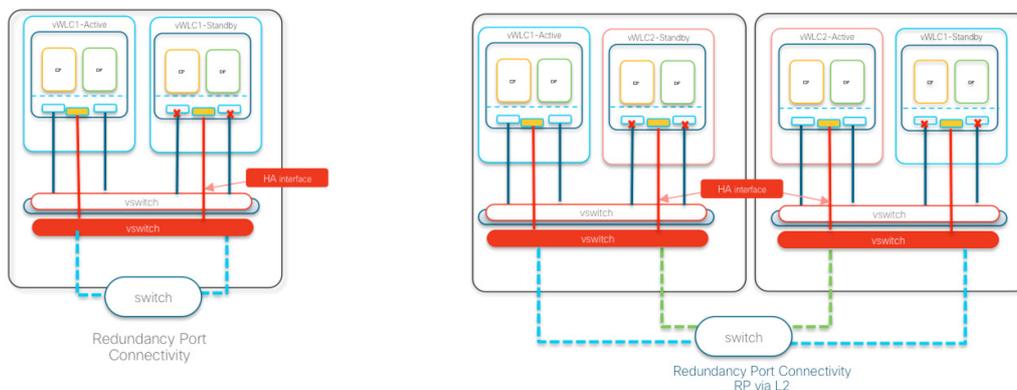
ESXi、KVM、Hyper-V で実行する Cisco Catalyst C9800-CL での SSO



注：高速コンバージェンスのためにアップリンクスイッチの PortFast トランクを構成する場合は、CLI 「spanning-tree port type edge trunk」または「spanning-tree portfast trunk」を使用することをお勧めします。

ESXi、KVM、Hyper-V で実行する Cisco Catalyst C9800-CL での SSO

仮想 Catalyst 9800 ワイヤレスコントローラは、単一サーバー セットアップまたはデュアル サーバー セットアップで HA ペアとして導入できます。



左の図に、同じサーバーに接続された冗長ポートを示します。

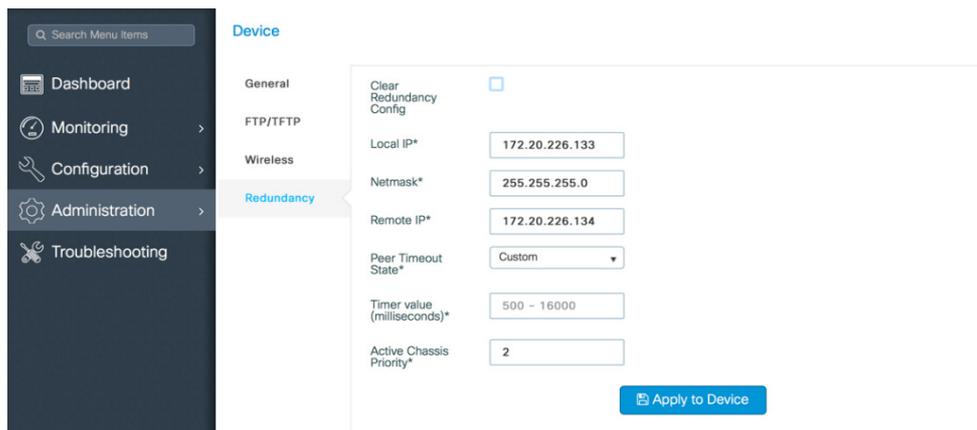
右の図に、別のサーバーに接続された冗長ポート L2 を示します。

9800-CL で HA ペアを形成するには、同じインターフェイス番号 (Gig3 など) を使用する必要があります。テンプレートのスケールも一致する必要があります。HyperV、VMware ESXi、KVM で 9800-CL の SSO はサポートされています。

GUI を使用した高可用性 SSO の設定

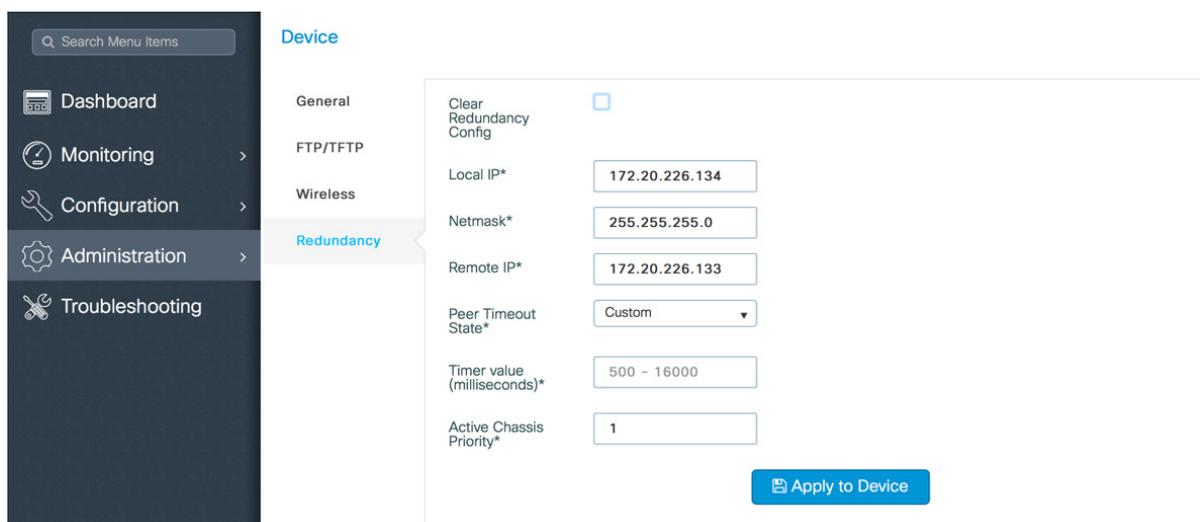
デバイスの冗長性は、[Administration] > [Device] > [Redundancy] ページで構成できます。

アクティブ コントローラで、スタンバイ コントローラよりも高い優先度を設定します。高い方のプライオリティ値を持つワイヤレスコントローラが、アクティブ/スタンバイ選出プロセス時にアクティブとして選択されます。リモート IP はスタンバイ コントローラの冗長ポート IP の IP アドレスです。



注: このページは、リリース 17.1 以降変更され、RMI を使用した HA ペアを構成するオプションが追加されました。構成用に更新された画面を確認するには、「リダンダンシー マネジメント インターフェイス」セクションを参照してください。

スタンバイコントローラでは、リモート IP がアクティブコントローラの冗長ポート IP に設定されます。



- 1) ローカル IP とリモート IP の両方の IP アドレスは同じサブネット内にある必要があります。
- 2) 169.254.X.X/16 サブネットを使用することをお勧めします。最後の 2 つのオクテットは、管理インターフェイスの最後の 2 つのオクテットから取得できます。
- 3) 9800 WLC の不具合により、RP ポートに 10.10.10.x/24 サブネットを使用しないでください。

CLI を使用した高可用性 SSO の設定

[Clear Redundancy config] は、SSO 構成をクリアし、コントローラをスタンドアロンモードに戻します。

注：リリース 17.1 以降では、リダンダンシー マネジメント インターフェイス (RMI) を使用して HA を構成することをお勧めします。RMI を使用する構成を確認するには、「リダンダンシー マネジメント インターフェイス」セクションを参照してください。

CLI を使用した高可用性 SSO の設定

■ **仮想 Catalyst 9800 ワイヤレスコントローラの 2 つの仮想 Catalyst 9800 ワイヤレスコントローラのインスタンスそれぞれで次のコマンドを使用して高可用性 SSO を有効にします。**

```
chassis redundancy ha-interface <RP interface> local-ip <local IP> <local IP subnet>
remote-ip <remote IP>
```

例：

仮想 Catalyst 9800 ワイヤレスコントローラ インスタンス - 1：

```
chassis redundancy ha-interface Gig 3 local-ip 172.23.174.85 /24 remote-
ip 172.23.174.86
```

仮想 Catalyst 9800 ワイヤレスコントローラ インスタンス - 2：

```
chassis redundancy ha-interface Gig 3 local-ip 172.23.174.86 /24 remote-
ip 172.23.174.85
```

■ **C9800-40 ワイヤレスコントローラと C9800-80 ワイヤレスコントローラの 2 台のワイヤレスコントローラ ユニットそれぞれで次のコマンドを使用して高可用性 SSO を有効にします。**

```
chassis redundancy ha-interface local-ip <local IP> <local IP subnet> remote-
ip <remote IP>
```

CLI から reload コマンドを実行して両方のワイヤレスコントローラをリロードします。

注：リリース 17.1 以降では、リダンダンシー マネジメント インターフェイス (RMI) を使用して HA を構成することをお勧めします。RMI を使用する構成を確認するには、「リダンダンシー マネジメント インターフェイス」セクションを参照してください。

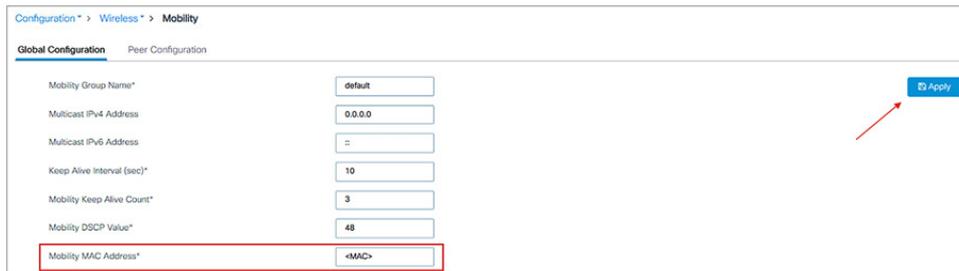
モビリティ MAC

ワイヤレスモビリティ MAC は、モビリティ通信に使用される MAC アドレスです。SSO シナリオでは、ワイヤレスモビリティ MAC アドレスを明示的に設定してください。設定しないと、SSO 後にモビリティトンネルがダウンします。SSO ペアのモビリティ MAC アドレスは、次のいずれかで設定できます。

- 各スタンドアロンコントローラで SSO ペアを形成する前に設定する。ソフトウェアリリース 16.12.3 より前ではこの方法をお勧めします。
- SSO ペアが形成された後のアクティブコントローラで設定する。

アクティブおよびスタンバイの選出プロセス

モビリティ MAC アドレスを設定する場合、GUI を使用できます。



The screenshot shows the 'Configuration' page for 'Wireless' > 'Mobility'. Under 'Global Configuration', there are several fields: 'Mobility Group Name*' (default), 'Multicast IPv4 Address' (0.0.0.0), 'Multicast IPv6 Address' (::), 'Keep Alive Interval (sec)*' (10), 'Mobility Keep Alive Count*' (3), and 'Mobility DSCP Value*' (48). The 'Mobility MAC Address*' field is highlighted with a red box and contains '<MAC>'. A red arrow points to the 'Apply' button in the top right corner.

アドレスを入力したら、[Apply] をクリックします。

注： GUI の MAC アドレスは、ワイヤレス管理インターフェイスから自動的に取得されますが、他の有効な MAC アドレスも使用できます。

CLI では、次のコマンドを使用します。

```
C9800#wireless mobility mac-address <MAC>
```

アクティブおよびスタンバイの選出プロセス

アクティブ C9800 ワイヤレスコントローラは、次のイベントのいずれかが発生しない限り、アクティブ コントローラとしてのロールが維持されます。

- ワイヤレスコントローラ HA ペアがリセットされる。
- アクティブ ワイヤレスコントローラが HA ペアから削除される。
- アクティブ ワイヤレスコントローラがリセットされるか、または電源オフになる。
- アクティブ ワイヤレスコントローラがフェールする。

次の要因のいずれかに基づき、次に示す順序でアクティブ ワイヤレスコントローラが選出されるか、または再選出されます。

1. 現在のアクティブ ワイヤレスコントローラであるワイヤレスコントローラ。
2. 最も高いプライオリティ値を持つワイヤレスコントローラ。

注：アクティブコントローラにしたいワイヤレスコントローラ C9800 に最も高いプライオリティ値を割り当てることをお勧めします。これにより、再選出が発生した場合にアクティブ コントローラとしてそのコントローラが選出されるようにします。

スイッチ プライオリティ値の設定

```
chassis chassis -number priority new-priority-number
```

chassis-number：シャーシ番号とシャーシの新しいプライオリティを指定します。シャーシ番号の範囲は 1 ~ 2 です。シャーシの再番号付けコマンドでは、再起動が必要です。

プライオリティ値の範囲は 1 ~ 2 です。スタックのプライオリティ 2 はプライマリ、プライオリティ 1 はスタンバイです。

HA SSO ペア構成の状態遷移

例

```
wireless controller#chassis 1 priority 2
```

show chassis ユーザー EXEC コマンドを使用すると、現在のプライオリティ値を表示できません。新しいプライオリティ値はすぐに有効となりますが、現在のアクティブ コントローラには影響しません。新しいプライオリティ値は、現在のアクティブ ワイヤレスコントローラまたは HA 冗長ペアをリロードしたときに新しいアクティブ コントローラとしてどのコントローラを選出するかの決定に役立ちます。

3. 起動時間が最短のワイヤレスコントローラ。
4. 最小の MAC アドレスを持つワイヤレスコントローラ。

シャーシの HA LED を使用して現在のアクティブ コントローラを識別できます。

HA SSO ペア構成の状態遷移

1. 非冗長モードのアクティブ ワイヤレスコントローラ

```
TLV(0): T=9, L=29, V=KEY_TLV_PACKAGE_COMPATIBILITY
FRU Key detected
TLV(1): T=9, L=11, V=FRU_RP_TYPE
found package fru type FRU_RP_TYPE
TLV(2): T=9, L=24, V=KEY_TLV_PACKAGE_BOOTARCH
ARCH Key detected
TLV(3): T=9, L=14, V=ARCH_i686_TYPE
found package arch type ARCH_i686_TYPE
TLV(4): T=9, L=20, V=KEY_TLV_BOARD_COMPAT
TLV(5): T=9, L=15, V=BOARD_qwlc_TYPE
TLV(6): T=9, L=24, V=KEY_TLV_CRYPTO_KEYSTRING
TLV(7): T=9, L=4, V=none
TLV(8): T=9, L=11, V=CW_BEGIN=$$
TLV(9): T=9, L=16, V=CW_FAMILY=$qwlc$
TLV(10): T=9, L=78, V=CW_IMAGE=$qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20180310_120257.SSA.bin$
TLV(11): T=9, L=19, V=CW_VERSION=$16.9.1$
TLV(12): T=9, L=52, V=CW_DESCRIPTION=$Cisco IOS Software, IOS-XE Software$
TLV(13): T=9, L=9, V=CW_END=$$
found DIGISIGN TLV type 12 length = 388

RSA Signed DEVELOPMENT Image Signature Verification Successful.
Validating subpackage signatures: addr=0x6e13e3f8, size=01c789ed

initramfs_size: 0x1c78dcd - 0x4b0a38 - 0x3e0 = 0x17c7fb5
Image validated
Booting image with bootparam="root=/dev/rw console=tty1 max_loop=64 pciehp.pciehp_force pcie_ports=native SR_BOOT=tftp://172.25.140.118/auto/
tftpboot/maahmed/qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20180310_120257.SSA.bin rd_start=0xaf06e000 rd_size=0x17c7fb5 pkg_start=0x33f68000
pkg_size=0x3a1d4000 bdfinfo_start=0xcd42b000 bdfinfo_size=0x35c34"
May 3 15:13:22.585: %BOOT-0-DRV_LOADFAIL: R0/0: binos: Failed to load driver modprobe ( /usr/binos/conf/driver_common.sh: line 99: indigovor:
command not found )
May 3 15:13:43.295: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May 3 15:13:45.742: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
Waiting for remote chassis to join
```

2. HA ペアリングのためのスタンバイ挿入

```
Chassis number is 1
All chassis in the stack have been discovered. Accelerating discovery
May 3 15:13:46.276: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May 3 15:13:46.877: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May 3 15:13:48.852: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May 3 15:13:53.654: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
May 3 15:13:56.934: %PMAN-3-PROC_EMPTY_EXEC_FILE: R0/0: pvp: Empty executable used for process bt_logger
```

Restricted Rights Legend

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR sec. 52.227-19 and subparagraph (c) (1) (iii) of the Rights in Technical Data and Computer Software clause at DFARS sec. 252.227-7013.

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706

HA SSO ペア構成の状態遷移

3. HA 同期処理中

```

directory.
*May 3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 2 on Chassis 1 is down
*May 3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 1 on Chassis 1 is up
*May 3 15:13:52.681: %STACKMGR-6-STACK_LINK_CHANGE: Chassis 2 R0/0: stack_mgr: Stack port 2 on Chassis 1 is up
*May 3 15:13:52.682: %STACKMGR-6-CHASSIS_ADDED: Chassis 2 R0/0: stack_mgr: Chassis 2 has been added to the stack.
*May 3 15:13:52.682: %STACKMGR-6-CHASSIS_ADDED: Chassis 2 R0/0: stack_mgr: Chassis 2 has been added to the stack.
*May 3 15:13:52.682: %STACKMGR-6-ACTIVE_ELECTED: Chassis 2 R0/0: stack_mgr: Chassis 1 has been elected ACTIVE.
*May 3 15:13:52.682: %CMRP-3-PFU_MISSING: Chassis 2 R0/0: cmand: The platform does not detect a power supply in slot 1
*May 3 15:14:41.704: %SYS-4-FREEMEMWARNING: SIP0/0: Free Memory has dropped below warning threshold.
*May 3 15:14:46.405: %SYS-6-BOOTTIME: Time taken to reboot after reload = 1073 seconds
*May 3 15:14:46.761: %PNP-6-PNP_DISCOVERY_STOPPED: PnP Discovery stopped (Startup Config Present)
*May 3 15:14:46.789: %SPA_OIR-6-ONLINECARD: SPA (BUILT-IN-4X10G) online in subslot 0/0
*May 3 15:14:46.883: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/0, link down due to local fault
*May 3 15:14:46.937: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/1, link down due to local fault
*May 3 15:14:46.977: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/2, link down due to local fault
*May 3 15:14:47.040: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/3, link down due to local fault
*May 3 15:14:48.780: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/1, changed state to down
*May 3 15:14:48.783: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/2, changed state to down
*May 3 15:14:48.784: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/3, changed state to down
*May 3 15:14:49.217: %IOSXE_SPA-6-UPDOWN: Interface TenGigabitEthernet0/0/0, link down due to remote fault
*May 3 15:14:49.032: %LINK-3-UPDOWN: SIP0/0: Interface TenGigabitEthernet0/0/0, changed state to down
*May 3 15:14:49.652: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to down
*May 3 15:14:50.043: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
*May 3 15:14:51.043: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed state to up
*May 3 15:14:54.229: %PKI-2-NON_AUTHORITATIVE_CLOCK: PKI functions can not be initialized until an authoritative time source, like NTP, can be obtained.
*May 3 15:14:55.456: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to up
*May 3 15:14:55.458: %LINK-3-UPDOWN: Interface Vlan1, changed state to down
*May 3 15:14:55.456: %LINK-3-UPDOWN: SIP0/0: Interface TenGigabitEthernet0/0/0, changed state to up
*May 3 15:14:57.892: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet0/0/0, changed state to up
*May 3 15:14:58.891: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
*May 3 15:14:59.892: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
*May 3 15:15:09.367: %IOSXE_REDUNDANCY-6-PEER: Active detected chassis 2 as standby.
*May 3 15:15:09.365: %STACKMGR-6-STANDBY_ELECTED: Chassis 1 R0/0: stack_mgr: Chassis 2 has been elected STANDBY.
*May 3 15:15:09.652: %PMAN-3-PROC_EMPTY_EXEC_FILE: Chassis 2 R0/0: pvp: Empty executable used for process bt_logger
*May 3 15:15:10.140: %PMAN-3-PROC_EMPTY_EXEC_FILE: Chassis 2 R0/0: pvp: Empty executable used for process ngiolite
*May 3 15:15:14.751: %IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P0 inserted
*May 3 15:15:14.754: %IOSXE_PEM-6-PEMOK: The PEM in slot P0 is functioning properly
*May 3 15:15:14.754: %IOSXE_PEM-6-INSPEM_FM: PEM/FM slot P2 inserted
*May 3 15:15:14.758: %IOSXE_PEM-6-PEMOK: The PEM in slot P2 is functioning properly
WLC>

WLC#
*May 3 15:15:39.434: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby insertion (raw-event=PEER_FOUND(4))
*May 3 15:15:39.434: %REDUNDANCY-5-PEER_MONITOR_EVENT: Active detected a standby insertion (raw-event=PEER_REDUNDANCY_STATE_CHANGE(5))
*May 3 15:15:41.404: % Redundancy mode change to SSO
*May 3 15:15:41.404: %VOICE_HA-7-STATUS: NONE->SSO; SSO mode will not take effect until after a platform reload.
*May 3 15:15:44.413: Syncing vlan database
*May 3 15:15:44.436: Vlan Database sync done from bootflash:vlan.dat to stby-bootflash:vlan.dat (1464 bytes)
WLC#
WLC#
WLC#
WLC#
WLC#
WLC#
WLC#show chas
WLC#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8769 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair

Chassis# Role Mac Address Priority H/W Current State IP
-----
*1 Active 00a3.8e23.8769 1 V02 Ready 172.20.226.134
2 Standby 00a3.8e23.8909 1 V02 HA sync in progress 172.20.226.133
    
```

4. SSO での端末状態

```

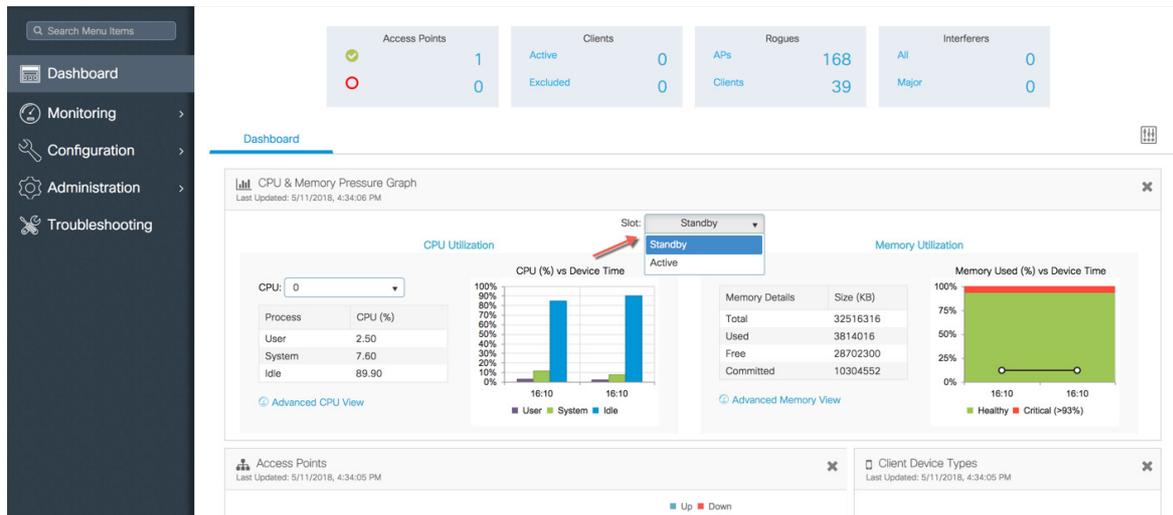
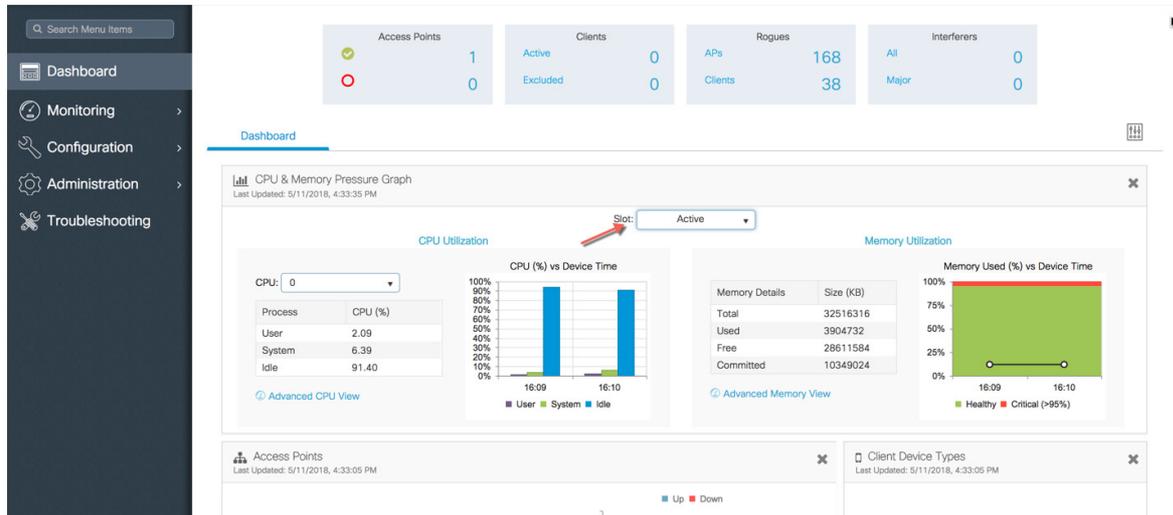
*May 3 15:18:46.564: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEEDED: Bulk Sync succeed
*May 3 15:18:46.565: %VOICE_HA-7-STATUS: VOICE HA bulk sync done.
*May 3 15:18:47.565: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
WLC#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8769 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair

Chassis# Role Mac Address Priority H/W Current State IP
-----
*1 Active 00a3.8e23.8769 1 V02 Ready
2 Standby 00a3.8e23.8909 1 V02 Ready
    
```

注：HA ペアの解消：HA 構成は、chassis clear コマンド後にリロードすることで無効にできます。

HA ペアのモニタリング

アクティブ ワイヤレスコントローラの管理 UI からアクティブとスタンバイの両方のシステムをモニタできます。これには、CPU およびメモリ使用率に関する情報と、詳細な CPU ビューおよびメモリビューが含まれています。



コントローラ Web UI で [Monitoring] > [System] > [Redundancy] に移動します。[Redundancy States] ページが表示されます。

The screenshot shows the 'Redundancy' page in the Cisco Catalyst 9800 Web UI. The left sidebar contains navigation options: Dashboard, Monitoring, Configuration, Administration, and Troubleshooting. The main content area is titled 'System' and includes tabs for Memory Utilization, CPU Utilization, Wireless Interface, Management Summary, and Redundancy. The Redundancy page displays a list of parameters and their values:

- My State: 13 -ACTIVE
- Peer State: 8 -STANDBY HOT
- Mode: Duplex
- Unit: Primary
- Unit ID: 2
- Redundancy Mode (Operational): sso
- Redundancy Mode (Configured): sso
- Redundancy State: sso
- Maintenance Mode: Disabled
- Manual Swact: enabled
- Communications: Up
- Client Count: 136
- Client_notification_TMR: 30000 milliseconds
- RF Debug Mask: 0x0
- Active UpTime: 21 hours, 55 minutes
- Standby UpTime: 21 hours, 48 minutes
- Standby Failures: 0
- Switchover System Experienced: 1

Below the parameters is a table with the following data:

Index	Previous Active	Current Active	Switch Over Time	Switch Over Reason
1	1	2	18:16:37 UTC Thu May 10 2018	user forced

パラメータ	説明
My State	<p>アクティブ CPU コントローラ モジュールの状態を表示します。表示される値は次のとおりです。</p> <p>Active</p> <p>Standby HOT</p> <p>Disable</p>
Peer State	<p>ピア (またはスタンバイ) CPU コントローラ モジュールの状態を表示します。表示される値は次のとおりです。</p> <p>Standby HOT</p> <p>Disable</p>
Mode	<p>冗長ピアの現在の状態を表示します。表示される値は次のとおりです。</p> <p>Simplex : 単一 CPU のコントローラモジュール。</p> <p>Duplex : 2 つの CPU が搭載されたコントローラモジュール。</p>
Unit ID	CPU コントローラ モジュールのユニット ID を表示します。

Redundancy Mode (Operational)	ユニットでサポートされる最新の動作冗長モードを表示します。
Redundancy Mode (Configured)	ユニットでサポートされる最新の設定冗長モードを表示します。
Redundancy State	ユニットの現在機能している冗長性状態を表示します。表示される値は次のとおりです。 SSO Not Redundant
Manual Swact	手動スイッチオーバーが有効になっているかどうかを表示します。
Communications	2 つのコントローラ間で通信がアップかダウンかを表示します。

同じページにスイッチオーバー履歴を表示します。次の表に、次のパラメータの説明を示します。

パラメータ	説明
Index	冗長ユニットのインデックス番号を表示します。
Previous Active	スイッチオーバーする前にアクティブだったコントローラを表示します。
Current Active	現在アクティブなコントローラを表示します。
Switch Over Time	スイッチオーバー発生時のシステム時刻を表示します。
Switch Over Reason	スイッチオーバーの理由を表示します。

CLI からの HA ペアのモニタリング

`show chassis` コマンドは HA ペアに関するサマリー情報を表示します。これには、MAC アドレス、ロール、スイッチプライオリティ、および冗長 HA ペア内の各ワイヤレスコントローラの現在の状態が含まれています。デフォルトでは、HA ペアのローカル MAC アドレスが最初に選択されたアクティブ コントローラの MAC アドレスになります。

冗長状態の確認

```
WLC#show chassis
```

```
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address  
Mac persistency wait time: Indefinite  
Local Redundancy Port Type: Twisted Pair
```

Chassis#	Role	Mac Address	Priority	H/W Version	Current State	IP
1	Standby	00a3.8e23.8760	1	V02	Ready	172.20.226.133
*2	Active	00a3.8e23.8900	1	V02	Ready	172.20.226.134

上に示すように、show chassis コマンドは、シャーシ番号に「*」を付加してコンソールで現在の C9800 ワイヤレスコントローラを示します。

冗長状態の確認

- 2 台のユニットの状態をモニタするには show redundancy コマンドを使用します。

```
wireless controller#show redundancy ?  
application      box 2 box application information  
clients           Redundancy Facility (RF) client list  
config-sync      Show Redundancy Config Sync status  
counters         Redundancy Facility (RF) operational counters  
domain           Specify the RF domain  
history          Redundancy Facility (RF) history  
idb-sync-history Redundancy Facility (RF) IDB sync history  
linecard-group   Line card redundancy group information  
rii              Display the redundancy interface identifier for Box to Box  
states          Redundancy Facility (RF) states  
switchover       Redundancy Facility (RF) switchover  
trace           Redundancy Facility (RF) trace  
|               Output modifiers  
<cr>           <cr>
```

- show redundancy コマンドは冗長システムと現在のプロセッサ情報を表示します。冗長システムの情報にはシステム稼働時間、スタンバイの障害、スイッチオーバーの理由、ハードウェアのモード、および設定され、動作している冗長モードが含まれます。表示される現在のプロセッサ情報には、イメージ バージョン、アクティブなロケーション、ソフトウェアの状態、ブート変数、設定登録値、および現在の状態での稼働時間などが含まれています。ピア プロセッサの情報は、アクティブ コントローラからのみ取得できます。

冗長状態の確認

```
WLC#show redundancy
Redundant System Information :
-----
    Available system uptime = 22 hours, 9 minutes
    Switchovers system experienced = 1
        Standby failures = 0
    Last switchover reason = user forced

    Hardware Mode = Duplex
    Configured Redundancy Mode = sso ←
    Operating Redundancy Mode = sso ←
    Maintenance Mode = Disabled
    Communications = Up

Current Processor Information :
-----
    Active Location = slot 2
    Current Software state = ACTIVE
    Uptime in current state = 21 hours, 43 minutes
    Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpr
e/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
    Copyright (c) 1986-2018 by Cisco Systems, Inc.
    Compiled Wed 09-May-18 06:35 by mcpre
    BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
    CONFIG_FILE =
    Configuration register = 0x2102

Peer Processor Information :
-----
    Standby Location = slot 1
    Current Software state = STANDBY HOT
    Uptime in current state = 21 hours, 35 minutes
    Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpr
e/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
    Copyright (c) 1986-2018 by Cisco Systems, Inc.
    Compiled Wed 09-May-18 06:35 by mcpre
    BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
    CONFIG_FILE =
    Configuration register = 0x2102
```

■ show redundancy states コマンドは、アクティブコントローラとスタンバイコントローラのすべての冗長状態を表示します。

```
WLC#show redundancy states ?
domain Specify the RF domain
| Output modifiers
<cr> <cr>

WLC#show redundancy states
my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 2

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
Redundancy State = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up

client count = 136
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0
```

■ 手動スイッチオーバーアクション (手動 Swact)、つまり、redundancy force-swact コマンドはスタンバイ ワイヤレスコントローラ上では実行できません。このコマンドはアクティブコントローラでのみ有効です。

スタンバイ ワイヤレスコントローラ コンソールへのアクセス

- スイッチオーバー履歴を表示するには、次のコマンドを使用します。

```
WLC#show redundancy switchover history
Index Previous Current Switchover Switchover
  active active reason time
-----
  1      1      2 user forced 18:16:37 UTC Thu May 10 2018
```

スタンバイ ワイヤレスコントローラ コンソールへのアクセス

アクティブ コントローラは、管理 IP アドレスを使用し、コンソール接続、Telnet、SSH、または Web ブラウザからアクセスできます。スタンバイ ワイヤレスコントローラでコンソールを使用するには、アクティブ Catalyst 9800 ワイヤレスコントローラから次のコマンドを実行します。

```
conf t
redundancy
main-cpu
standby console enable
```

スタンバイ コンソール上のプロンプトに「-stby」が追加され、次に示すようにスタンバイ ワイヤレスコントローラ コンソールが反映されます。

```
WLC-stby#show chassis
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
Chassis# Role Mac Address Priority H/W Current IP
-----
*1 Standby 00a3.8e23.8760 1 V02 Ready 0.0.0.0
 2 Active 00a3.8e23.8900 1 V02 Ready 0.0.0.0
```

注：上に示すように、show chassis コマンドは、シャーシ番号に「*」を付加してコンソールで現在の C9800 ワイヤレスコントローラを示します。この場合は、スタンバイ ユニットのコンソールです。

スイッチオーバー機能

```
WLC-stby>en
WLC-stby#show red
WLC-stby#show redun
WLC-stby#show redundancy
Redundant System Information :
-----
    Available system uptime = 22 hours, 2 minutes
Switchovers system experienced = 1

    Hardware Mode = Duplex
Configured Redundancy Mode = sso
Operating Redundancy Mode = sso
Maintenance Mode = Disabled
Communications = Up

Current Processor Information :
-----
    Standby Location = slot 1
    Current Software state = STANDBY HOT
    Uptime in current state = 21 hours, 29 minutes
    Image Version = Cisco IOS Software [Fuji], WLC9000 Software (X86_64_LINUX_IO
SD-UNIVERSALK9_WLC-M), Experimental Version 16.10.20180509:065558 [polaris_dev-/nobackup/mcpre/BLD-BLD_POLARIS_DEV_LATEST_20180509_073715 183]
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Wed 09-May-18 06:35 by mcpre
    BOOT = bootflash:qwlc-universalk9_wlc.BLD_POLARIS_DEV_LATEST_201805
09_073715.SSA.bin,1;
    CONFIG_FILE =
    Configuration register = 0x2102

Peer (slot: 2, state: ACTIVE) information is not available because this is the standby proces
sor
```

スイッチオーバー機能

プロセス障害によるスイッチオーバー

このタイプのスイッチオーバーは、アクティブ ユニットで実行中の主要プロセスに障害またはクラッシュが発生したときに行われます。このような障害が発生すると、アクティブ ユニットがリロードし、ホットスタンバイが引き継いで新しいアクティブ ユニットになります。障害が発生したシステムが起動すると、そのシステムはホットスタンバイ状態に移行します。スタンバイ ユニットがまだホットスタンバイの状態になっていない場合は、両方のユニットがリロードし SSO はできません。スタンバイ (ホットであってもなくても) でのプロセス障害がリロードを引き起こします。

電源障害によるスイッチオーバー

アクティブからスタンバイ ユニットへのスイッチオーバーは、現在のアクティブ ユニットの電源障害によって発生します。現在のスタンバイ ユニットが新しいアクティブ ユニットとなり、障害が発生したシステムが起動すると、そのシステムはホットスタンバイ状態に移行します。

手動スイッチオーバー

これは、アクティブとスタンバイ ユニット間でユーザーによって開始される強制スイッチオーバーです。現在のスタンバイ ユニットが新しいアクティブ ユニットとなり、障害が発生したシステムが起動すると、そのシステムはホットスタンバイ状態に移行します。手動スイッチオーバーを実行するには、`redundancy force-switchover` コマンドを実行します。このコマンドによって、アクティブからスタンバイ コントローラへのグレースフル スwitchオーバーが開始されます。アクティブ コントローラがリロードし、スタンバイが新たにアクティブ コントローラを引き継ぎます。

フェールオーバー プロセス

アクティブ ワイヤレスコントローラ

```
WLC#show ap summary
Number of APs: 1

AP Name          Slots  AP Model  Ethernet MAC  Radio MAC  Location          Country  IP Address
State
-----
AP005D.735C.B544  3      3802I    005d.735c.b544  b4de.31d0.5800  default location  US       172.20.226.186
Registered

WLC#show wireless client sum
Number of Local Clients: 1

MAC Address      AP Name          WLAN  State          Protocol Method  Role
-----
e8b2.ac94.757e  AP005D.735C.B544  1     Run            11ac             None         Local

Number of Excluded Clients: 0

WLC#redundancy force-switchover
System configuration has been modified. Save? [yes/no]: yes
Building configuration...
[OK]Proceed with switchover to standby RP? [confirm]
Manual Swact = enabled

Chassis 1 reloading, reason - Non participant detected
```

スタンバイ ワイヤレスコントローラ

アクセス ポイントおよびクライアントのステートフル スwitchオーバー (SSO) とは、すべてのアクセス ポイントおよびクライアントのセッションがステートフルにスイッチオーバーし、セッションを失うことなくネットワーク内で動作し続けることです。これによりネットワークの可用性が向上し、サービス ダウンタイムが削減されます。

冗長ペアが形成されると HA が有効になります。つまり、アクセス ポイントとクライアントはアクティブからスタンバイへのスイッチオーバー時でも接続されたままの状態を維持します。

AP とクライアント SSO の状態同期の確認

```

WLC#stby#
May 10 18:16:37.123: %PLATFORM-6-HASTATUS: RP switchover, received chassis event to become active
May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_NOT_PRESENT)
May 10 18:16:37.169: %REDUNDANCY-3-REDUNDANCY_ALARMS: Unable to assert REDUNDANCY alarm
May 10 18:16:37.169: %REDUNDANCY-3-REDUNDANCY_ALARMS: Unable to assert REDUNDANCY alarm
May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_DOWN)
May 10 18:16:37.169: %REDUNDANCY-3-SWITCHOVER: RP switchover (PEER_REDUNDANCY_STATE_CHANGE)
May 10 18:16:37.175: %PLATFORM-6-HASTATUS: RP switchover, sent message became active. IOS is ready to switch to primary after chassis confirmation
May 10 18:16:37.180: %PLATFORM-6-HASTATUS: RP switchover, received chassis event became active
May 10 18:16:37.789: %VOICE_HA-2-SWITCHOVER_IND: SWITCHOVER, from STANDBY_HOT to ACTIVE state.
May 10 18:16:37.797: %LINK-3-UPDOWN: Interface Lsmp10, changed state to up
May 10 18:16:37.798: %LINK-3-UPDOWN: Interface E0BC0, changed state to up
May 10 18:16:37.798: %LINK-3-UPDOWN: Interface LIIN0, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface Lsmp10, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface E0BC0, changed state to up
May 10 18:16:38.798: %LINEPROTO-5-UPDOWN: Line protocol on Interface LIIN0, changed state to up
May 10 18:16:39.786: %LINK-3-UPDOWN: Interface Null0, changed state to up
May 10 18:16:39.786: %LINK-3-UPDOWN: Interface TenGigabitEthernet0/0/0, changed state to up
May 10 18:16:39.787: %LINK-3-UPDOWN: Interface Vlan1, changed state to up
May 10 18:16:39.788: %LINK-3-UPDOWN: Interface Vlan112, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Null0, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface TenGigabitEthernet0/0/0, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to up
May 10 18:16:40.787: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan112, changed state to up
WLC#
May 10 18:16:49.798: %LINK-3-UPDOWN: Interface GigabitEthernet0, changed state to up
May 10 18:16:50.799: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0, changed state to up
WLC#show ap sum
WLC#show ap summary
Number of APs: 1

```

AP Name	State	Slots	AP Model	Ethernet MAC	Radio MAC	Location	Country	IP Address
AP005D.735C.B544	Registered	3	3802I	005d.735c.b544	b4de.31d0.5800	default location	US	172.20.226.186

```

WLC#show wireless client summary
Number of Local Clients: 1

```

MAC Address	AP Name	WLAN	State	Protocol Method	Role	
e8b2.ac94.757e	AP005D.735C.B544	1	Run	11ac	None	Local

```

Number of Excluded Clients: 0

```

AP とクライアント SSO の状態同期の確認

スタンバイ ワイヤレスコントローラがアクティブとして正常にスイッチオーバーすると、これまでアクティブ ワイヤレスコントローラに接続されていたすべてのアクセス ポイントとクライアントは新しいアクティブ コントローラに接続された状態を維持する必要があります。

これは、次のコマンドを実行することで確認できます。

- **show ap uptime** : スイッチオーバー後のアクセスポイントの稼働時間がリセットされていないことを確認します。

- **show wireless client summary** : 新しいアクティブコントローラに接続されているクライアントを表示します。

```

WLC#show ap uptime
Number of APs: 1

```

AP Name	Ethernet MAC	Radio MAC	AP Up Time	Association Up Time
AP005D.735C.B544	005d.735c.b544	b4de.31d0.5800	1 day 0 hour 47 minutes 22 seconds	1 day 0 hour 45 minutes 33 s

```

WLC#
WLC#show wireless client summary
Number of Local Clients: 1

```

MAC Address	AP Name	WLAN	State	Protocol Method	Role	
e8b2.ac94.757e	AP005D.735C.B544	1	Run	11ac	None	Local

```

Number of Excluded Clients: 0

```

SSO フェールオーバー時間のメトリック

メトリック	時間
障害の検出	500 ~ 1000 ミリ秒のオーダー

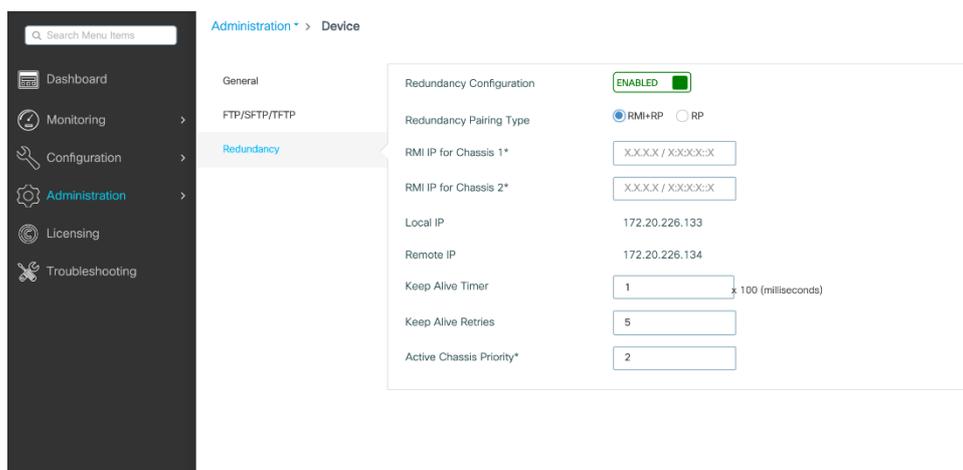
リダンダンシー マネジメント インターフェイス

SSO ペア間の単一の RP リンクで、RP のハートビートに障害が発生した場合、障害がリンクに限定されているかどうか、またはもう一方のコントローラに障害が発生しているかどうかを確認する方法はありません。アクティブとスタンバイの間の状態同期トラフィックを処理する冗長ポート (RP リンク) は、シングルポイント障害です。

リリース 17.1 では、リダンダンシー マネジメント インターフェイス (RMI) がアクティブコントローラとスタンバイコントローラ間のセカンダリリンクとして導入されました。このリリースでは、リダンダンシー マネジメント インターフェイスを使用して実行されるデフォルト ゲートウェイ チェックもサポートされます。

リリース 17.4 では、RMI インターフェイスの IPv6 サポートが導入されています。ワイヤレス管理インターフェイスでは、1 つの管理 IPv6 アドレスと 1 つの RMI IPv6 アドレスがサポートされています。RMI IPv4 または RMI IPv6 のいずれかがサポートされ、RMI IPv4 と RMI IPv6 は同時にはサポートされません。CLI の形式は IPv6 でも同じですが、IPv4 アドレスが IPv6 アドレスに置き換えられます。

WebUI を使用したリダンダンシー マネジメント インターフェイスの構成



リダダンシー マネジメント インターフェイス

- シャーシ 1 と 2 の RMI IP は、アクティブコントローラとスタンバイコントローラの両方で同じです。
- シャーシ 1 および 2 の RP IP 構成は 169.254.x.x として自動生成されます。ここで、x.x は RMI IP から取得します。
- RMI のネットマスクは、ワイヤレス管理 VLAN で設定されているネットマスクから取得されます。
- WebUI はリリース 17.4 で RMI IPv6 をサポートしています。

プログラムによる RMI IP の設定

アクティブコントローラ :

管理 VLAN のセカンダリアドレスは、アクティブの RMI です。アクティブのプライマリアドレスは、管理 IP です。次に示すように、インターフェイスに複数の「セカンダリ」アドレスを設定できます。RMI の目的のために、1 つのセカンダリ IP のみが定義されます。セカンダリ IP はプログラムで設定する必要があります。

IPv6 の場合、「セカンダリ」アドレスの概念はありません。IPv6 の場合、ワイヤレス管理 IP と RMI IP は 2 つの異なる IP として表示されます。

たとえば、冗長管理インターフェイス Vlan52 の シャーシ 1 のアドレスが 2020:0:0:1::211、シャーシ 2 のアドレスが 2020:0:0:1::212 に設定された CLI の場合を考えます。

アクティブコントローラは次のように設定されます。

```
interface Vlan52
ip address 10.100.0.1 255.252.0.0
ipv6 address 2020:0:0:1::1/64
ipv6 address 2020:0:0:1::211/64
ipv6 enable
ipv6 nd na glean
no mop enabled
no mop sysid
end
```

スタンバイコントローラ :

アドレスはアクティブによって要求されるため、管理 IP を持つことはできません。したがって、スタンバイコントローラでは、RMI IP をプライマリアドレスとしてプログラムで設定する必要があります。スタンバイがアクティブになったら、管理 IP をプライマリとして、RMI IP をセカンダリとしてプログラムする必要があります。

「セカンダリ」IP の概念は、IPv4 にのみ関連します。

```
interface Vlan52
no ip address
ipv6 address 2020:0:0:1::212/64
ipv6 enable
ipv6 nd na glean
no mop enabled
no mop sysid
end
```

RMI IPv4 を使用したデュアルスタックのサポート

RMI IPv4 を設定すると、ワイヤレス管理インターフェイスで IPv6 IP を設定できます。このアドレスは明示的に設定する必要があります。RMI が有効になっている場合、スタンバイでは設定された IPv6 アドレスがプログラムで削除され、スタンバイがアクティブに移行したときに元に戻されます。このアドレスは、コントローラがアクティブリカバリモードのときには削除されます。これにより、重複アドレス検出が回避されます。

RMI IPv6 を使用したデュアルスタックのサポート

このケースは、リリース 17.4 で発生します。17.4 では、RMI IPv6 を設定して、IPv6 のワイヤレス管理 IP を使用できます。また、ワイヤレス管理インターフェイスには IPv4 IP を設定できます。スタンバイ RMI インターフェイスが有効になると、IPv6 および IPv4 の管理 IP が設定解除され、IPv6 RMI が設定されます。スタンバイからアクティブに移行すると、管理 IP が復元されます。

ピアタイムアウトの設定

- アクティブシャーシとスタンバイシャーシが相互にキープアライブメッセージを送信して、両方が使用可能であることを確認します。ピアタイムアウトは、設定されたピアタイムアウトでピアシャーシからキープアライブメッセージを受信しなかった場合にピアシャーシが失われたと判断するために使用されます。
- デフォルトのタイムアウトは 100 ms ですが、1000 ms まで設定可能です。キープアライブの再試行回数はデフォルトで 5 回ですが、10 回まで設定できます。
- CLI コマンド :

```
WLC#chassis redundancy keep-alive timer ?
    <1-10> Chassis peer keep-alive time interval in multiple of 100 ms (enter 1
    for default)
WLC#chassis redundancy keep-alive retries ?
    <5-10> Chassis peer keep-alive retries before claiming peer is down (enter 5
    for default)
```

下位互換性のために、RP ベースの SSO 構成もサポートされますが、デフォルト ゲートウェイ チェックをサポートしないため、推奨されません。

リダンダンシー マネジメント インターフェイス

Administration > Device

Redundancy Configuration **ENABLED** Apply

Redundancy Pairing Type RMI+RP RP

Local IP*

Netmask*

HA Interface

Remote IP*

Keep Alive Timer x 100 (milliseconds)

Keep Alive Retries

Active Chassis Priority*

CLI を使用したリダンダンシー マネジメント インターフェイスの構成

17.1 より前では、RP ベースの SSO 構成のみがサポートされていました (つまり、chassis redundancy ha-interface <RP interface> local-ip <local IP> <local IP subnet> remote-ip <remote IP>)。

17.1 以降では、ユーザーは RMI+RP または RP ベースの構成を使用できます。RMI+RP 構成を使用して HA ペアが形成されると、HA ペアのクリアと形成をする RP ベース方式の exec CLI は使用できません。

注：17.x リリースでは、RMI を使用して HA を最初から起動する際に、シャーシ番号を再設定する必要があります。

コントローラをペアリングする前に、Cisco Catalyst 9800-CL クラウド ワイヤレスコントローラで **chassis redundancy ha-interface GigabitEthernet interface-number** コマンドを定義する必要があります。この手順は、Cisco Catalyst 9800-CL シリーズ ワイヤレスコントローラにのみ適用されます。選択したインターフェイスは、2 つのコントローラ間の HA 通信の専用インターフェイスとして使用されます。

デフォルトでは、シャーシ番号は 1 です。RP ポートの IP アドレスは RMI から取得されます。両方のコントローラでシャーシ番号が同じ場合、ローカル RP ポートで取得される IP は同じになり、検出は失敗します。これにより、アクティブ-アクティブケースが発生します。

このシナリオを回避するには、次の CLI を実行します。

```
WLC#chassis 1 renumber ?
  <1-2> Renumbr local chassis id assignment

WLC(config)# redun-management interface <VLAN> chassis 1 address <RMI IP of chassis 1>
chassis 2 address <RMI IP of chassis 2>
```

設定例：

WLC 1：

```
WLC(config)# redun-management interface Vlan112 chassis 1 address 172.20.226.148 chassis 2
address 172.20.226.149
```

WLC 2：(同じ CLI)

```
WLC(config)# redun-management interface Vlan112 chassis 1 address 172.20.226.148 chassis 2
address 172.20.226.149
```

リダンダンシー マネジメント インターフェイス

シャーシ番号で個々のコントローラを識別し、この番号は RMI IP を設定する前に設定する必要があります。ペアを形成する前に、両方のコントローラで同じ CLI を実行する必要があります。RMI IP 構成によって HA ペアリングがトリガーされ、SSO ペアが形成されます。

RMI と RP 構成の確認

```
WLC-9800#show chassis rmi
Sep 20 21:26:13.024: %SYS-5-CONFIG_I: Configured from console by console
Chassis/Stack Mac Address : 00a3.8e23.8760 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
```

Chassis#	Role	Mac Address	Priority	H/W Version	Current State	IP	RMI-IP
1	Standby	00a3.8e23.8760	2	V02	Ready	169.254.226.149	172.20.226.149
*2	Active	00a3.8e23.8900	1	V02	Ready	169.254.226.148	172.20.226.148

```
WLC-9800#show romvar
ROMMON variables:
  SWITCH_NUMBER = 1
  LICENSE_BOOT_LEVEL =
...
  RANDOM_NUM = 842430634
  SWITCH_PRIORITY = 1
  RMI_INTERFACE_NAME = Vlan112
  RMI_CHASSIS_LOCAL_IP = 172.20.226.148
  RMI_CHASSIS_REMOTE_IP = 172.20.226.149
  CHASSIS_HA_LOCAL_IP = 169.254.226.148
  CHASSIS_HA_REMOTE_IP = 169.254.226.149
  CHASSIS_HA_LOCAL_MASK = 255.255.255.0
```

The following shows the scenario where the RP IP is derived from RMI IPv6 address:

```
D3-5-Dao#show chassis rmi
Chassis/Stack Mac Address : 00a3.8e23.a540 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
```

Chassis#	Role	Mac Address	Priority	H/W Version	Current State	IP	RMI-IP
*1	Active	706d.1536.23c0	1	V02	Ready	169.254.254.17	2020:0:0:1::211
2	Standby	00a3.8e23.a540	1	V02	Ready	169.254.254.18	2020:0:0:1::212

RMI と RP ペアリングの組み合わせ

アップグレードと以前の HA 構成を使用しない HA ペアリング

既存の方式 (exec RP ベースの CLI) または RMI IP ベースの方式を選択するオプションが表示されます。

ユーザーが exec CLI ベースの方式を選択した場合、RP IP は 16.12 までと同様に設定されます。

RMI 構成が完了すると、次が実行されます。

リダンダンシー マネジメント インターフェイス

RMI IP から取得した IP を使用して RP IP を生成します。RMI 構成は RMI IP の設定とコントローラのペアリングにも使用します (ペアリング時にスタンバイのみがハードウェア プラットフォームでリロードされます。9800-CL VM の場合はアクティブとスタンバイの両方がリロードされます)。この場合、exec RP ベースの CLI はブロックされます。

オプション 1 : RMI ベースの構成 (推奨)

1. 17.1 にアップグレードし、RP を接続します
2. RMI+RP を構成します
3. RP IP は RMI IP から取得されます
4. RP ベースの exec コマンドはブロックされます
5. ROMMON RP および RMI 変数が設定されます

オプション 2 : RP ベースの構成

1. 17.1 にアップグレードし、RP を接続します
2. GUI または CLI を使用して RP を構成します
3. RP ベースの構成によってローカル IP とリモート IP が設定されます
4. ROMMON RP 変数がローカル IP およびリモート IP に設定されます

すでにペアリングされているコントローラのアップグレード

コントローラがすでに HA ペアになっている場合は、既存の exec RP CLI が引き続き使用できます。

RMI ベースの HA ペアリング (推奨) に移行する場合は、RMI を有効にできます。

これにより、RP IP が RMI から取得した IP で上書きされます。HA ペアはすぐには影響を受けませんが、次のリロードが発生するとコントローラによって新しい IP が選択されます。

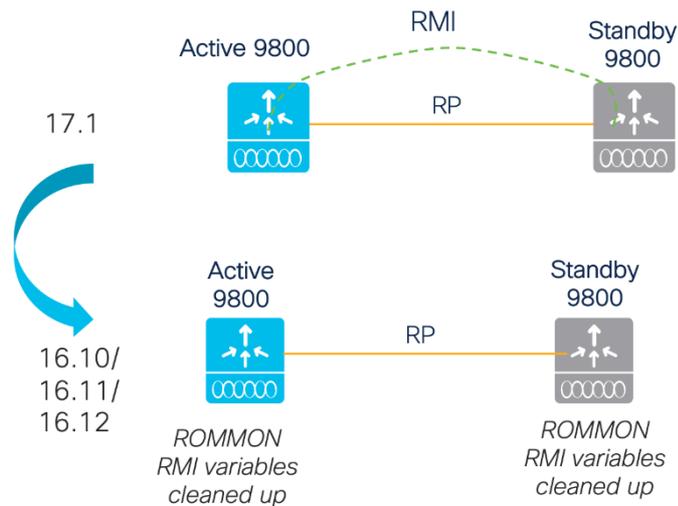
RMI 機能を有効にするには、リロードする必要があります。

コントローラがリロードされると、RMI から取得した新しい RP IP でペアが開始されます。exec RP ベースの CLI はブロックされます。

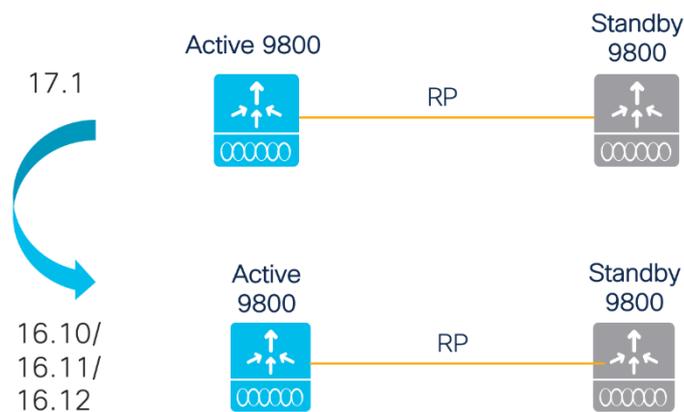
ダウングレード

RMI ベースの構成が使用された場合、ダウングレード後にシステムは RP ベースの構成にフォールバックします。

デフォルト ゲートウェイ チェック



RP ベースの構成が使用された場合、ダウングレード後もシステムは RP ベースの構成を使用します。



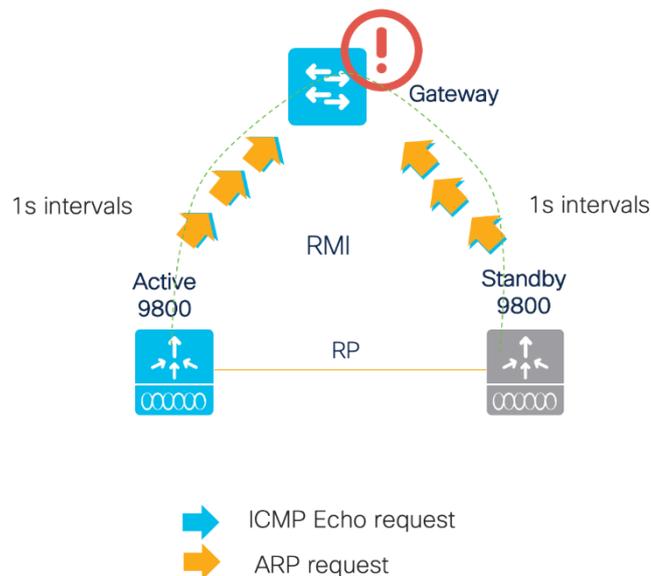
デフォルト ゲートウェイ チェック

デフォルト ゲートウェイ チェックを実行するには、Internet Control Message Protocol (ICMP) ping をゲートウェイに定期的に送信します。アクティブコントローラとスタンバイコントローラの両方が、送信元 IP として RMI IP を使用します。これらのメッセージは 1 秒間隔で送信されます。ゲートウェイへの到達に 8 回連続して失敗すると、コントローラはそのゲートウェイを到達不能と宣言します。

4 回の ICMP エコー要求で ICMP エコー応答が得られないと、ARP 要求が試行されます。8 秒間応答がない場合 (4 回の ICMP エコー要求に続いて 4 回の ARP 要求)、ゲートウェイは到達不能と見なされます。

IPv6 デフォルトゲートウェイの検出は、リリース 17.4 以降でサポートされています。IPv4 での ICMP および ARP の代わりに、IPv6 では ICMP を使用してゲートウェイ障害を検出します。

ゲートウェイ障害検出間隔の設定



Catalyst 9800 ワイヤレスコントローラには、アクティブ-アクティブシナリオを回避する 2 つのリカバリ状態があります。

論理的にリカバリモードとは、サービスを提供するための「リソース」の一部をコントローラが使用できない状態を意味します。現在、RP、RMI、およびゲートウェイがリソースです。ポートはリカバリモードで管理停止状態になるため、トラフィックは通過しません。

■ スタンバイ-リカバリ：ゲートウェイがダウンすると、スタンバイがスタンバイ-リカバリモードになります。スタンバイとは、その状態がアクティブと同期されることを意味します。ただし、もう一方のリソース（ゲートウェイ）がないため、スタンバイ-リカバリモードになります。スタンバイがスタンバイ-リカバリモードになると、スタンバイはアクティブの機能を引き継ぐことができません。スタンバイ-リカバリは、ゲートウェイが到達可能になったことを検出すると、リロードなしでスタンバイに戻ります。

■ アクティブ-リカバリ：RP がダウンしたときのモードです。アクティブ-リカバリの内部状態はアクティブと同期しません。RP リンクがアップ状態になると、アクティブ-リカバリがリロードされるため、一括同期でスタンバイとして起動できます。

ゲートウェイがダウンした結果としてスイッチオーバーがトリガーされた場合、スイッチオーバー履歴にはスイッチオーバーの理由がゲートウェイダウンとして表示されます。

ゲートウェイ障害検出間隔の設定

ゲートウェイ障害検出間隔は、リリース 17.4 以降で次の CLI を使用して設定できます。

```
WLC(config)#management gateway-failover interval <6 - 12>
```

デフォルト値は 8 です。

このパラメータは、YANG、SNMP、および WebUI でも設定できます。設定パラメータは、IPv6 ゲートウェイモニタリングにも適用されます。

ゲートウェイ障害検出間隔の設定

Administration > Device

General

FTP/SFTP/TFTP

Redundancy

Redundancy Configuration ENABLED Apply

Redundancy Pairing Type RMI+RP RP

RMI IP for Chassis 1* 9.4.41.110

RMI IP for Chassis 2* 9.4.41.120

Management Gateway Failover ENABLED

Gateway Failure Interval (seconds) 10

Local IP 169.254.41.110

Remote IP 169.254.41.120

Keep Alive Timer 1 x 100 (milliseconds)

Keep Alive Retries 5

Chassis Renumber 1

Active Chassis Priority* 1

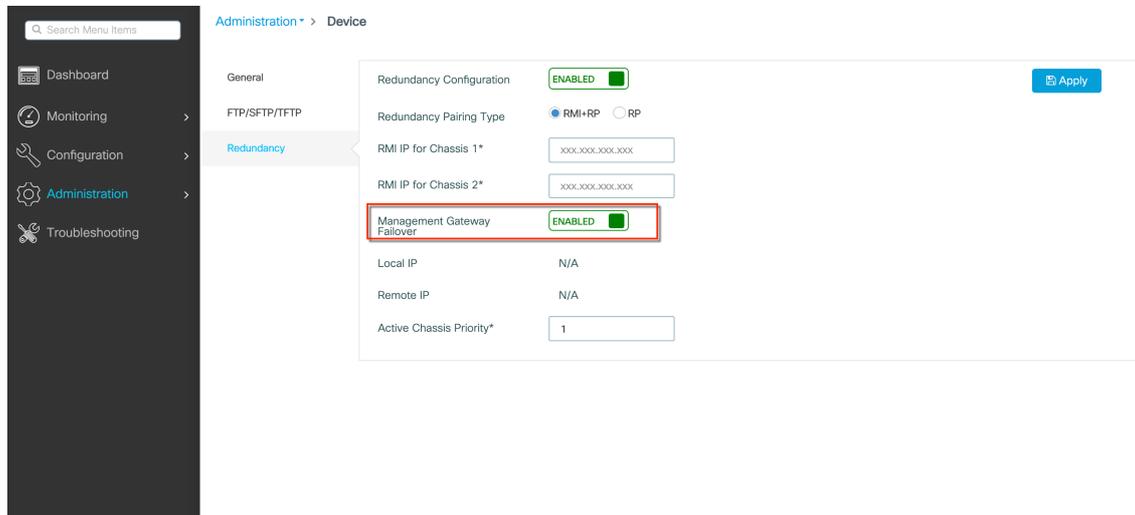
Standby Chassis Priority* 1

NETCONF/YANG サポートのサンプル JSON

```
{
  "Cisco-IOS-XE-native:management": {
    "Cisco-IOS-XE-rmi-dad:gateway-failover": {
      "enable": true,
      "interval": 10
    }
  }
}
```

デフォルト ゲートウェイ チェックの WebUI 設定

デフォルト ゲートウェイ チェック オプションは、[Administration] > [Device] > [Redundancy] > [Management Gateway Failover] で設定できます。



デフォルト ゲートウェイ チェックの CLI 設定

ゲートウェイチェック機能を有効にして、この機能で使用されるデフォルトゲートウェイ IP を指定するには、次の CLI を設定する必要があります。

```
WLC-9800 (config)#management gateway-failover enable
WLC-9800#ip default-gateway <IP>
```

ゲートウェイチェックが有効になっているかどうかを確認するには、CLI の show redundancy state を使用します。

```
WLC-9800#show redundancy states
  my state = 13 -ACTIVE
  peer state = 8 -STANDBY HOT
    Mode = Duplex
    Unit = Primary
    Unit ID = 2
  Redundancy Mode (Operational) = sso
  Redundancy Mode (Configured) = sso
  Redundancy State = sso
  ...
  Gateway Monitoring = Enabled
```

17.2 では、「ip default-gateway <IP>」は使用できなくなります。ゲートウェイ IP は、設定されたスタティック IP ルートから取得されます。HA インフラストラクチャは、RMI ネットワークに一致するスタティックルート IP を選択します。複数のスタティックルートが設定されている場合、最も広いネットワーク範囲に設定されたルートが選択されます。同じネットワーク範囲に複数のゲートウェイを構成できます。同じネットワークに複数のゲートウェイがある場合は、最も広いマスクと最も小さいゲートウェイ IP が選択されます。ゲートウェイ IP は、必要に応じて、スタティックルートの設定更新時に再評価されます。

システムおよびネットワークの障害処理

一連のスタティックルートからゲートウェイ IP を選択する上記のメカニズムは、リリース 17.4 の IPv6 に適用されます。

注：

- 物理ポートダウンシナリオの場合、17.3.2 より前のリリースではゲートウェイチェック機能により検出されるため、検出には 8 秒かかります。リリース 17.3.2 以降では、ポートがダウン状態になると、ゲートウェイのダウンに関連するすべてのアクションがトリガーされます。ポートがダウンしたことが検出されるため、SSO を示すための新しい理由コードが使用されます。「Active GW Lost」の代わりに「Active RMI Port Down」が使用されます。

- リリース 17.1 では、物理ポートのステータスがアクティブコントローラからスタンバイコントローラに同期されます。これはリリース 17.2 で修正されており、アクティブコントローラとスタンバイコントローラは、それぞれのポートステータスを維持します。

システムおよびネットワークの障害処理

スタンバイコントローラがクラッシュすると、リブートしてスタンバイとして起動します。一括同期が実行され、スタンバイがホットになります。アクティブコントローラがクラッシュすると、スタンバイがアクティブになります。新しいアクティブはマスターのロールを引き受け、デュアルアクティブを検出しようとしています。

次のマトリクスを参照すると、WLC スイッチオーバーのトリガー条件がはっきりとわかります。

システムに関する問題				
トリガー	RP リンクステータス	RMI を介したピアの到達可能性	スイッチオーバー	結果
クリティカルプロセスのクラッシュ	アップ	到達可能	対応	スイッチオーバー発生
強制スイッチオーバー	アップ	到達可能	対応	スイッチオーバー発生
クリティカルプロセスのクラッシュ	アップ	到達不能	対応	スイッチオーバー発生
強制スイッチオーバー	アップ	到達不能	対応	スイッチオーバー発生
クリティカルプロセスのクラッシュ	ダウン	到達可能	非対応	アクションはありません。1 台のコントローラがリカバリモードになります。

強制スイッチオーバー	ダウン	到達可能	該当なし	アクションはありません。1 台のコントローラがリカバリモードになります。
クリティカルプロセスのクラッシュ	ダウン	到達不能	非対応	二重障害：「ネットワークエラーの処理」を参照
強制スイッチオーバー	ダウン	到達不能	該当なし	二重障害：「ネットワークエラーの処理」を参照

RP リンク	RMI を介したピアの到達可能性	アクティブからのゲートウェイ	スタンバイからゲートウェイ	スイッチオーバー	結果
アップ	アップ	到達可能	到達可能	非対応	動作なし
アップ	アップ	到達可能	到達不能	非対応	[アクションなし (No Action)]。ゲートウェイに到達できないため、スタンバイはこの状態では SSO に対応できません。スタンバイは、スタンバイ-リカバリモードになります。RP がダウンすると、スタンバイ (リカバリモード) がアクティブになります。

アップ	アップ	到達不能	到達可能	対応	ゲートウェイ到達可能性メッセージは、RMI + RP リンク経由で交換されます。スタンバイがアクティブになるように、アクティブのリブートが実行されます。
アップ	アップ	到達不能	到達不能	非対応	これにより、アクティブ SVI がダウンすると、スタンバイ SVI もダウンします。スイッチオーバーがトリガーされます。新しいアクティブによってゲートウェイが到達可能であると検出された場合は、アクティブ-スタンバイのリカバリでシステムが安定します。それ以外の場合、スイッチオーバーはピンポン方式で行われます。
アップ	ダウン	到達可能	到達可能	非対応	アクションなし

アップ	ダウン	到達可能	到達不能	非対応	ゲートウェイに到達できないため、スタンバイはこの状態では SSO に対応できません。LMP メッセージは RP リンク経由でも交換されるため、スタンバイはリカバリモードになります。
アップ	ダウン	到達不能	到達可能	対応	ゲートウェイ到達可能性メッセージは RP リンク経由でも交換されます。スタンバイがアクティブになるように、アクティブのリポートが実行されます。
アップ	ダウン	到達不能	到達不能	非対応	これにより、アクティブ SVI がダウンすると、スタンバイ SVI もダウンします。スイッチオーバーがトリガーされます。新しいアクティブによってゲートウェイが到達可能であると検出された場合は、アクティブ-スタンバイのリカバリでシステムが安定します。それ以外の場合、スイッチオーバーはピンポン方式で行われます。

ダウン	アップ	到達可能	到達可能	対応	スタンバイがアクティブになり、(以前の)アクティブはアクティブ-リカバリに移行します。コンフィギュレーションモードは、アクティブ-リカバリモードでは無効です。すべてのインターフェイスが「管理上ダウン」し、ワイヤレス管理インターフェイスに RMI IP が設定されます。アクティブ-リカバリモードのコントローラは、RP リンクがアップ状態になるとリロードしてスタンバイになります。
ダウン	アップ	到達可能	到達不能	対応	同上
ダウン	アップ	到達不能	到達可能	対応	同上
ダウン	アップ	到達不能	到達不能	対応	同上

HA ペアリング解除動作

ダウン	ダウン	到達可能	到達可能	対応	二重障害：アクティブコントローラが 2 台になるため、ネットワークの競合が生じます。スタンバイがアクティブになります。従来のアクティブも存在します。ロールネゴシエーションは、接続が復元された後に実行され、最後にアップ状態になった方のアクティブが維持されます。
ダウン	ダウン	到達可能	到達不能	対応	同上
ダウン	ダウン	到達不能	到達可能	対応	同上
ダウン	ダウン	到達不能	到達不能	対応	同上

HA ペアリング解除動作

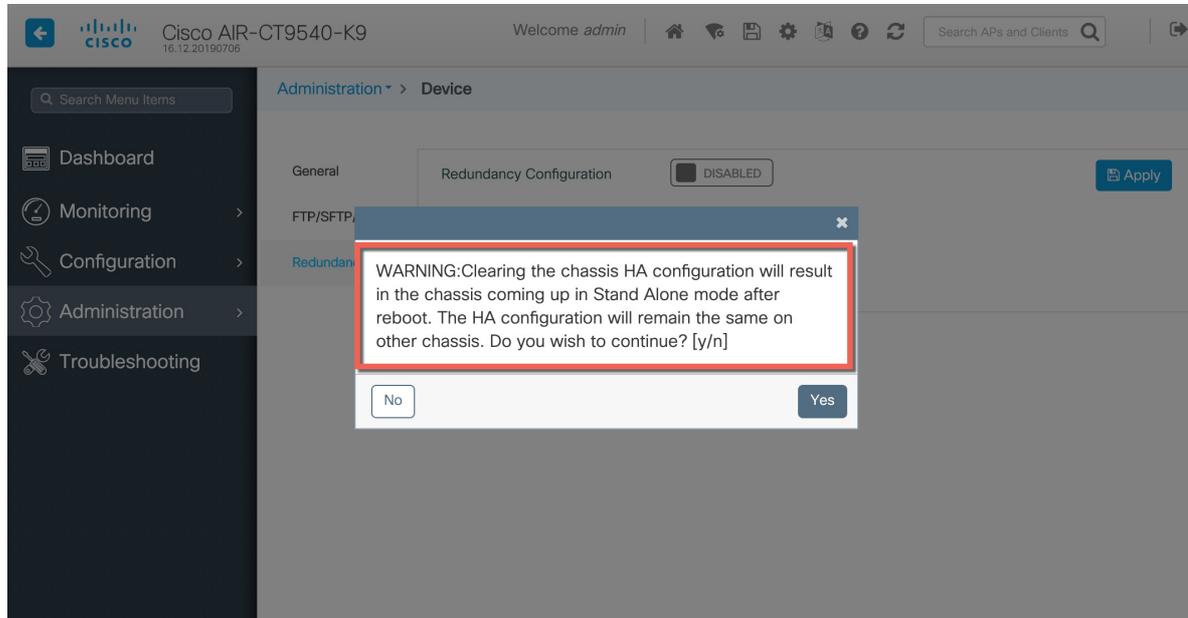
リリース 16.10 および 16.11 では、コマンド「clear chassis redundancy」を発行して HA ペアを切り離すと、スタンバイコントローラがリブートし、アクティブコントローラとまったく同じ構成で起動し、次のメッセージが表示される重複 IP アドレスエラーが発生します。

```
WLC#sh log | i DUP
Mar 21 21:53:46.307 CET: %IP-4-DUPADDR: Duplicate address 120.0.0.1 on Vlan120, sourced by
d4c9.3ccc.f98b
Mar 21 21:54:16.947 CET: %IP-4-DUPADDR: Duplicate address 172.18.50.60 on
GigabitEthernet0, sourced by d4c9.3ccc.f981
```

HA ペアリング解除動作

16.12 および 17.1 で実装される解決策は、HA のペアリング解除後に、スタンバイコントローラのスタートアップ構成と HA 構成がクリアされ、スタンバイが 0 日目に移行することです。

コマンドを実行する前に、アクティブコントローラで次の警告が表示されます。



CLI でも同様です。

```
WLC#clear chassis redundancy
WARNING: Clearing the chassis HA configuration will result in both the chassis move into Stand Alone mode. This involves reloading the standby chassis after clearing its HA configuration and startup configuration which results in standby chassis coming up as a totally clean after reboot. Do you wish to continue? [y/n]? [yes]:
* Apr 3 23:42:22.985: received clear chassis .. ha_supported:lyes
WLC#
*Apr 3 23:42:25.042: clearing peer startup config
*Apr 3 23:42:25.042: chkpt send: sent msg type 2 to peer..
*Apr 3 23:42:25.043: chkpt send: sent msg type 1 to peer..
*Apr 3 23:42:25.043: Clearing HA configurations
*Apr 3 23:42:26.183: Successfully sent Set chassis mode msg for chassis 1.chasfs file updated
*Apr 3 23:42:26.359: %IOSXE_REDUNDANCY-6-PEER_LOST: Active detected chassis 2 is no longer standby
```

スタンバイコントローラでは、構成がクリアされることを示す次のメッセージが表示されます。

```
WLC-stby#
*Apr 3 23:40:40.537: mcprp_handle_spa_oir_tsm_event: subslot 0/0 event=2
*Apr 3 23:40:40.537: spa_oir_tsm subslot 0/0 TSM: during state ready, got event 3(ready)
*Apr 3 23:40:40.537: @@@ spa_oir_tsm subslot 0/0 TSM: ready -> ready
*Apr 3 23:42:25.041: Removing the startup config file on standby
*Apr 3 23:42:26.466: Calling HA configs clear on standby
*Apr 3 23:42:26.466: Clearing HA configurations
*Apr 3 23:42:27.499: Successfully sent Set chassis mode msg for chassis 2.chasfs file updated
```

SSO ペアでの LACP、PAGP サポート

注：RMI ベースの構成を使用している場合に SSO ペアを解除するには、RMI 構成コマンドの「no」バージョンを使用した後にリロードします。

```
WLC(config)# no redun-management interface <VLAN> chassis 1 address <RMI IP of chassis 1>
chassis 2 address <RMI IP of chassis 2>
```

SSO ペアでの LACP、PAGP サポート

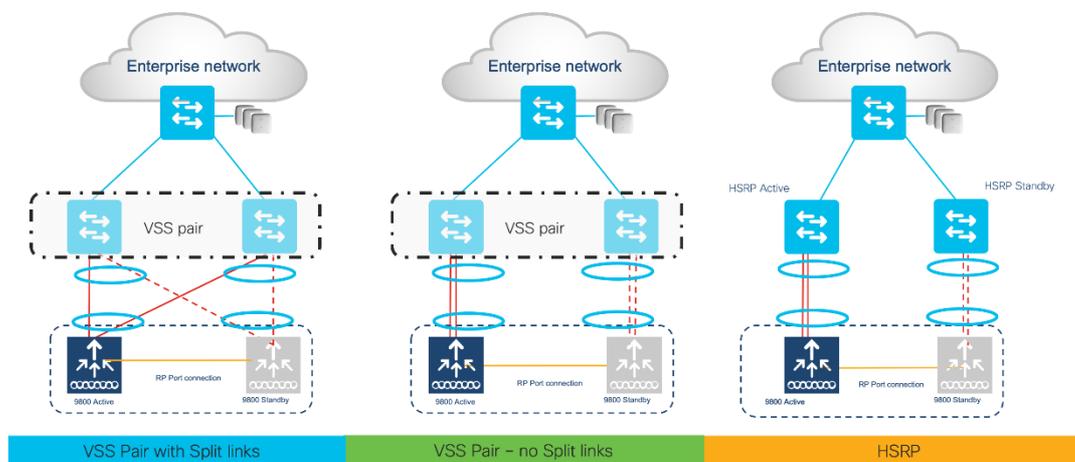
LACP プロトコル (IEEE 802.3ad) では、2 つのデバイス間でリンク集約制御プロトコルデータユニット (LACPDU) を交換することによって、物理イーサネット インターフェイスが集約されます。

LACP、PAGP のサポートは、スタンバイコントローラのリンクまたは接続障害を検出してモニタし、スイッチオーバー (SSO) 時にクライアント データ トラフィックをシームレスに転送するために、SSO ペアが必要です。17.1 より前は、SSO モードでは LAG モード ON のみがサポートされていました。17.1 では、LACP (アクティブおよびパッシブ) と PAGP の両方が SSO モードでサポートされます。

この機能は、Cisco Catalyst 9800-L、Cisco Catalyst 9800-40、および Cisco Catalyst 9800-80 (モジュールポートを含む) でサポートされます。

サポートされる LACP、PAGP トポロジ

次のトポロジは、SSO および LACP/PAGP でサポートされます。



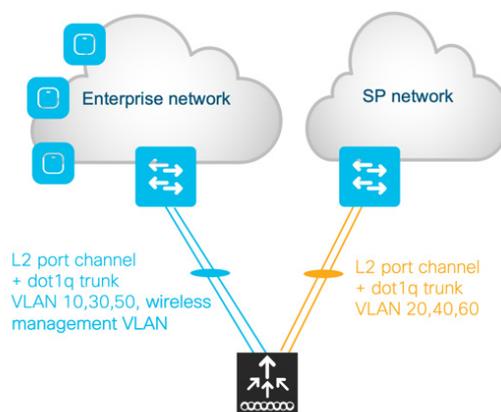
次は、LACP、PAGP トポロジではサポートされません。

- Auto-LAG はサポートされません。
- C9800-CL および AP 上の EWC はサポートされません。
- L3 ポートチャネルはサポートされません。

マルチシャーシリンク集約グループ

リリース 17.2.1 以降、マルチシャーシリンク集約グループは、スタンドアロンコントローラおよび HA ペアでサポートされます。マルチシャーシ LAG を使用すると、複数のアップリンクをコントローラから個別のアップリンクスイッチに接続できます。

これにより、マルチスイッチトポロジに接続された場合に、スイッチ インフラストラクチャと VLAN ベースのトラフィック分割にコントローラを接続する柔軟性が得られます。たとえば、エンタープライズ トラフィックとは完全に異なるスイッチまたはネットワーク上のゲストトラフィックを分離できます。各 LAG は単一のスイッチに接続し、それぞれの VLAN の割り当て先 LAG は異なっている必要があります。

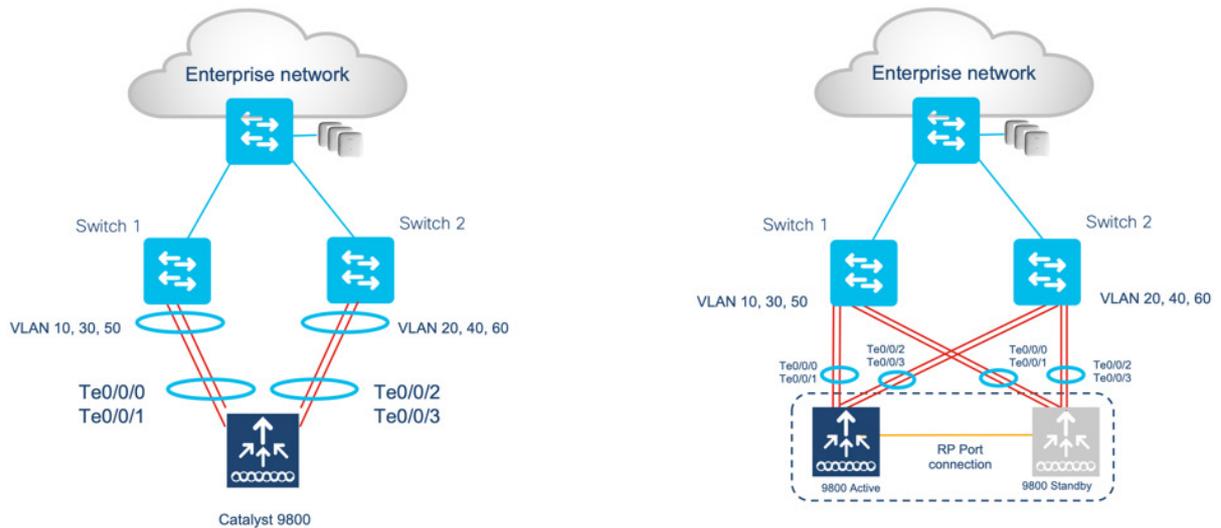


注：ループを作成しないように構成するのは、ユーザーの責任範囲です。

サポートされるマルチシャーシ LAG トポロジ

- マルチシャーシ LAG は、LAG モード ON およびダイナミック LAG (LACP および PAGP) でサポートされます。
- マルチシャーシ LAG は、次に示すように、スタンドアロンコントローラと HA ペアでサポートされます。

SSO ペアでの LACP、PAGP サポート



注：複数の LAG を持つコントローラは単一のスイッチに接続できますが、それぞれの VLAN の接続先 LAG は異なっている必要があります。

サポート対象プラットフォーム：

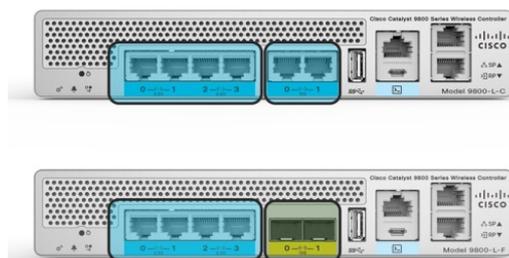
マルチシャーシ LAG は、次のプラットフォームでサポートされます。

- Catalyst 9800-L ワイヤレスコントローラ
- Catalyst 9800-40 ワイヤレスコントローラ
- Catalyst 9800-80 ワイヤレスコントローラ

サポートされる LAG ポートのグループ化

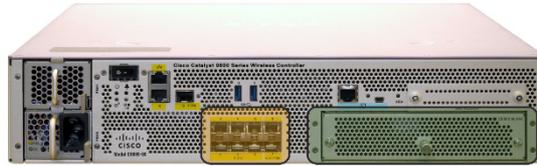
ベストプラクティスは、ポートチャンネルで同じタイプと速度のポートを使用することです。

- 個別のポートチャンネルで 2.5G/1G および 10G/mGig ポートを備えた 9800-L-C
- 個別のポートチャンネルで 2.5G/1G および 10G/1G 光ファイバポートを備えた 9800-L-F



分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

ベイ 0 およびベイ 1 (モジュラスロット) の 9800-80 ポートは、同じポートチャンネルグループに結合できません。ベストプラクティスは、ポートチャンネルで同じスロットのポートを使用することです。



分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

ワイヤレスコントローラの設定

アクティブ WLC :

```
WLC#sh etherchannel summary
```

```
Flags: D - down          P - bundled in port-channel
```

```
       I - stand-alone s - suspended
```

```
       H - Hot-standby (LACP only)
```

```
       R - Layer3        S - Layer2
```

```
       U - in use       f - failed to allocate aggregator
```

```
       M - not in use, minimum links not met
```

```
       u - unsuitable for bundling
```

```
       w - waiting to be aggregated
```

```
       d - default port
```

```
       A - formed by Auto LAG
```

```
Number of channel-groups in use: 1
```

```
Number of aggregators:          1
```

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

Group	Port-channel	Protocol	Ports	
2	Po2(SU)	LACP	Te0/0/0(P)	Te0/0/3(P)

```
WLC#sh run int po2
```

```
Building configuration...
```

```
Current configuration : 54 bytes
```

```
!
```

```
interface Port-channel2
```

```
switchport mode trunk
```

```
end
```

```
WLC#sh run int te0/0/0
```

```
Building configuration...
```

```
Current configuration : 114 bytes
```

```
!
```

```
interface TenGigabitEthernet0/0/0
```

```
switchport mode trunk
```

```
no negotiation auto
```

```
channel-group 2 mode active
```

```
end
```

```
WLC#sh run int te0/0/3
```

```
Building configuration...
```

```
Current configuration : 114 bytes
```

```
!
```

```
interface TenGigabitEthernet0/0/3
```

```
switchport mode trunk
```

```
no negotiation auto
```

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

```
channel-group 2 mode active
end
```

スタンバイ WLC :

WLC-stby#sh etherchannel summary

Flags: D - down P - bundled in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use f - failed to allocate aggregator

M - not in use, minimum links not met

u - unsuitable for bundling

w - waiting to be aggregated

d - default port

A - formed by Auto LAG

Number of channel-groups in use: 1

Number of aggregators: 1

Group	Port-channel	Protocol	Ports
2	Po2(SU)	LACP	Te0/0/0(P) Te0/0/3(P)

WLC-stby#sh run int po2

Building configuration...

Current configuration : 54 bytes

!

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

```
interface Port-channel2
switchport mode trunk
end
```

```
WLC-stby#sh run int te0/0/0
Building configuration...
```

Current configuration : 114 bytes

!

```
interface TenGigabitEthernet0/0/0
switchport mode trunk
no negotiation auto
channel-group 2 mode active
end
```

```
WLC-stby#sh run int te0/0/3
Building configuration...
```

Current configuration : 114 bytes

!

```
interface TenGigabitEthernet0/0/3
switchport mode trunk
no negotiation auto
channel-group 2 mode active
end
```

```
WLC-stby#
```

VSS の設定

Router#sh etherchannel summary

Flags: D - down P - bundled in port-channel

I - stand-alone s - suspended

H - Hot-standby (LACP only)

R - Layer3 S - Layer2

U - in use N - not in use, no aggregation

f - failed to allocate aggregator

M - not in use, no aggregation due to minimum links not met

m - not in use, port not aggregated due to minimum links not met

u - unsuitable for bundling

d - default port

w - waiting to be aggregated

Number of channel-groups in use: 9

Number of aggregators: 9

Group	Port-channel	Protocol	Ports
6	Po6(RD)	-	
10	Po10(RU)	-	Te1/5/4(P)
20	Po20(RU)	-	Te2/5/4(P)
30	Po30(SU)	LACP	Gi1/4/1(P) Gi2/4/1(P)
40	Po40(SD)	-	
60	Po60(SU)	LACP	Gi1/4/3(P) Gi2/4/4(P)
61	Po61(SU)	LACP	Gi1/4/4(P) Gi2/4/3(P)
833	Po833(SU)	-	Te1/1/1(P) Te1/1/2(P)
865	Po865(SU)	-	Te2/1/1(P) Te2/1/2(P)

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

```
Router#sh run int po60
```

```
Building configuration...
```

```
Current configuration : 67 bytes
```

```
!
```

```
interface Port-channel60
```

```
switchport
```

```
switchport mode trunk
```

```
end
```

```
Router#sh run int po61
```

```
Building configuration...
```

```
Current configuration : 67 bytes
```

```
!
```

```
interface Port-channel61
```

```
switchport
```

```
switchport mode trunk
```

```
end
```

```
Router#sh run int gi1/4/3
```

```
Building configuration...
```

```
Current configuration : 103 bytes
```

```
!
```

```
interface GigabitEthernet1/4/3
```

```
switchport
```

```
switchport mode trunk
```

分割リンクを使用して VSS ペアに接続する HA SSO ペアの LAG 設定例

```
channel-group 60 mode active
end
```

```
Router#sh run int gi2/4/4
```

```
Building configuration...
```

```
Current configuration : 103 bytes
```

```
!
```

```
interface GigabitEthernet2/4/4
```

```
switchport
```

```
switchport mode trunk
```

```
channel-group 60 mode active
```

```
end
```

```
Router#sh run int Gi1/4/4
```

```
Building configuration...
```

```
Current configuration : 103 bytes
```

```
!
```

```
interface GigabitEthernet1/4/4
```

```
switchport
```

```
switchport mode trunk
```

```
channel-group 61 mode active
```

```
end
```

```
Router#sh run int Gi2/4/3
```

```
Building configuration...
```

```
Current configuration : 103 bytes
```

```
!
```

```
interface GigabitEthernet2/4/3
```

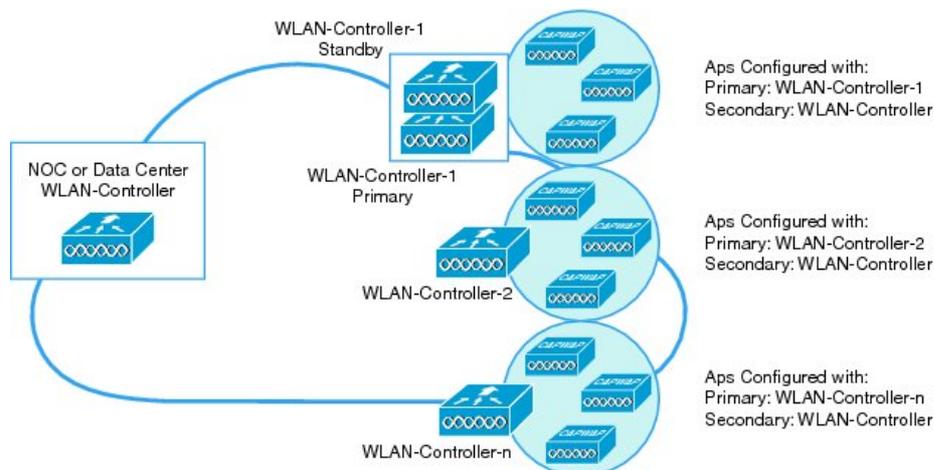
HA セットアップでのコントローラの置き換え

```
switchport
switchport mode trunk
channel-group 61 mode active
end
```

HA セットアップでのコントローラの置き換え

- ペアを解除せずに HA ペアからアクティブコントローラを削除します。アクティブコントローラがなくなると、スタンバイコントローラがアクティブのロールを引き継ぎます。
- 以前のアクティブコントローラと同じ構成で新しい 9800 コントローラを準備します。つまり、同じソフトウェアバージョン、ライセンスレベル、IP アドレス WMI、RMI、モビリティ MAC で構成します。
- 新しいコントローラを SSO でペアリングする前に、アクティブコントローラがリブートしたとしても、現在のアクティブコントローラがアクティブなままにするには、現在のアクティブコントローラに高い優先順位を設定します。
- 冗長ポート (RP) を使用して新しい 9800 コントローラを物理的に接続します。
- 新しい 9800 コントローラでの SSO 構成の有効化
- 新しい 9800 コントローラがリブートし、現在のアクティブコントローラとペアリングしたスタンバイとして起動します。

SSO ハイブリッド導入による N+1



352083

RMI を使用したスタンバイのモニタリング

上に示したように、SSO 冗長ペアと N+1 プライマリ、セカンダリ、およびターシャリ モデルのハイブリッド トポロジがサポートされています。DR サイトのセカンダリコントローラは、Catalyst C9800-L、C9800-40、C9800-80、または C9800-CL ワイヤレスコントローラです。Catalyst 9800 ワイヤレスコントローラから CUWN コントローラにフェール バックするアクセス ポイントはコードを再度ダウンロードしてから CUWN ワイヤレスコントローラに参加します。また、CUWN コントローラから Catalyst 9800 ワイヤレスコントローラにフェール バックするアクセス ポイントも同様のことを実行します。

RMI を使用したスタンバイのモニタリング

この機能により、アクティブコントローラを経由せずに、プログラムインターフェイス (NETCONF/YANG、RESTCONF) および CLI を使用して、HA ペアのスタンバイコントローラのシステム正常性をモニタできます。これには、CPU、メモリ、インターフェイスステータス、PSU (電源装置) 障害、ファン障害、温度などのパラメータのモニタが含まれます。このモニタ機能は、Cisco Catalyst 9800-CL プライベートクラウド、9800-L、9800-40、および 9800-80 ワイヤレスコントローラでサポートされます。

RMI インターフェイスを使用すると、次のことができます。

- ポート 22 で IOS SSH サーバーに接続し、show CLI の選択セットを実行します。
- ポート 830 で NETCONF SSH サーバーに接続し、プログラムインターフェイスを使用して NETCONF/YANG にアクセスします。
- HTTPS ポート 443 に接続し、RESTCONF を使用してプログラムインターフェイスを使用します。

ローカル認証および RADIUS を使用する外部 AAA サーバーのユーザーログイン情報をローカルに設定できます。SSH 認証は、ユーザー名とパスワードを使用して実行します。スタンバイコントローラは、証明書ベースの認証を処理するために PKI インフラストラクチャを実行しません。外部 AAA サーバーは、スタンバイコントローラで静的に設定できるデフォルトルートを紹介して到達可能にする必要があります。

syslog は、コンソールログとしてスタンバイコントローラでサポートされます。

RMI IPv6 を使用したスタンバイモニタリングは、リリース 17.4 以降でサポートされています。

スタンバイモニタリング CLI

- 電源、ファン、温度のステータスを確認するには、次の CLI を物理アプライアンスで使用できます。仮想プラットフォームの場合、この出力は空になります。

Show environment

```
9800-stby#show environment summary
```

```
Number of Critical alarms: 0
Number of Major alarms: 0
Number of Minor alarms: 0
```

Slot	Sensor	Current	State	Reading
Threshold (Minor, Major, Critical, Shutdown)				
P0	Vin	218	V AC	na
P0	Iin	1	A	na
P0	Vout	12	V DC	na

RMI を使用したスタンバイのモニタリング

```

P0          Iout          Normal          20    A          na
P0          Temp1        Normal          31    Celsius
(na ,na ,na ,na ) (Celsius)
P0          Temp2        Normal          42    Celsius
(na ,na ,na ,na ) (Celsius)
P0          Temp3        Normal          43    Celsius
(na ,na ,na ,na ) (Celsius)
P1          Vin          Normal          0     V AC         na
P1          Iin          Normal          0     A           na
P1          Vout         Normal          0     V DC         na
P1          Iout          Normal          1     A           na
P1          Temp1        Normal          28    Celsius
(na ,na ,na ,na ) (Celsius)
P1          Temp2        Normal          29    Celsius
(na ,na ,na ,na ) (Celsius)
P1          Temp3        Normal          0     Celsius
(na ,na ,na ,na ) (Celsius)
R0          VRRX1: VX1    Normal          751   mV          na
R0          VRRX1: VX2    Normal          6937  mV          na
R0          VRRX1: VX3    Normal          1217  mV          na
R0          VRRX1: VX5    Normal          1222  mV          na
R0          VRRX1: VP1    Normal          1705  mV          na
R0          VRRX1: VP2    Normal          2489  mV          na
R0          VRRX1: VP3    Normal          1300  mV          na
R0          VRRX1: VP4    Normal          5070  mV          na
R0          VRRX1: VH     Normal          11993 mV          na
R0          VRRX2: VX1    Normal          853   mV          na
R0          VRRX2: VX4    Normal          1016  mV          na
R0          VRRX2: VX5    Normal          1019  mV          na
R0          VRRX2: VP1    Normal          3325  mV          na
R0          VRRX2: VP3    Normal          1826  mV          na
R0          VRRX2: VP4    Normal          1050  mV          na
R0          VRRX2: VH     Normal          11987 mV          na
R0          VRRX3: VX1    Normal          994   mV          na
R0          VRRX3: VX2    Normal          1002  mV          na
R0          VRRX3: VX4    Normal          750   mV          na
R0          VRRX3: VX5    Normal          751   mV          na
R0          VRRX3: VP1    Normal          2477  mV          na
R0          VRRX3: VP2    Normal          1197  mV          na
R0          VRRX3: VP3    Normal          1517  mV          na
R0          VRRX3: VP4    Normal          1514  mV          na
R0          VRRX3: VH     Normal          11987 mV          na
R0          Temp: RCRX IN Normal          26    Celsius
(52 ,57 ,62 ,73 ) (Celsius)
R0          Temp: RCRX OUT Normal          41    Celsius
(62 ,67 ,72 ,80 ) (Celsius)
R0          Temp: Yoda     Normal          47    Celsius
(71 ,76 ,81 ,90 ) (Celsius)
R0          Temp: XEPhy    Normal          49    Celsius
(110,120,130,140) (Celsius)
R0          Temp: CPU Die  Normal          47    Celsius
(61 ,66 ,71 ,80 ) (Celsius)
R0          Temp: FC FANS  Fan Speed 40% 26    Celsius
(36 ,44 ,0 ) (Celsius)

```

- スタンバイコントローラのインターフェイスステータスを取得する場合は、次の CLI を使用できません。

show ip interface brief

Eg.

9800-stby#show ip int brief

```

Interface          IP-Address      OK? Method Status          Protocol
GigabitEthernet1   unassigned     YES unset  down           down
GigabitEthernet0   unassigned     YES NVRAM  administratively down  down
Capwap1            unassigned     YES unset  up             up
Capwap2            unassigned     YES unset  up             up
Capwap3            unassigned     YES unset  up             up
Capwap4            unassigned     YES unset  up             up
Capwap5            unassigned     YES unset  up             up
Capwap6            unassigned     YES unset  up             up
Capwap7            unassigned     YES unset  up             up
Capwap8            unassigned     YES unset  up             up
Capwap9            unassigned     YES unset  up             up
Capwap10           unassigned     YES unset  up             up

```

RMI を使用したスタンバイのモニタリング

Vlan1	unassigned	YES	NVRAM	down	down
Vlan56	unassigned	YES	unset	down	down
Vlan111	111.1.1.85	YES	NVRAM	up	up

1. スタンバイの IOS タスク CPU を確認するには、CLI の **show processes** を使用します。

```
9800-stby#show processes ?
<1-2147483647> IOS(d) Process Number
cpu           Show CPU usage per IOS(d) process
heapcheck    Show IOS(d) scheduler heapcheck configuration
history      Show ordered IOS(d) process history
memory       Show memory usage per IOS(d) process
platform     Show information per IOS-XE process
timercheck   Show IOS(d) processes configured for timercheck
|           Output modifiers
<cr>        <cr>
```

スタンバイ モニタリング プログラム インターフェイス

スタンバイコントローラの CPU、メモリ、およびインターフェイスのステータスは、プログラムインターフェイスでモニタできます。この目的に必要な運用モデルのリストを次に示します。

- **Cisco-IOS-XE-device-hardware-oper.yang** : デバイスのすべての FRU (シャーンを含む) にシリアル番号があります。また、システム内のすべてのハードウェアに関する情報もあります。
 - **Cisco-IOS-XE-process-cpu-oper.yang** : 過去 1 分、5 分、5 秒の間隔の平均 CPU 使用率、および IOS タスクのプロセス単位の CPU 統計があります。
 - **Cisco-IOS-XE-platform-software-oper.yang** : 5 秒間隔の平均 CPU 使用率とプロセスに割り当てられたメモリを示します。
2. **Cisco-IOS-XE-process-memory-oper.yang** : プロセス単位のメモリ使用率を示します。
 - **Cisco-IOS-XE-interfaces-oper.yang** : 状態と統計情報を含むインターフェイスの運用データがあります。インターフェイスに関するその他の運用データも多数あります。

RMI IPv4 への SSH 接続を使用してスタンバイコントローラをモニタ する手順

1. アクティブコントローラ上で SSH を有効にします。これを実行するには、RSA キーを生成する必要があります。

```
9800(config)#crypto key generate rsa
% You already have RSA keys defined named ak_vewlc_small.cisco.com.
% Do you really want to replace them? [yes/no]: yes
Choose the size of the key modulus in the range of 2048 to 4096 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

How many bits in the modulus [2048]: 2048
% Generating 2048 bit RSA keys, keys will be non-exportable...
[OK] (elapsed time was 0 seconds)
9800(config)#
```

RMI を使用したスタンバイのモニタリング

次に示すように、ローカル AAA または外部 AAA (RADIUS) をローカル AAA フォールバックで設定します。

```
line vty 0 4
password Cisco
authorization exec DEVICE_ADMIN
login authentication DEVICE_ADMIN
length 0
transport input ssh

line vty 5 15
password Cisco
authorization exec DEVICE_ADMIN
login authentication DEVICE_ADMIN
transport input telnet ssh
transport output telnet ssh

aaa authentication login DEVICE_ADMIN group AAA_GROUP_ISE1 local
aaa authorization exec DEVICE_ADMIN group AAA_GROUP_ISE1 local
aaa group server radius AAA_GROUP_ISE1
server name ISE1
radius server ISE1
address ipv4 <RMI IP> auth-port 1812 acct-port 1813
key <key>
```

注：TACACS はスタンバイではサポートされません。方式リストに「LOCAL」が追加されていることを確認します。したがって、ユーザーはスタンバイではローカルに認証されます。

```
aaa authentication login VTY_authen_tacacs group tacacs_ise_group local
aaa authentication login VTY_authen_tacacs group tacacs_ise_group local
```

2. 管理 VLAN のデフォルトルートが設定されていることを確認します。

```
ip route <Destination prefix> <Destination prefix mask> <Forwarding router's address>
```

3. スタンバイコントローラの RMI IP アドレスを使用してスタンバイコントローラにログインします。

```
ssh <username>@<RMI IP>
Password:
```

注：Netconf-YANG SSH を使用するには、次のコマンドを使用します。

```
ssh <username>@<RMI IP> -p 830
```

Netconf-YANG SSH では、デフォルトポート 830 のみを使用できます。

4. **show environment summary**、**show processes**、**show ip interface brief** コマンドを実行して、CPU、メモリ、インターフェイスステータス、PSU (電源装置) 障害、ファン障害、および温度を表示します。

RESTCONF を使用するスタンバイモニタリングのコマンド

GET 要求 :

```
curl --request GET --url https://<Standby RMI IP>:443/restconf/data/Cisco-IOS-XE-native:native/hostname --header 'accept: application/yang-data+json' --header 'cache-control: no-cache' --header 'content-type: application/yang-data+json' -k -u username:password
```

例 :

```
$curl --request GET --url https://<Standby RMI IP>:443/restconf/data/Cisco-IOS-XE-native:native/hostname --header 'accept: application/yang-data+json' --header 'cache-control: no-cache' --header 'content-type: application/yang-data+json' -k -u username:password  
{  
  "Cisco-IOS-XE-native:hostname": "Catalyst 9800 HA2"  
}
```

PUT 要求はスタンバイではサポートされず、access-denied エラーが返されます。

リリース 17.5 のスタンバイモニタリング

このリリースには、次のような機能強化が含まれています。

1. スタンバイコントローラでの直接モニタリング
2. アクティブコントローラからのスタンバイパラメータのモニタリング

以降のセクションでは、これらの両方について詳しく説明します。

スタンバイコントローラでの直接モニタリング

1. **IF-MIB** のサポート : この MIB はインターフェイス統計情報をモニタするために使用されます。

```
HA-stby#snmp get-bulk v2c <ip address> public n 0 m 1000 oid ifMIB
```

```
SNMP Response: reqid 1, errstat 0, erridx 0
```

```
ifName.1 = Gi1  
ifName.2 = Gi0  
ifName.3 = Vo0  
ifName.4 = Nu0  
ifName.5 = V11  
ifName.6 = V184  
ifName.7 = V1111  
ifName.8 = V1184
```

```
umang_ha-stby#shk v2c <ip address> public n 0 m 1000 oid ifAdminStatus
```

```
SNMP Response: reqid 2, errstat 0, erridx 0
```

```
ifAdminStatus.1 = 1  
ifAdminStatus.2 = 2  
ifAdminStatus.3 = 1  
ifAdminStatus.4 = 1
```

RMI を使用したスタンバイのモニタリング

```
ifAdminStatus.5 = 2
ifAdminStatus.6 = 1
ifAdminStatus.7 = 1
ifAdminStatus.8 = 1
ifOperStatus.1 = 2
ifOperStatus.2 = 2
ifOperStatus.3 = 1
ifOperStatus.4 = 2
ifOperStatus.5 = 2
ifOperStatus.6 = 2
ifOperStatus.7 = 2
ifOperStatus.8 = 1
```

注：スタンバイからのトラップはサポートされていません。

2. スタンバイシャーシですべてのセンサーをリストするための **show env all** の使用のサポート。
 - a) HA システムにイメージをロードします。
 - b) アクティブおよびスタンバイが稼働している状態で、スタンバイで **show env all** を実行します。

HA-stby#sh env all

```
Sensor List: Environmental Monitoring
Sensor Location State Reading
Temp: BRDTEMP1 R0 Normal 35 Celsius
Temp: BRDTEMP2 R0 Normal 33 Celsius
Temp: CPU Die R0 Normal 45 Celsius
```

3. 外部サーバーへのスタンバイ **syslog**

ユーザーが IOS SSH または NETCONF 経由でログインしていれば、スタンバイは外部 syslog サーバーに syslog を個別に送信できます。

- a) HA ペアを稼働させます。
- b) 外部ロギングサーバーを設定します。
(config)#logging host <ip>
- c) ユーザーが SSH または NETCONF SSH 経由でスタンバイにログインすると、スタンバイ IP から syslog を受信できるかどうかをロギングサーバーで確認します。

Netconf :

```
Sep 28 10:52:43.263: %DMI-5-AUTH_PASSED: Chassis 2 R0/0: dmiauthd: User
'asomesul' authenticated successfully from 64.104.149.222:58970 and was
authorized for netconf over ssh. External groups: PRIV15
```

RMI を使用したスタンバイのモニタリング

IOS SS :

Sep 28 10:56:02.163: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: asomesul] [Source: 9.4.16.172] [localport: 22] at 10:56:02 UTC Mon Sep 28 2020
Sep 28 10:56:04.183: %SYS-6-LOGOUT: User asomesul has exited tty session 1(9.4.16.172)



アクティブコントローラからのスタンバイパラメータのモニタリング

1. **CISCO-PROCESS-MIB** のサポート：この MIB は、CPU とプロセスの統計情報をモニタするために使用されます。これは、CPU/メモリ情報用です。

次の出力例では、**index 5** はシャーシ 1、**index 6** はシャーシ 2 です。

```
#$!k v2c <ip address> public n 0 m 1000 oid cpmProcessEntry.2
```

```
SNMP Response: reqid 16, errstat 0, erridx 0
```

```
cpmProcessEntry.2.5.2890 = linux_iosd-imag  
cpmProcessEntry.2.5.10111 = vman  
cpmProcessEntry.2.5.11712 = lman  
cpmProcessEntry.2.5.13447 = cmand  
cpmProcessEntry.2.5.17443 = cli_agent  
cpmProcessEntry.2.5.21398 = psd  
cpmProcessEntry.2.5.22986 = smand  
cpmProcessEntry.2.5.23265 = fman_fp_image  
cpmProcessEntry.2.5.23936 = repm  
cpmProcessEntry.2.5.24412 = plogd  
cpmProcessEntry.2.5.25914 = cman_fp  
cpmProcessEntry.2.5.26655 = hman  
cpmProcessEntry.2.5.26981 = fman_rp  
cpmProcessEntry.2.5.27625 = dbm  
cpmProcessEntry.2.6.10140 = vman  
cpmProcessEntry.2.6.11662 = lman  
cpmProcessEntry.2.6.13007 = cmand  
cpmProcessEntry.2.6.21071 = fman_fp_image  
cpmProcessEntry.2.6.23341 = psd  
cpmProcessEntry.2.6.23731 = cman_fp  
cpmProcessEntry.2.6.25148 = repm  
cpmProcessEntry.2.6.25424 = plogd  
cpmProcessEntry.2.6.26475 = hman  
cpmProcessEntry.2.6.26796 = fman_rp  
cpmProcessEntry.2.6.27369 = dbm  
cpmProcessEntry.2.6.27660 = cli_agent  
cpmProcessEntry.2.6.28153 = linux_iosd-imag  
cpmProcessEntry.2.6.30537 = smand  
cpmProcessEntry.4.5.2890 = 1  
cpmProcessEntry.4.5.10111 = 1  
cpmProcessEntry.4.5.11712 = 3  
cpmProcessEntry.4.5.13447 = 0  
cpmProcessEntry.4.5.17443 = 1  
cpmProcessEntry.4.5.21398 = 2  
cpmProcessEntry.4.5.22986 = 2  
cpmProcessEntry.4.5.23265 = 2  
cpmProcessEntry.4.5.23936 = 0  
cpmProcessEntry.4.5.24412 = 3  
cpmProcessEntry.4.5.25914 = 0  
cpmProcessEntry.4.5.26655 = 3  
cpmProcessEntry.4.5.26981 = 1  
cpmProcessEntry.4.5.27625 = 3  
cpmProcessEntry.4.6.10140 = 2
```

RMI を使用したスタンバイのモニタリング

```
cpmProcessEntry.4.6.11662 = 2
cpmProcessEntry.4.6.13007 = 2
cpmProcessEntry.4.6.21071 = 3
cpmProcessEntry.4.6.23341 = 3
cpmProcessEntry.4.6.23731 = 2
cpmProcessEntry.4.6.25148 = 3
cpmProcessEntry.4.6.25424 = 1
cpmProcessEntry.4.6.26475 = 1
```

2. **CISCO-LWAPP-HA-MIB** のサポート : この MIB は、SSO に関連する HA パラメータをモニタするために使用されます。

サポート対象フィールド :

9800-HA#\$!k v2c 1.1.1.1 public n 0 m 1000 oid ciscoLwappHaMIB

```
SNMP Response: reqid 62, errstat 0, erridx 0
cLMcHaPortName.0 = GigabitEthernet2
cLMcHaPortLocIpAddrType.0 = 1
cLMcHaPortLocIp.0 = A9 FE B8 4E
cLMcHaPortMask.0 = FF FF FF 00
cLMcHaPortRemotIpAddrType.0 = 1
cLMcHaPortRemIp.0 = A9 FE B8 4F
cLMcHaKeepAliveTimeOut.0 = 100
cLMcHaChassisPriority.0 = 1
cLMcHaClearConfig.0 = 2
cLMcHaKeepAliveRetryCount.0 = 5
cLMcRmiConfigAction.0 = 1
cLMcRmiInterface.0 = Vlan184
cLMcRmiChassisANum.0 = 1
cLMcRmiChassisAIpAddrType.0 = 1
cLMcRmiChassisAIp.0 = 09 07 B8 4E
cLMcRmiChassisBNum.0 = 2
cLMcRmiChassisBIpAddrType.0 = 1
cLMcRmiChassisBIp.0 = 09 07 B8 4F
cLMcRmiGatewayFailover.0 = 1
cLMcRmiGatewayFailoverInterval.0 = 8
```

3. **cLHaPeerHotStandbyEvent** のサポート : このオブジェクトは、ピアがホットスタンバイになったことを表します。

- a) HA ペアを稼働させます。
- b) スタンバイをリロードします。
- c) アクティブから次を実行します。 `snmp get v2c <ip address> public oid cLHaPeerHotStandbyEvent.0`
- d) スタンバイはホットではないため、アクティブは 0 を返します。
- e) スタンバイがリロードしてスタンバイがホットになった後は、再び 1 に変更されます。

RMI を使用したスタンバイのモニタリング

スタンバイのリロード後

```
#snmp get v2c <ip address> public oid cLHaPeerHotStandbyEvent.0  
SNMP Response: reqid 31, errstat 0, erridx 0  
cLHaPeerHotStandbyEvent.0 = 0
```

スタンバイホットへのリロード後

```
#snmp get v2c <ip address> public oid cLHaPeerHotStandbyEvent.0  
SNMP Response: reqid 33, errstat 0, erridx 0  
cLHaPeerHotStandbyEvent.0 = 1
```

4. **cLHaBulkSyncCompleteEvent** のサポート：このオブジェクトは、一括同期が完了した時刻を表します。

- HA ペアを稼働させます。
- スタンバイをリロードします。
- アクティブから次を実行します。 `snmp get <ip address> public oid cLHaBulkSyncCompleteEvent.0`
- スタンバイはホットではないため、アクティブは 0 を返します。
- スタンバイがリロードされ、スタンバイがホットになった後に、一括同期が再度更新されます。

スタンバイリロード後

```
#snmp get <ip address> public oid cLHaBulkSyncCompleteEvent.0  
SNMP Response: reqid 30, errstat 0, erridx 0  
cLHaBulkSyncCompleteEvent.0 = 0
```

スタンバイからスタンバイホットへ

```
#snmp get v2c <ip address> public oid cLHaBulkSyncCompleteEvent.0  
SNMP Response: reqid 32, errstat 0, erridx 0  
cLHaBulkSyncCompleteEvent.0 = 1601288785
```

5. **show env** コマンドを使用して、アクティブおよびスタンバイのセンサーを一覧表示します。

- HA システムにイメージをロードします。
- アクティブおよびスタンバイが稼働したら、アクティブで `show env all` を実行します。

```
#sh env all  
Sensor List: Environmental Monitoring  
Sensor          Location      State      Reading
```

RMI を使用したスタンバイのモニタリング

```
Temp: BRDTEMP1      R0      Normal      39 Celsius
Temp: BRDTEMP2      R0      Normal      36 Celsius
Temp: CPU Die       R0      Normal      47 Celsius
Stby Temp: BRDTEMP1  R0      Normal      36 Celsius
Stby Temp: BRDTEMP2  R0      Normal      33 Celsius
Stby Temp: CPU Die   R0      Normal      46 Celsius
```

6. show env chassis <standby chassis num> r0 コマンドを使用して、スタンバイセンサーを一覧表示します。

- a) HA システムにイメージをロードします。
- b) アクティブおよびスタンバイが稼働したら、アクティブで show env chassis <standby chassis num> r0 を実行します。スタンバイからアクティブコントローラのセンサーを表示するには、アクティブシャーシのシャーシ番号を指定して同じコマンドを使用します。

```
#sh env cha 2 r0
Sensor List: Environmental Monitoring
Sensor Location State Reading
Stby Temp: BRDTEMP1 R0 Normal 35 Celsius
Stby Temp: BRDTEMP2 R0 Normal 33 Celsius
Stby Temp: CPU Die R0 Normal 45 Celsius
```

7. YANG を使用してアクティブおよびスタンバイのセンサーを取得します。

- a) HA システムにイメージをロードします。
- b) NETCONF on the CISCO-IOS-XE-environment-oper を実行します。

以下は、この情報を取得するために使用する xpath モジュールです。

```
Module    Cisco-IOS-XE-environment-oper
Revision  2019-05-01
Revision Info  Added semantic version
Description  This module contains a collection of YANG definitions
for monitoring Environment of a Network Element.
Copyright (c) 2015 by Cisco Systems, Inc.
All rights reserved.
Organization  Cisco Systems, Inc.
Imports
"ciscon-semver"
Namespace   http://cisco.com/ns/yang/Cisco-IOS-XE-environment-oper
Prefix environment-ios-xe-oper
Namespace Prefixes
ciscon-semver http://cisco.com/ns/yang/ciscon-semver
environment-ios-xe-oper http://cisco.com/ns/yang/Cisco-IOS-XE-environment-oper
Modtype     module
Operations
"get"
```

次に、参考のために yang の出力を示します。

```
Table Record Index 0 = {
[0] state = Normal
[1] current_reading = 35
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 60
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 53
[6] high_critical_threshold = 60
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Temp: BRDTEMP1
[9] location = R0
}
Table Record Index 1 = {
[0] state = Normal
[1] current_reading = 33
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 64
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 57
[6] high_critical_threshold = 64
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Temp: BRDTEMP2
[9] location = R0
}
Table Record Index 2 = {
[0] state = Normal
[1] current_reading = 45
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 104
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 93
[6] high_critical_threshold = 104
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Temp: CPU Die
[9] location = R0
}
Table Record Index 3 = {
[0] state = Normal
[1] current_reading = 39
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 60
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 53
[6] high_critical_threshold = 60
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Stby Temp: BRDTEMP1
[9] location = R0
}
```

RMI を使用したスタンバイのモニタリング

```
Table Record Index 4 = {
[0] state = Normal
[1] current_reading = 36
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 64
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 57
[6] high_critical_threshold = 64
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Stby Temp: BRDTEMP2
[9] location = R0
}
Table Record Index 5 = {
[0] state = Normal
[1] current_reading = 47
[2] sensor_units = SENSOR_UNIT_CELSIUS
[3] low_critical_threshold = 104
[4] low_normal_threshold = -2147483647
[5] high_normal_threshold = 93
[6] high_critical_threshold = 104
[7] sensor_name = SENSOR_TYPE_TEMPERATURE
[8] name = Stby Temp: CPU Die
[9] location = R0
}
```

8. **snmpwalk** を使用して、アクティブおよびスタンバイの電源、ファン、および RP センサーの情報を取得します。

- a) HA システムにイメージをロードします。
- b) スタンバイが参加したら、任意の Linux マシンから **snmpwalk** を実行します。
- c) CLI **show inventory raw** を実行し、スタンバイとアクティブのセンサーを確認します。
- d) エンティティ MIB およびセンサー MIB で **snmpwalk** を実行し、電源、ファン、および RP センサーが CLI出力の値と一致することを確認します。

次に、参考のために **snmpwalk** の出力を示します。

```
SNMPv2-SMI::mib-2.47.1.1.1.1.7.1 = STRING: "Multi Chassis System"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.2 = STRING: "Chassis 1"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.3 = STRING: "Chassis 1 Power Supply Bay 0"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.4 = STRING: "Chassis 1 Power Supply Module 0"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.5 = STRING: "Vin P0/0"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.6 = STRING: "Iin P0/1"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.7 = STRING: "Vout P0/2"
SNMPv2-SMI::mib-2.47.1.1.1.1.7.8 = STRING: "Iout P0/3"
```

RMI を使用したスタンバイのモニタリング

```
SNMPv2-SMI::mib-2.47.1.1.1.1.7.9 = STRING: "Temp1 P0/4"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.10 = STRING: "Temp2 P0/5"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.11 = STRING: "Temp3 P0/6"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.14 = STRING: "Chassis 1 Power Supply 0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.23 = STRING: "Chassis 1 Power Supply Bay 1"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.44 = STRING: "Chassis 1 Fan Tray"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.55 = STRING: "Chassis 1 Fan 2/0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.56 = STRING: "Chassis 1 Fan 2/1"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.57 = STRING: "Chassis 1 Fan 2/2"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.58 = STRING: "Chassis 1 Fan 2/3"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.500 = STRING: "Chassis 2"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.501 = STRING: "Chassis 2 Power Supply Bay 0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.502 = STRING: "Chassis 2 Power Supply Module 0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.503 = STRING: "Stby Vin P0/0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.504 = STRING: "Stby lin P0/1"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.505 = STRING: "Stby Vout P0/2"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.506 = STRING: "Stby Iout P0/3"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.507 = STRING: "Stby Temp1 P0/4"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.508 = STRING: "Stby Temp2 P0/5"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.509 = STRING: "Stby Temp3 P0/6"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.512 = STRING: "Chassis 2 Power Supply 0"  
SNMPv2-SMI::mib-2.47.1.1.1.1.7.521 = STRING: "Chassis 2 Power Supply Bay 1"
```

SNMP を使用したアクティブを通したスタンバイコントローラのインターフェイスステータス

17.6 では、CISCO_LWAPP-HA-MIB の stby インターフェイスエントリに関連する次の MIB オブジェクトがサポートされています。

- stbyIfIndex
- stbyIfName
- stbyIfPhysAddress

RMI を使用したスタンバイのモニタリング

- stbyifOperStatus
- stbyifAdminStatus

アクティブコントローラのワイヤレス管理インターフェイス、冗長性管理インターフェイス、およびサービスポート（デバイス管理インターフェイス）で、SNMP を使用できます。

MIB	MIB オブジェクト	説明	OID
CISCO-LWAPP-HA-MIB	stbyIfTable	インターフェイスエントリのリスト	1.3.6.1.4.1.9.9.843.1.8.1
CISCO-LWAPP-HA-MIB	stbyIfEntry	特定のインターフェイスに該当する管理情報を含むエントリ	1.3.6.1.4.1.9.9.843.1.8.1.1
CISCO-LWAPP-HA-MIB	stbyIfIndex	各インターフェイスに固有のゼロより大きい値。値は、1 から開始され連続的に割り当てられていることが推奨されます。各インターフェイスサブレイヤの値は、エンティティのネットワーク管理システムの少なくとも 特定の再初期化から次の再初期化まで、一貫性を維持している必要があります。	1.3.6.1.4.1.9.9.843.1.8.1.1.1
CISCO-LWAPP-HA-MIB	stbyIfName	インターフェイスのテキスト名このオブジェクトの値は、ローカルデバイスによって割り当てられたインターフェイス名であり、デバイスのコンソールに入力されるコマンドでの使用に適しています。これは、「le0」などのテキスト名、または「1」などの単純なポート番号であり、デバイスのインターフェイス名指定シンタックスに応じて異なります。ifTable 内の複数のエントリが同時に、デバイスにより名前が指定された単一のインターフェイスを表している場合、各エントリは同じ値のifName を持ちます。その他の（プロキシされた）デバイス上の特定インターフェイスに関する SNMP クエリに回答するエージェントの場合、このようなインターフェイスのifName の値は、プロキシされたデバイスのローカル名となります。ローカル名がない場合や、いかなる	1.3.6.1.4.1.9.9.843.1.8.1.1.2

RMI を使用したスタンバイのモニタリング

		方法でもこのオブジェクトが適用されない場合、このオブジェクトには長さが 0 の文字列が格納されます。	
CISCO-LWAPP-HA-MIB	stbyifPhysAddress	プロトコルサブレイヤにおけるインターフェイスのアドレス。たとえば、802.x インターフェイスの場合、通常このオブジェクトは MAC アドレスを含みます。インターフェイスのメディア固有の MIB は、このオブジェクトの値のビット順とバイト順、および形式を定義しています。このようなアドレスを持たないインターフェイス（シリアル回線など）の場合、このオブジェクトにはゼロ長のオクテット文字列が含まれます。	1.3.6.1.4.1.9.9.843.1.8.1.1.3
CISCO-LWAPP-HA-MIB	stbyifAdminStatus	<p>インターフェイスの適切な状態。testing(3) 状態は、渡すことができる動作パケットがないことを示します。管理対象システムが初期化されると、すべてのインターフェイスは ifAdminStatus が down(2) 状態で開始します。明示的な管理アクションの結果として、または管理対象システムにより保持されている設定情報に従って、ifAdminStatus は up(1) または testing(3) 状態に変わります（または down(2) 状態を維持します）。</p> <p>これは、アクティブの実際の設定に基づいています。 （たとえば、管理上ダウンしている場合は 2 に、その他の場合は、アクティブとスタンバイが 2 になります）</p>	1.3.6.1.4.1.9.9.843.1.8.1.1.4.
CISCO-LWAPP-HA-MIB	stbyifOperStatus	インターフェイスの現在の動作状態。testing(3) 状態は、渡すことができる動作パケットがないことを示します。ifAdminStatus が down(2) の場合、ifOperStatus は down(2) となります。ifAdminStatus が up(1) に変化した場合、インターフェイスでネットワークトラフィックを送受信できる状態であれば ifOperStatus	1.3.6.1.4.1.9.9.843.1.8.1.1.5

RMI を使用したスタンバイのモニタリング

		<p>は up(1) に変わります。インターフェイスが外部アクションを待機している場合 (シリアル回線が着信接続を待機している場合など) は、ifOperStatus は dormant(5) に変わります。インターフェイスが up(1) 状態に移行するのを妨げる障害が存在する場合に限り、ifOperStatus は down(2) 状態を維持します。インターフェイスのコンポーネント (通常はハードウェア) が欠落している場合、ifOperStatus は notPresent(6) 状態を維持します。</p> <p>これは、インターフェイスの実際の状態に基づきます。</p>	
--	--	--	--

注 : MTU、Ifspeed、ifLastChange に関連するスタンバイ インターフェイス データはサポートされていません

次に、スタンバイ インターフェイス エンティティの例を示します。

```

WLC#snmp get-bulk v2c 111.1.1.215 public n 0 m 1000 oid stbyIfEntry
SNMP Response: reqid 1, errstat 0, erridx 0
stbyIfIndex.1 = 1
stbyIfIndex.2 = 2
stbyIfIndex.3 = 3
stbyIfIndex.4 = 4
stbyIfIndex.5 = 5
stbyIfIndex.6 = 6
stbyIfIndex.7 = 7
stbyIfIndex.8 = 8
stbyIfIndex.9 = 9
stbyIfIndex.10 = 10
stbyIfName.1 = Gi1
stbyIfName.2 = Gi0
stbyIfName.3 = Vo0
stbyIfName.4 = Nu0
stbyIfName.5 = V11
stbyIfName.6 = Lo0
stbyIfName.7 = V1111
stbyIfName.8 = V1333
stbyIfName.9 = V1444
stbyIfName.10 = V1555
stbyIfPhysAddress.1 = 00 0C 29 23 DB 56
stbyIfPhysAddress.2 = 00 00 00 00 00 00
  
```

RMI を使用したスタンバイのモニタリング

```
stbyifPhysAddress.3 =  
stbyifPhysAddress.4 =  
stbyifPhysAddress.5 = 00 1E 7A 4C 89 FF  
stbyifPhysAddress.6 =  
stbyifPhysAddress.7 = 00 1E 7A 4C 89 FF  
stbyifPhysAddress.8 = 00 1E 7A 4C 89 FF  
stbyifPhysAddress.9 = 00 1E 7A 4C 89 FF  
stbyifPhysAddress.10 = 00 1E 7A 4C 89 FF  
stbyifAdminStatus.1 = 1  
stbyifAdminStatus.2 = 2  
stbyifAdminStatus.3 = 1  
stbyifAdminStatus.4 = 1  
stbyifAdminStatus.5 = 1  
stbyifAdminStatus.6 = 2  
stbyifAdminStatus.7 = 1  
stbyifAdminStatus.8 = 1  
stbyifAdminStatus.9 = 1  
stbyifAdminStatus.10 = 1  
stbyifOperStatus.1 = 1  
stbyifOperStatus.2 = 2  
stbyifOperStatus.3 = 1  
stbyifOperStatus.4 = 1  
stbyifOperStatus.5 = 1  
stbyifOperStatus.6 = 2  
stbyifOperStatus.7 = 1  
stbyifOperStatus.8 = 2  
stbyifOperStatus.9 = 2  
stbyifOperStatus.10 = 2
```

RMI を使用したスタンバイのモニタリング

法的情報

このマニュアルに記載されている仕様および製品に関する情報は、予告なしに変更されることがあります。このマニュアルに記載されている表現、情報、および推奨事項は、すべて正確であると考えていますが、明示的であれ黙示的であれ、一切の保証の責任を負わないものとします。このマニュアルに記載されている製品の使用は、すべてユーザー側の責任になります。

対象製品のソフトウェアライセンスおよび限定保証は、製品に添付された『Information Packet』に記載されています。添付されていない場合には、代理店にご連絡ください。

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

ここに記載されている他のいかなる保証にもよらず、各社のすべてのマニュアルおよびソフトウェアは、障害も含めて「現状のまま」として提供されます。シスコおよびこれら各社は、商品性の保証、特定目的への準拠の保証、および権利を侵害しないことに関する保証、あるいは取引過程、使用、取引慣行によって発生する保証をはじめとする、明示されたまたは黙示された一切の保証の責任を負わないものとします。

いかなる場合においても、Cisco およびその供給者は、このマニュアルに適用できるまたは適用できないことによって、発生する利益の損失やデータの損傷をはじめとする、間接的、派生的、偶発的、あるいは特殊な損害について、あらゆる可能性が Cisco またはその供給者に知らされていても、それらに対する責任を一切負わないものとします。

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワークポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

ハード コピーおよびソフト コピーの複製は公式版とみなされません。最新版はオンライン版を参照してください。

Cisco は世界各国 200 箇所にオフィスを開設しています。各オフィスの住所、電話番号、FAX 番号については、Cisco のウェブサイト www.cisco.com/go/offices をご覧ください。

Cisco の商標

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Cisco 著作権

© 2020 Cisco Systems, Inc. All rights reserved.