ファイアウォール ステートフル シャーシ間冗長性の設定

ファイアウォール ステートフル シャーシ間冗長性機能を使用すると、相互にバックアップとして動作するルータのペアを設定できます。この機能を設定し、複数のフェールオーバー条件に基づいてアクティブ ルータを判断できます。フェールオーバーが発生すると、中断なくスタンバイ ルータが引き継ぎ、トラフィック フォワーディング サービスの実行とダイナミック ルーティング テーブルのメンテナンスを開始します。

ファイアウォール ステートフル シャーシ間冗長性の前提条件

- ファイアウォールに接続しているインターフェイスは、同じ冗長インターフェイス識別子（RII）を持つ必要があります。
- アクティブ デバイスおよびスタンバイ デバイスは、Cisco IOS XE ゾーンベース ファイアウォールの設定を同じにする必要があります。
- アクティブ デバイスとスタンバイ デバイスは、同じバージョンの Cisco IOS XE ソフトウェアで実行する必要があります。アクティブ デバイスとスタンバイは、スイッチを介して接続する必要があります。
- 組み込みサービス プロセッサ（ESP）は、アクティブ デバイスとスタンバイ デバイスの両方で一致する必要があります。

ファイアウォールステートフルシャーシ間冗長性に関する制約事項

- LAN および MESH シナリオはサポートされません。
- ボックス間の高可用性（HA）とボックス内の HA の共存はサポートされていないので、シャーシ内にデュアル エンベデッド サービス プロセッサ（ESP）またはデュアル ルート プロセッサ（RP）を持つCisco ASR 1006およびCisco ASR 1013プラットフォームはサポートされていません。

  シャーシ内に単一の ESP と単一の RP を持つ Cisco ASR 1006 および Cisco ASR 1013 プラットフォームは、シャーシ間冗長性をサポートします。
- デュアル IOS デーモン（IOSd）が設定されている場合、デバイスはファイアウォール ステートフル シャーシ間冗長性の設定をサポートしません。

ファイアウォールステートフルシャーシ間冗長性について

ファイアウォール ステートフル シャーシ間冗長性の機能

相互にホット スタンバイとして動作するようにルータのペアを設定できます。この冗長性は、インターフェイスベースで設定します。冗長インターフェイスのペアは、冗長グループと呼ばれます。次の図に、アクティブ/スタンバイ デバイスのシナリオを示します。また、1 つの発信インターフェイスを持つルータのペアについて、冗長グループを設定する方法を示します。アクティブ/アクティブ デバイス シナリオを表現する「冗長グループの設定：*2* つの発信インターフェイス」の図に、2 つの発信インターフェイスを使用するルータのペアに 2 つの冗長グループを設定する方法を示します。

いずれの場合でも、設定可能なコントロール リンクおよびデータ同期リンクによって冗長ルータは参加します。コントロール リンクは、ルータのステータスを通信するために使用されます。データ同期リンクは、ネットワーク アドレス変換（NAT）およびファイアウォールからステートフル情報を転送し、これらのアプリケーションについてステートフルデータベースを同期するために使用されます。

また、いずれの場合でも、冗長インターフェイスのペアは、同じ固有ID番号（RIIと呼ばれます）で設定されます。

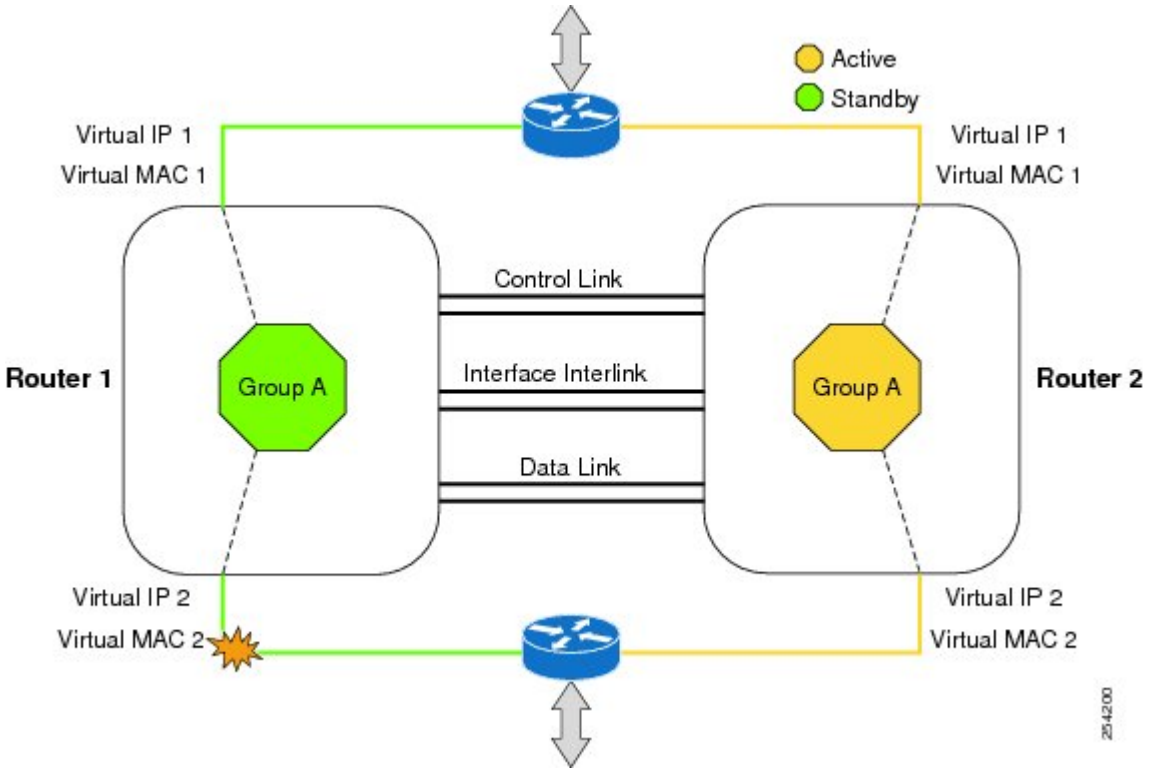
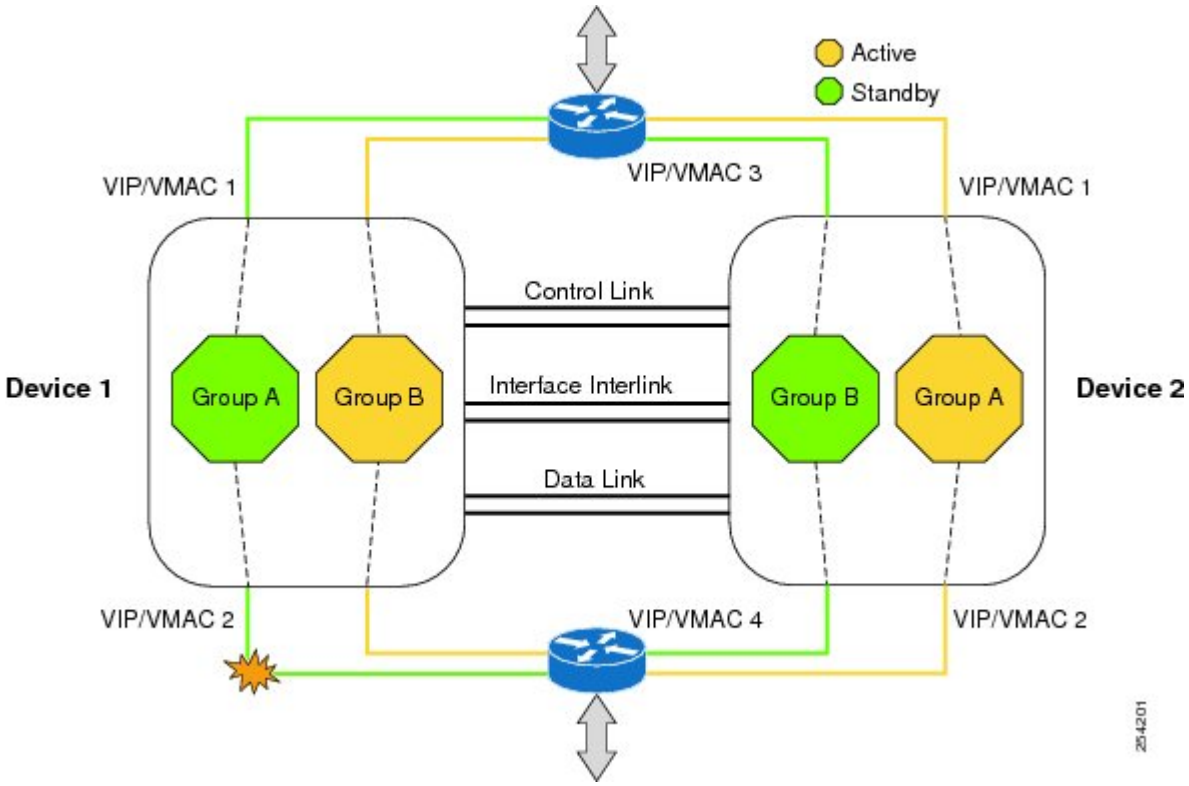


図 *1* : 冗長グループの設定：*2* つの発信インターフェイス



冗長グループ メンバーのステータスは、コントロール リンクで送信される hello メッセージを使用することで判断できます。設定可能な時間内に、いずれかのルータが hello メッセージに応答しない場合、エラーが発生したと見なされ、スイッチオーバーが開始されます。ミリ秒単位でエラーを検出するには、双方向フォワーディング検出（BFD）プロトコルと統合されたフェールオーバー プロトコルをコントロール リンクで実行します。hello メッセージについて次のパラメータを設定できます。

- Active timer
- Standby timer
- hellotime：hello メッセージが送信される間隔
- holdtime：アクティブまたはスタンバイ ルータをダウン状態と宣言するまでの時間

hellotime は、Hot Standby Router Protocol（HSRP）に合わせてデフォルトで 3 秒に設定されます。holdtime のデフォルトは 10 秒です。これらのタイマーは、**timers hellotime msec** コマンドを使用してミリ秒単位で設定することもできます。

スイッチオーバーの影響を受けるインターフェイスのペアを判断するには、冗長インターフェイスの各ペアについて、固有の ID 番号を設定する必要があります。この ID 番号は RII と呼ばれ、インターフェイスに関連付けられています。

また、スタンバイ ルータに対するスイッチオーバーは、他の条件でも発生する可能性があります。スイッチオーバーが発生する別の要因として、各ルータで設定可能な優先順位設定があります。最も優先度が高いルータがアクティブ ルータになります。アクティブ ルータまたはスタンバイ ルータで障害が発生した場合、重みと呼ばれる設定可能な数値分、ルータの優先度が減らされます。アクティブ ルータの優先度が、スタンバイ ルータの優先度を下回る場合、スイッチオーバーが発生し、スタンバイ ルータがアクティブ ルータになります。このデフォルトの動作を無効にするには、冗長グループについて preemption 属性をディセーブルにします。また、インターフェイスの L1 状態がダウン状態になった場合、各インターフェイスを設定して優先度を減らします。この数は、冗長グループに設定されているデフォルトの値よりも優先されます。

冗長グループの優先度が変更されるエラー イベントごとに、タイム スタンプ、影響を受けた冗長グループ、以前の優先度、新しい優先度、およびエラー イベントの原因の説明を含む syslog エントリが生成されます。

スイッチオーバーが発生する原因となるもう 1 つの状況は、ルータまたはインターフェイスの優先度が、設定可能なしきい値レベルを下回る場合です。

一般的に、スタンバイ ルータへのスイッチオーバーは次の条件で発生します。

- アクティブ ルータで停電またはリロードが発生した場合（クラッシュも含まれます）。
- アクティブ ルータのランタイム優先度が、スタンバイ ルータの優先度を下回った場合。
- アクティブ ルータのランタイム優先度が、設定したしきい値を下回った場合。
- アクティブルータの冗長グループを手動でリロードするには、**redundancy application reload group** *rg-number* コマンドを使用します。

- 任意のモニタ対象インターフェイスで 2 つの連続する hello メッセージに失敗した場合、インターフェイスは強制的にテストモードになります。この問題が発生すると、いずれのユニットもまずインターフェイス上のリンク ステータスを確認してから、次のテストを実行します。
  - ネットワーク アクティビティ テスト
  - ARP テスト
  - ブロードキャスト ping テスト

ファイアウォール ステートフル シャーシ間冗長性機能では、冗長グループのトラフィックは、その冗長グループの入力インターフェイスに関連付けられている仮想 IP アドレスを使用してルーティングされます。仮想 IP アドレスに送信されたトラフィックは、冗長グループがアクティブ状態になっているルータで受信されます。冗長グループのフェールオーバー中は、仮想 IP アドレスへのトラフィックが新しくアクティブになった冗長グループに自動的にルーティングされます。

冗長グループのトラフィックがスタンバイ ルータの物理 IP アドレスを使用してルーティングされてスタンバイ冗長グループに到達した場合、ファイアウォールはスタンバイ冗長グループに到達したトラフィックをドロップします。一方、トラフィックがアクティブ冗長グループに到達した場合は、確立された TCP または UDP セッションがスタンバイ冗長グループに同期されます。

排他的仮想 IP アドレスと排他的仮想 MAC アドレス

仮想 IP（VIP）アドレスと仮想 MAC（VMAC）アドレスは、セキュリティ アプリケーションが、トラフィックを受信するインターフェイスを制御するために使用します。インターフェイスは別のインターフェイスとペアにされ、これらのインターフェイスは同じ冗長グループ（RG）に関連付けられます。アクティブな RG に関連付けられているインターフェイスは、VIP アドレスと VMAC を排他的に所有します。アクティブ デバイスの Address Resolution Protocol（ARP）プロセスによって、VIP への ARP 要求に対する ARP 応答が送信されます。また、インターフェイスのイーサネット コントローラは、VMAC 宛てのパケットを受信するようにプログラミングされます。RG のフェールオーバーが発生すると、VIP と VMAC の所有権は変化します。新しくアクティブになった RG に関連付けられたインターフェイスは、gratuitous ARP を送信し、インターフェイスのイーサネット コントローラをプログラミングして、VMAC 宛てのパケットを受け入れます。

IPv6 のサポート

各冗長グループ（RG）を、同じ冗長インターフェイス識別子（RII）で IPv4 と IPv6 の両方の仮想 IP（VIP）アドレスのトラフィック インターフェイスに割り当てることができます。各 RG は RII ごとに一意の仮想 MAC（VMAC）アドレスを使用します。RG では、IPv6 リンクローカル VIP とグローバル VIP がインターフェイス上に共存します。

トラフィック インターフェイス上の各 RG に対して IPv4 VIP、リンクローカル IPv6 VIP、および/またはグローバル IPv6 VIP を設定できます。IPv6 リンクローカル VIP は、スタティック ルートまたはデフォルト ルートを設定する場合に主に使用されます。IPv6 グローバル VIP は、LAN トポロジと WAN トポロジの両方で広く使用されています。

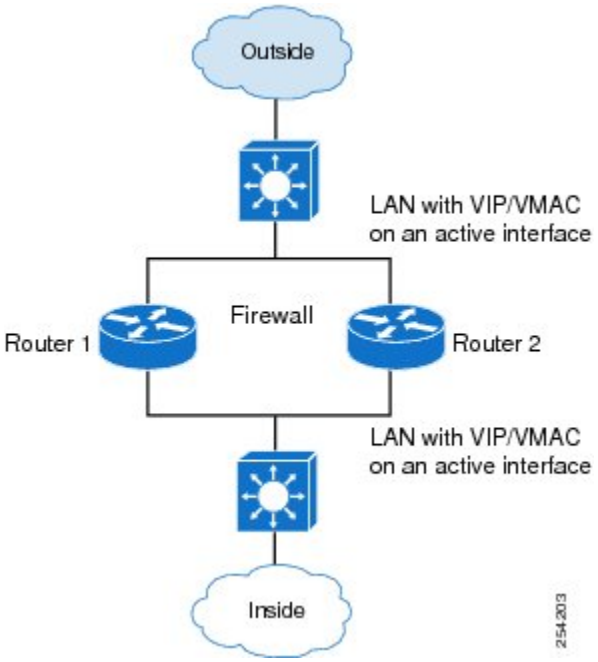
IPv4 VIP を設定する前に、物理 IP アドレスを設定する必要があります。

サポートされるトポロジ

LAN-LAN トポロジは、ファイアウォール ステートフル シャーシ間冗長性アーキテクチャでサポートされます。

LAN/LAN

次の図に、LAN/LAN トポロジを示します。専用のアプリケーションベースのファイアウォール ソリューションを使用するときに、アップストリームまたはダウンストリーム ルータから適切な仮想 IP アドレスへのスタティック ルーティングを設定することで、多くの場合、トラフィックは適切なファイアウォールに送信されます。さらに、Aggregation Services Router（ASR）は、アップストリームまたはダウンストリーム ルータとのダイナミック ルーティングに参加します。LAN 方向のインターフェイスでサポートされるダイナミック ルーティング構成では、ルーティング プロトコルのコンバージェンスへの依存が生じないようにしてください。依存があると、高速フェールオーバー要件に適合しなくなります。



LAN/LAN の設定の詳細については、「例：LAN/LAN の設定」を参照してください。

ゾーンベース ファイアウォールでの VRF 対応シャーシ間冗長性

Cisco IOS XE リリース 3.14S では、ゾーンベース ファイアウォールが VRF 対応シャーシ間冗長性をサポートします。アクティブ デバイスとスタンバイ デバイスの VPN ルーティングおよび転送（VRF）名は同じにする必要があります。アクティブ デバイスとスタンバイ デバイスの両方で同じ VRF 設定を使用できる必要があります。

ゾーンベース ファイアウォールでの VRF 対応シャーシ間冗長性機能では、アクティブ デバイスとスタンバイ デバイス上でボックスツーボックス ハイ アベイラビリティ セッション同期メッセージと一緒に VRF ハッシュ キーを送信する VRF マッピング機能が使用されます。

ファイアウォールステートフルシャーシ間冗長性の設定方法

冗長アプリケーション グループの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **application redundancy**
5. **group** *id*
6. **name** *group-name*
7. **shutdown**
8. **priority** *value* [**failover threshold** *value*]
9. **preempt**
10. **track** *object-number* {**decrement** *value* | **shutdown**}
11. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**redundancy** 例： `Device(config)# redundancy`	冗長コンフィギュレーションモードを開始します。
ステップ 4	**application redundancy** 例： `Device(config-red)# application redundancy`	冗長アプリケーションコンフィギュレーションモードを開始します。

	コマンドまたはアクション	目的
ステップ 5	**group** *id* 例： `Device(config-red-app)# group 1`	冗長アプリケーション グループ コンフィギュレーション モードを開始します。
ステップ 6	**name** *group-name* 例： `Device(config-red-app-grp)# name group1`	（任意）プロトコル インスタンスに任意のエイリアスを指定します。
ステップ 7	**shutdown** 例： `Device(config-red-app-grp)# shutdown`	（任意）冗長グループを手動でシャットダウンします。
ステップ 8	**priority** *value* [**failover threshold** *value*] 例： `Device(config-red-app-grp)# priority 100 failover threshold 50`	（任意）冗長グループの初期優先度とフェールオーバーしきい値を指定します。
ステップ 9	**preempt** 例： `Device(config-red-app-grp)# preempt`	グループでのプリエンプションをイネーブルにし、優先度とは無関係にスタンバイ デバイスがアクティブ デバイスをプリエンプション処理できるようにします。
ステップ 10	**track** *object-number* {**decrement** *value* | **shutdown**} 例： `Device(config-red-app-grp)# track 200 decrement 200`	冗長グループの優先度を指定します。この値は、イベントが発生した場合に減らされます。
ステップ 11	**end** 例： `Device(config-red-app-grp)# end`	冗長アプリケーション グループ コンフィギュレーション モードを終了して特権 EXEC モードを開始します。

冗長グループ プロトコルの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **application redundancy**
5. **protocol** *id*
6. **name** *group-name*
7. **timers hellotime** {*seconds* | **msec** *milliseconds*} **holdtime** {*seconds* | **msec** *milliseconds*}

8. **authentication** {**text** *string* | **md5 key-string** [**0** | **7**] *key-string* **timeout** *seconds* | **key-chain** *key-chain-name*}
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**redundancy** 例： `Device(config)# redundancy`	冗長コンフィギュレーション モードを開始します。
ステップ 4	**application redundancy** 例： `Device(config-red)# application redundancy`	冗長アプリケーション コンフィギュレーション モードを開始します。
ステップ 5	**protocol** *id* 例： `Device(config-red-app)# protocol 1`	コントロールインターフェイスに接続されるプロトコルインスタンスを指定し、冗長アプリケーション プロトコル コンフィギュレーション モードを開始します。
ステップ 6	**name** *group-name* 例： `Device(config-red-app-prtcl)# name prot1`	（任意）名前を使用して冗長グループ（RG）を設定します。
ステップ 7	**timers hellotime** {*seconds* | **msec** *milliseconds*} **holdtime** {*seconds* | **msec** *milliseconds*} 例： `Device(config-red-app-prtcl)# timers hellotime 3 holdtime 9`	hello メッセージが送信される間隔と、デバイスがダウン状態と宣言されるまでの時間を指定します。
ステップ 8	**authentication** {**text** *string* | **md5 key-string** [**0** | **7**] *key-string* **timeout** *seconds* | **key-chain** *key-chain-name*} 例： `Device(config-red-app-prtcl)# authentication md5 key-string 0 n1 timeout 100`	認証情報を指定します。

	コマンドまたはアクション	目的
ステップ 9	**end** 例： `Device(config-red-app-prtcl)# end`	冗長アプリケーションプロトコルコンフィギュレーションモードを終了し、特権EXECモードを開始します。

仮想 IP アドレスおよび冗長インターフェイス識別子の設定

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **redundancy rii** *id*
5. **redundancy group** *id* **ip** *virtual-ip* **exclusive** [**decrement** *value*]
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *type number* 例： `Device(config)# interface GigabitEthernet 0/1/1`	インターフェイスを設定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 4	**redundancy rii** *id* 例： `Device(config-if)# redundancy rii 600`	冗長グループ用に冗長インターフェイス識別子（RII）を設定します。 • 有効な範囲は 1 ～ 65535 です。
ステップ 5	**redundancy group** *id* **ip** *virtual-ip* **exclusive** [**decrement** *value*] 例： `Device(config-if)# redundancy group 1 ip 10.10.1.1 exclusive decrement 20`	インターフェイスを冗長グループに関連付け、仮想 IP アドレスをイネーブルにします。

	コマンドまたはアクション	目的
ステップ 6	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

コントロール インターフェイスおよびデータ インターフェイスの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **redundancy**
4. **application redundancy**
5. **group** *ID*
6. **data** *interface-type interface-number*
7. **control** *interface-type interface-number* **protocol** *id*
8. **timers delay** *seconds* [**reload** *seconds*]
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**redundancy** 例： `Device(config)# redundancy`	冗長コンフィギュレーションモードを開始します。
ステップ 4	**application redundancy** 例： `Device(config-red)# application redundancy`	冗長アプリケーションコンフィギュレーションモードを開始します。
ステップ 5	**group** *ID* 例： `Device(config-red-app)# group 1`	冗長アプリケーション グループ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 6	**data** *interface-type interface-number* 例： `Device(config-red-app-grp)# data GigabitEthernet 0/0/0`	冗長グループに使用されるデータ インターフェイスを指定します。
ステップ 7	**control** *interface-type interface-number* **protocol** *id* 例： `Device(config-red-app-grp)# control gigabitethernet 0/0/2 protocol 1`	冗長グループに使用されるコントロール インターフェイスを指定します。 • このインターフェイスは、コントロール インターフェイス プロトコルのインスタンスにも関連付けられます。
ステップ 8	**timers delay** *seconds* [**reload** *seconds*] 例： `Device(config-red-app-grp)# timers delay 100 reload 400`	障害の発生後、またはシステムのリロード後に起動するロールのネゴシエートを遅らせるために、冗長グループが待機する時間を指定します。
ステップ 9	**end** 例： `Device(config-red-app-grp)# end`	冗長アプリケーション グループ コンフィギュレーション モードを終了して特権EXECモードを開始します。

ファイアウォール ステートフル シャーシ間冗長性の管理とモニタリング

ファイアウォール ステートフル シャーシ間冗長性機能を管理およびモニタするには、次のコマンドを使用します。

手順の概要

1. **enable**
2. **debug redundancy application group config** {**all** | **error** | **event** | **func**}
3. **debug redundancy application group faults** {**all** | **error** | **event** | **fault** | **func**}
4. **debug redundancy application group media** {**all** | **error** | **event** | **nbr** | **packet** {**rx** | **tx**} | **timer**}
5. **debug redundancy application group protocol** {**all** | **detail** | **error** | **event** | **media** | **peer**}
6. **debug redundancy application group rii** {**error** | **event**}
7. **debug redundancy application group transport** {**db** | **error** | **event** | **packet** | **timer** | **trace**}
8. **debug redundancy application group vp** {**error** | **event**}
9. **show redundancy application group** [*group-id* | **all**]
10. **show redundancy application transport** {**client** | **group** [*group-id*]}
11. **show redundancy application control-interface group** [*group-id*]
12. **show redundancy application faults group** [*group-id*]
13. **show redundancy application protocol** {*protocol-id* | **group** [*group-id*]
14. **show redundancy application if-mgr group** [*group-id*]

15. **show redundancy application data-interface group** [*group-id*]
16. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**debug redundancy application group config** {**all** | **error** | **event** | **func**} 例： `Device# debug redundancy application group config all`	冗長グループ アプリケーションの設定を表示します。
ステップ 3	**debug redundancy application group faults** {**all** | **error** | **event** | **fault** | **func**} 例： `Device# debug redundancy application group faults error`	冗長グループ アプリケーションの障害を表示します。
ステップ 4	**debug redundancy application group media** {**all** | **error** | **event** | **nbr** | **packet** {**rx** | **tx**} | **timer**} 例： `Device# debug redundancy application group media timer`	冗長グループ アプリケーションのグループ メディア情報を表示します。
ステップ 5	**debug redundancy application group protocol** {**all** | **detail** | **error** | **event** | **media** | **peer**} 例： `Device# debug redundancy application group protocol peer`	冗長グループ アプリケーションのグループ プロトコル情報を表示します。
ステップ 6	**debug redundancy application group rii** {**error** | **event**} 例： `Device# debug redundancy application group rii event`	冗長グループ アプリケーションのグループ RII 情報を表示します。
ステップ 7	**debug redundancy application group transport** {**db** | **error** | **event** | **packet** | **timer** | **trace**} 例：	冗長グループ アプリケーションのグループ トランスポート情報を表示します。

	コマンドまたはアクション	目的
	`Device# debug redundancy application group transport trace`	
ステップ 8	**debug redundancy application group vp** {**error** | **event**} 例： `Device# debug redundancy application group vp event`	冗長グループ アプリケーションのグループ VP 情報を表示します。
ステップ 9	**show redundancy application group** [*group-id* | **all**] 例： `Device# show redundancy application group all`	冗長グループ情報を表示します。
ステップ 10	**show redundancy application transport** {**client** | **group** [*group-id*]} 例： `Device# show redundancy application transport group 1`	冗長グループのトランスポート固有の情報を表示します。
ステップ 11	**show redundancy application control-interface group** [*group-id*] 例： `Device# show redundancy application control-interface group 2`	冗長グループのコントロール インターフェイス情報を表示します。
ステップ 12	**show redundancy application faults group** [*group-id*] 例： `Device# show redundancy application faults group 2`	冗長グループの障害固有の情報を表示します。
ステップ 13	**show redundancy application protocol** {*protocol-id* | **group** [*group-id*] 例： `Device# show redundancy application protocol 3`	冗長グループのプロトコル固有の情報を表示します。
ステップ 14	**show redundancy application if-mgr group** [*group-id*] 例： `Device# show redundancy application if-mgr group 2`	冗長グループのインターフェイス マネージャ情報を表示します。
ステップ 15	**show redundancy application data-interface group** [*group-id*]	データインターフェイス固有の情報を表示します。

	コマンドまたはアクション	目的
	例： `Device# show redundancy application data-interface group 1`	
ステップ 16	**end** 例： `Device# end`	現在のコンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ファイアウォールステートフルシャーシ間冗長性の設定例

例：冗長アプリケーション グループの設定

次に、優先順位属性とプリエンプション属性のある group1 という名前の冗長グループを設定する例を示します。

```
Device# configure terminal
Device(config)# redundancy
Device(config-red)# application redundancy
Device(config-red-app)# group 1
Device(config-red-app-grp)# name group1
Device(config-red-app-grp)# priority 100 failover-threshold 50
Device(config-red-app-grp)# preempt
Device(config-red-app-grp)# track 200 decrement 200
Device(config-red-app-grp)# end
```

例：冗長グループ プロトコルの設定

次に、hello time メッセージと hold time メッセージ用のタイマーが設定されている冗長グループを設定する例を示します。

```
Device# configure terminal
Device(config)# redundancy
Device(config-red)# application redundancy
Device(config-red-app)# protocol 1
Device(config-red-app-prtcl)# timers hellotime 3 holdtime 9
Device(config-red-app-prtcl)# authentication md5 key-string 0 n1 timeout 100
Device(config-red-app-prtcl)# bfd
Device(config-red-app-prtcl)# end
```

例：仮想 IP アドレスと冗長インターフェイス識別子の設定

次に、ギガビット イーサネット インターフェイス 0/1/1 の冗長グループ仮想 IP アドレスを設定する例を示します。

```
Device# configure terminal
Device(config)# interface GigabitEthernet 0/1/1
Device(conf-if)# redundancy rii 600
Device(config-if)# redundancy group 2 ip 10.2.3.4 exclusive decrement 200
Device(config-if)# end
```

例：コントロール インターフェイスとデータ インターフェイスの設定

```
Device# configure terminal
Device(config-red)# application redundancy
Device(config-red-app-grp)# group 1
Device(config-red-app-grp)# data GigabitEthernet 0/0/0
Device(config-red-app-grp)# control GigabitEthernet 0/0/2 protocol 1
Device(config-red-app-grp)# timers delay 100 reload 400
Device(config-red-app-grp)# end
```

例：LAN-LAN トポロジの設定

次のサンプルLAN-LAN構成で、ステートフルな冗長性を確保するために、2つの発信インターフェイスを備えたルータのペアを設定する方法を示します。この例では、GigabitEthernet 0/1/1が入力インターフェイスで、GigabitEthernet 0/2/1 が出力インターフェイスです。両方のインターフェイスがゾーンに割り当てられ、ゾーン間のトラフィックを記述するためのクラスマップが定義されます。また、冗長性を確保するようにインターフェイスが設定されます。「検査」アクションがアプリケーション レベル ゲートウェイ（ALG）を呼び出して、ピンホールを開き、他のポート上のトラフィックを許可します。ピンホールは、保護されたネットワークへの制御されたアクセスを特定のアプリケーションが取得できるようにするためにALG経由で開かれるポートです。

アクティブ デバイスである Device 1 の設定を以下に示します。

```
! Configures redundancy, control and data interfaces
redundancy
 mode none
 application redundancy
  group 2
   preempt
   priority 200 failover threshold 100
   control GigabitEthernet 0/0/4 protocol 2
   data GigabitEthernet 0/0/3
!
  protocol 2
   timers hellotime ms 250 holdtime ms 750
!
! Configures a VRF
ip vrf vrf1
!
! Configures parameter maps to add parameters that control the behavior of actions and
match criteria.
parameter-map type inspect pmap-udp
 redundancy
  redundancy delay 10
!
parameter-map type inspect pmap-tcp
```

```
 redundancy
  redundancy delay 10
!
! Defines class-maps to describes traffic between zones
class-map type inspect match-any cmap-udp
 match protocol udp
!
class-map type inspect match-any cmap-ftp-tcp
 match protocol ftp
 match protocol tcp
!
! Associates class-maps with policy-maps to define actions to be applied
policy-map type inspect p1
 class type inspect cmap-udp
  inspect pmap-udp
!
 class type inspect cmap-ftp-tcp
  inspect pmap-tcp
!
! Identifies and defines network zones
zone security z-int
!
zone security z-hi
!
! Sets zone pairs for any policy other than deny all and assign policy-maps to zone-pairs
 by defining a service-policy
zone-pair security hi2int source z-hi destination z-int
 service-policy type inspect p1
!
! Assigns interfaces to zones
interface GigabitEthernet 0/0/1
 ip vrf forwarding vrf1
 ip address 10.1.1.3 255.255.0.0
 ip virtual-reassembly
 zone-member security z-hi
 negotiation auto
 redundancy rii 20
 redundancy group 2 ip 10.1.1.10 exclusive decrement 50
!
interface GigabitEthernet 0/0/2
 ip vrf forwarding vrf1
 ip address 192.0.2.2 255.255.255.240
 ip virtual-reassembly
 zone-member security z-int
 negotiation auto
 redundancy rii 21
 redundancy group 2 ip 192.0.2.12 exclusive decrement 50
!
interface GigabitEthernet 0/0/4
 ip address 198.51.100.17 255.255.255.240
!
interface GigabitEthernet 0/0/4
 ip address 203.0.113.49 255.255.255.240
!
ip route vrf vrf1 192.0.2.0 255.255.255.240 GigabitEthernet0/0/2 10.1.1.4
ip route vrf vrf1 10.1.0.0 255.255.0.0 GigabitEthernet0/0/1 10.1.0.4
!
```

スタンバイ デバイスである Device 2 の設定を以下に示します。

```
! Configures redundancy, control and data interfaces
redundancy
 mode none
 application redundancy
  group 2
```

```
   preempt
   priority 200 failover threshold 100
   control GigabitEthernet 0/0/4 protocol 2
   data GigabitEthernet 0/0/3
!
  protocol 2
   timers hellotime ms 250 holdtime ms 750
!
! Configures a VRF
ip vrf vrf1
!
! Configures parameter maps to add parameters that control the behavior of actions and
match criteria.
parameter-map type inspect pmap-udp
 redundancy
  redundancy delay 10
!
parameter-map type inspect pmap-tcp
 redundancy
  redundancy delay 10
!
! Defines class-maps to describes traffic between zones
class-map type inspect match-any cmap-udp
 match protocol udp
!
class-map type inspect match-any cmap-ftp-tcp
 match protocol ftp
 match protocol tcp
!
! Associates class-maps with policy-maps to define actions to be applied
policy-map type inspect p1
 class type inspect cmap-udp
  inspect pmap-udp
!
 class type inspect cmap-ftp-tcp
  inspect pmap-tcp
!
! Identifies and defines network zones
zone security z-int
!
zone security z-hi
!
! Sets zone pairs for any policy other than deny all and assign policy-maps to zone-pairs
 by defining a service-policy
zone-pair security hi2int source z-hi destination z-int
 service-policy type inspect p1
!
! Assigns interfaces to zones
interface GigabitEthernet 0/0/1
 ip vrf forwarding vrf1
 ip address 10.1.1.6 255.255.0.0
 ip virtual-reassembly
 zone-member security z-hi
 negotiation auto
 redundancy rii 20
 redundancy group 2 ip 10.1.1.12 exclusive decrement 50
!
interface GigabitEthernet 0/0/2
 ip vrf forwarding vrf1
 ip address 192.0.2.5 255.255.255.240
 ip virtual-reassembly
 zone-member security z-int
 negotiation auto
 redundancy rii 21
```

```
 redundancy group 2 ip 192.0.2.10 exclusive decrement 50
!
interface GigabitEthernet 0/0/4
 ip address 198.51.100.21 255.255.255.240
!
interface GigabitEthernet 0/0/4
 ip address 203.0.113.53 255.255.255.240
!
ip route vrf vrf1 192.0.2.0 255.255.255.240 GigabitEthernet0/0/2 10.1.1.4
ip route vrf vrf1 10.1.0.0 255.255.0.0 GigabitEthernet0/0/1 10.1.0.4
!
```

ファイアウォールステートフルシャーシ間冗長性に関する追加情報

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Master Command List, All Releases』
セキュリティ コマンド	• 『Security Command Reference: Commands A to C』 • 『Security Command Reference: Commands D to L』 • 『Security Command Reference: Commands M to R』 • 『Security Command Reference: Commands S to Z』

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

ファイアウォールステートフルシャーシ間冗長性に関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1:* ファイアウォール ステートフル シャーシ間冗長性に関する機能情報

機能名	リリース	機能情報
ファイアウォール ステートフル シャーシ間冗長性	Cisco IOS XE リリース 3.1(S)	ファイアウォール ステートフル シャーシ間冗長性機能を使用すれば、デバイスのペアを互いのバックアップとして機能するように設定することができます。 次のコマンドが導入または変更されました。**application redundancy**、**authentication**、**control**、**data**、**debug redundancy application group config**、**debug redundancy application group faults**、**debug redundancy application group media**、**debug redundancy application group protocol**、**debug redundancy application group rii**、**debug redundancy application group transport**、**debug redundancy application group vp**、**group**、**name**、**preempt**、**priority**、**protocol**、**redundancy rii**、**redundancy group**、**track**、**timers delay**、**timers hellotime**、**show redundancy application group**、**show redundancy application transport**、**show redundancy application control-interface**、**show redundancy application faults**、**show redundancy application protocol**、**show redundancy application if-mgr**、**show redundancy application data-interface**。

機能名	リリース	機能情報
ゾーンベース ファイアウォールでの VRF 対応ステートフル シャーシ間冗長性	Cisco IOS XE リリース 3.14S	Cisco IOS XE リリース 3.14S では、ゾーンベース ファイアウォールが VRF 対応シャーシ間冗長性をサポートします。アクティブ デバイスとスタンバイ デバイスの VPN ルーティングおよび転送（VRF）名は同じにする必要があります。アクティブ デバイスとスタンバイ デバイスの両方で同じ VRF 設定を使用できる必要があります。

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。