



Secure Firewall 3100 の ASA クラスタ

クラスタリングを利用すると、複数の ASA をグループ化して 1 つの論理デバイスとすることができます。クラスタは、単一デバイスのすべての利便性（管理、ネットワークへの統合）を備える一方で、複数デバイスによって高いスループットおよび冗長性を達成します。



(注) クラスタリングを使用する場合、一部の機能はサポートされません。[クラスタリングでサポートされない機能 \(92 ページ\)](#) を参照してください。

- [ASA クラスタリングの概要 \(1 ページ\)](#)
- [ASA クラスタリングのライセンス \(5 ページ\)](#)
- [ASA クラスタリングの要件と前提条件 \(7 ページ\)](#)
- [ASA クラスタリングのガイドライン \(9 ページ\)](#)
- [ASA クラスタリングの設定 \(15 ページ\)](#)
- [クラスタノードの管理 \(54 ページ\)](#)
- [ASA クラスタのモニタリング \(60 ページ\)](#)
- [ASA クラスタリングの例 \(72 ページ\)](#)
- [クラスタリングの参考資料 \(92 ページ\)](#)
- [Secure Firewall 3100 の ASA クラスタリングの履歴 \(110 ページ\)](#)

ASA クラスタリングの概要

ここでは、クラスタリングアーキテクチャとその動作について説明します。

クラスタをネットワークに適合させる方法

クラスタは、複数のファイアウォールで構成され、これらは 1 つのユニットとして機能します。ファイアウォールをクラスタとして機能させるには、次のインフラストラクチャが必要です。

- クラスタ内通信用の、隔離された高速バックプレーンネットワーク。クラスタ制御リンクと呼ばれます。

- 各ファイアウォールへの管理アクセス（コンフィギュレーションおよびモニタリングのため）。

クラスタをネットワーク内に配置するときは、クラスタが送受信するデータのロードバランシングを、アップストリームおよびダウンストリームのルータがスパンドEtherChannelを使用していることが必要です。クラスタ内の複数のメンバーのインターフェイスをグループ化して1つのEtherChannelとします。このEtherChannelがユニット間のロードバランシングを実行します。

クラスタ メンバー

クラスタメンバーは連携して動作し、セキュリティポリシーおよびトラフィックフローの共有を達成します。ここでは、各メンバーのロールの特長について説明します。

ブートストラップコンフィギュレーション

各デバイスで、最小限のブートストラップコンフィギュレーション（クラスタ名、クラスタ制御リンクインターフェイスなどのクラスタ設定）を設定します。通常、クラスタリングを有効にする最初のノードが制御ノードになります。以降のノードに対してクラスタリングをイネーブルにすると、そのノードはデータノードとしてクラスタに参加します。

制御ノードとデータノードの役割

クラスタ内のメンバーの1つが制御ノードになります。複数のクラスタノードが同時にオンラインになる場合、制御ノードは、ブートストラップコンフィギュレーション内のプライオリティ設定によって決まります。プライオリティは1～100の範囲内で設定され、1が最高のプライオリティです。他のすべてのメンバーはデータノードです。一般的には、クラスタを作成した後で最初に追加したノードが制御ノードとなります。これは単に、その時点でクラスタに存在する唯一のノードであるからです。

すべてのコンフィギュレーション作業（ブートストラップコンフィギュレーションを除く）は、制御ノード上のみで実行する必要があります。コンフィギュレーションは、データノードに複製されます。物理的アセット（たとえばインターフェイス）の場合は、制御ノードのコンフィギュレーションがすべてのデータノード上でミラーリングされます。たとえば、内部インターフェイスとしてイーサネット1/2を設定し、外部インターフェイスとしてイーサネット1/1を設定した場合、これらのインターフェイスは内部および外部インターフェイスとしてデータノードでも使用されます。

機能によっては、クラスタ内でスケールしないものがあり、そのような機能については制御ノードがすべてのトラフィックを処理します。

クラスタ インターフェイス

データインターフェイスは、スパンドEtherChannel。詳細については、[クラスタインターフェイスについて（15 ページ）](#)を参照してください。

クラスタ制御リンク

各ユニットの、少なくとも1つのハードウェアインターフェイスをクラスタ制御リンク専用とする必要があります。詳細については、[クラスタ制御リンク \(15 ページ\)](#) を参照してください。

コンフィギュレーションの複製

クラスタ内のすべてのノードは、単一の設定を共有します。設定の変更は制御ノードでのみ可能（ブートストラップ設定は除く）で、変更はクラスタに含まれる他のすべてのノードに自動的に同期されます。

ASA クラスタ管理

ASA クラスタリングを使用することの利点の1つは、管理のしやすさです。ここでは、クラスタを管理する方法について説明します。

管理ネットワーク

すべてのユニットを単一の管理ネットワークに接続することを推奨します。このネットワークは、クラスタ制御リンクとは別のものです。

管理インターフェイス

管理インターフェイスについては、専用管理インターフェイスの1つを使用することを推奨します。管理インターフェイスは、個別インターフェイスとして設定することも（ルーテッドモードとトランスペアレントモードの両方）、スパンド EtherChannel インターフェイスとして設定することもできます。

管理用には、個別インターフェイスを使用することを推奨します。個別インターフェイスならば、必要に応じて各ユニットに直接接続できますが、スパンド EtherChannel インターフェイスでは、現在の制御ユニットへのリモート接続しかできません。



(注) 管理インターフェイスに対してダイナミックルーティングをイネーブルにすることはできません。スタティックルートを使用する必要があります。

個別インターフェイスの場合は、メインクラスタ IP アドレスはそのクラスタの固定アドレスであり、常に現在の制御ユニットに属します。インターフェイスごとに、管理者はアドレス範囲も設定します。これで、各ユニット（現在の制御ユニットも含まれます）がその範囲内のローカルアドレスを使用できるようになります。このメインクラスタ IP アドレスによって、管理アクセスのアドレスが一本化されます。制御ユニットが変更されると、メインクラスタ IP アドレスは新しい制御ユニットに移動するので、クラスタの管理をシームレスに続行できます。ローカル IP アドレスは、ルーティングに使用され、トラブルシューティングにも役立ちます。

たとえば、クラスタを管理するにはメインクラスタ IP アドレスに接続します。このアドレスは常に、現在の制御ユニットに関連付けられています。個々のメンバを管理するには、ローカル IP アドレスに接続します。

TFTP や syslog などの発信管理トラフィックの場合、制御ユニットを含む各ユニットは、ローカル IP アドレスを使用してサーバーに接続します。

スバンド EtherChannel インターフェイスの場合は、IP アドレスは 1 つだけ設定でき、その IP アドレスは常に制御ユニットに関連付けられます。EtherChannel インターフェイスを使用してデータユニットに直接接続することはできません。管理インターフェイスは個別インターフェイスとして設定することを推奨します。各ユニットに接続できるようにするためです。デバイス ローカル EtherChannel を管理に使用できます。

制御ユニット管理とデータユニット管理

すべての管理とモニタリングは制御ノードで実行できます。制御ノードから、すべてのノードのランタイム統計情報、リソース使用状況、その他のモニタリング情報を確認できます。また、クラスタ内のすべてのノードに対してコマンドを発行したり、コンソールメッセージをデータノードから制御ノードに複製したりできます。

必要に応じて、データノードを直接モニタできます。制御ノードからも可能ですが、ファイル管理（設定のバックアップやイメージの更新など）をデータノード上で実行できます。次の機能は、制御ノードからは使用できません。

- ノードごとのクラスタ固有統計情報のモニタリング。
- ノードごとの Syslog モニタリング（コンソールレプリケーションが有効な場合にコンソールに送信される Syslog を除く）。
- SNMP
- NetFlow

暗号キー複製

制御ノード上で暗号キーを作成すると、そのキーはすべてのデータノードに複製されます。メインクラスタ IP アドレスへの SSH セッションがある場合、制御ノードで障害が発生すると接続が切断されます。新しい制御ノードでは、SSH 接続に対して同じキーが使用されるため、新しい制御ノードに再接続するときに、キャッシュ済みの SSH ホストキーを更新する必要はありません。

ASDM 接続証明書 IP アドレス不一致

デフォルトでは、自己署名証明書は、ローカル IP アドレスに基づいて ASDM 接続に使用されます。ASDM を使用してメインクラスタ IP アドレスに接続すると、IP アドレス不一致に関する警告メッセージが表示される場合があります。これは、証明書で使用されているのがローカル IP アドレスであり、メインクラスタ IP アドレスではないためです。このメッセージは無視して、ASDM 接続を確立できます。ただし、この種の警告を回避するには、新しい証明書を登録し、この中でメインクラスタ IP アドレスと、IP アドレスプールからのすべてのローカル IP アドレスを指定します。この証明書を各クラスタ メンバに使用します。詳細については、

「<https://www.cisco.com/c/en/us/td/docs/security/asdm/identity-cert/cert-install.html>」を参照してください。

サイト間クラスタリング

サイト間インストールの場合、推奨されるガイドラインに従ってれば、ASAクラスタリングを活用できます。

各クラスタ シャーシを、個別のサイト ID に属するように設定できます。

サイト ID は、サイト固有の MAC アドレスおよび IP アドレスと連動します。クラスタから送信されたパケットは、サイト固有の MAC アドレスおよび IP アドレスを使用するのに対し、クラスタで受信したパケットは、グローバル MAC アドレスおよび IP アドレスを使用します。この機能により、MAC フラッピングの原因となる 2 つの異なるポートで両方のサイトから同じグローバル MAC アドレスをスイッチが学習するのを防止します。代わりに、スイッチはサイトの MAC アドレスのみを学習します。サイト固有の MAC アドレスおよび IP アドレスは、スパンド EtherChannel のみを使用したルーテッドモードでサポートされます。

サイト ID は、LISP インスペクションを使用するフローモビリティ、データセンターのサイト間クラスタリングのパフォーマンスを向上し、ラウンドトリップ時間の遅延を減少させるためのディレクタ ローカリゼーション、およびトラフィック フローのバックアップ オーナーが常にオーナーとは異なるサイトにある接続のサイト冗長性を有効にするためにも使用されます。

サイト間クラスタリングの詳細については、以下の項を参照してください。

- Data Center Interconnect のサイジング : [ASA クラスタリングの要件と前提条件 \(7 ページ\)](#)
- サイト間のガイドライン : [ASA クラスタリングのガイドライン \(9 ページ\)](#)
- クラスタ フロー モビリティの設定 : [クラスタ フロー モビリティの設定 \(48 ページ\)](#)
- ディレクタ ローカリゼーションの有効化 : [ディレクタ ローカリゼーションの有効化 \(46 ページ\)](#)
- サイト冗長性の有効化 : [ディレクタ ローカリゼーションの有効化 \(46 ページ\)](#)
- サイト間での例 : [サイト間クラスタリングの例 \(88 ページ\)](#)

ASA クラスタリングのライセンス

Smart Software Manager Regular およびオンプレミス

各ユニットには、標準ライセンス (デフォルトで有効) と同じ暗号化ライセンスが必要です。ライセンス不一致の問題を回避するために、クラスタリングを有効にする前に、ライセンスサーバで各ユニットのライセンスを取得することをお勧めします。また、高度暗号化ライセンスを使用する場合は、クラスタ制御リンクの暗号化に関する問題も発生します。

クラスタリング機能自体にライセンスは必要ありません。データユニットのコンテキストライセンスに追加料金はかかりません。

高度暗号化ライセンスは、登録トークンを適用すると、対象となるお客様の場合自動的に有効化されます。ASA 設定で有効化される高度暗号化 (3DES/AES) 機能ライセンスについては、以下を参照してください。

ASA ライセンス設定では、標準ライセンスはすべてのユニットで常にデフォルトで有効になっています。制御ユニットにのみスマートライセンスを設定できます。設定はデータユニットに複製されますが、一部のライセンスに対しては、データユニットはこの設定を使用しません。この設定はキャッシュ状態のままになり、制御ユニットのみがこのライセンスを要求します。ライセンスは単一のクラスタライセンスにまとめられ、クラスタの各ユニットで共有されます。この集約ライセンスはデータユニットにもキャッシュされ、その中の 1 つが将来制御ユニットとなったときに使用されます。各ライセンスタイプは次のように処理されます：

- 標準：各ユニットには、サーバーからの標準のライセンスが必要です。
- コンテキスト：制御ユニットのみがサーバーからコンテキストライセンスを要求します。デフォルトで標準ライセンスは 2 のコンテキストを含み、すべてのクラスタメンバー上に存在します。各ユニットの標準ライセンスの値と、制御ユニットのコンテキストライセンスの値は、集約されたクラスタライセンスでのプラットフォーム制限まで統合されます。次に例を示します。
 - クラスタ内に 6 つの Secure Firewall 3100 があります。標準ライセンスは 2 のコンテキストを含みます。6 ユニットの場、合計で 12 のコンテキストが加算されます。制御ユニット上で追加の 20 コンテキストライセンスを設定します。したがって、集約されたクラスタライセンスは 32 のコンテキストを含みます。シャーシごとのプラットフォームの制限が 100 であるため、結合されたライセンスでは最大 100 のコンテキストが許容されます。32 コンテキストは制限の範囲内です。したがって、制御ユニット上で最大 32 コンテキストを設定できます。各データユニットも、コンフィギュレーションの複製を介して 32 コンテキストを持つことになります。
 - クラスタ内に 3 つの Secure Firewall 3100 ユニットがあります。標準ライセンスは 2 のコンテキストを含みます。3 ユニットの場、合計で 6 のコンテキストが加算されます。制御ユニット上で追加の 100 コンテキストライセンスを設定します。したがって、集約されたクラスタライセンスは 106 のコンテキストを含みます。ユニットごとのプラットフォームの制限が 100 であるため、統合されたライセンスでは最大 100 のコンテキストが許容されます。106 コンテキストは制限を超えています。したがって、制御ユニット上で最大 100 のコンテキストのみを設定できます。各データユニットも、設定の複製を介して 100 のコンテキストを持つことになります。この場合では、制御ユニットのコンテキストライセンスとして 94 のコンテキストのみを設定する必要があります。
- 高度暗号化 (3DES) (追跡目的用) —制御ユニットのみがこのライセンスを要求し、ライセンスの集約によりすべてのユニットがこれを使用できます。

新しい制御ユニットが選定されると、このユニットが集約ライセンスを引き続き使用します。また、制御ユニットのライセンスを再要求するために、キャッシュされたライセンス設定も使

用します。古い制御ユニットがデータユニットとしてクラスタに再度参加すると、制御ユニットのライセンス権限付与が解放されます。アカウントに利用可能なライセンスがない場合、データユニットがライセンスを解放する前に、制御ユニットのライセンスがコンプライアンス違反状態になることがあります。保持されたライセンスは 30 日間有効ですが、この猶予期間以降もコンプライアンス違反となる場合、特別なライセンスを必要とする機能の設定変更を行なえません。ただし、動作には影響ありません。新しいアクティブ装置は、ライセンスのコンプライアンスが確保されるまで 35 秒ごとに権限承認更新要求を送信します。ライセンス要求が完全に処理されるまで、設定の変更を控えてください。ユニットがクラスタから離れた場合、キャッシュされた制御ユニットの設定は削除されます。一方で、ユニットごとの権限は保持されます。この場合、クラスタ外のユニットのコンテキストライセンスを再要求する必要があります。

永続ライセンスの予約

永続ライセンスを予約するには、シャーンごとに個別のライセンスを購入し、クラスタリングを設定する前にライセンスを有効にする必要があります。

ASA クラスタリングの要件と前提条件

モデルの要件

- Secure Firewall 3100 : 最大 8 ユニット

ASA のハードウェアおよびソフトウェア要件

クラスタ内のすべてのユニット :

- 同じ DRAM を使用する同じモデルである必要があります。フラッシュメモリの容量は同一である必要はありません。
- イメージアップグレード時を除き、同じソフトウェアを実行する必要があります。ヒットレスアップグレードがサポートされます。
- セキュリティ コンテキスト モードが一致している必要があります (シングルまたはマルチ)。
- (シングル コンテキスト モード) ファイアウォール モードが一致している必要があります (ルーテッドまたはトランスペアレント)。
- コンフィギュレーション複製前の初期クラスタ制御リンク通信のために、新しいクラスタメンバーは、制御ユニットと同じ SSL 暗号化設定 (`ssl encryption` コマンド) を使用する必要があります。

スイッチ要件

- ASA でクラスタリングを設定する前に、スイッチのコンフィギュレーションを完了する必要があります。

- サポートされているスイッチのリストについては、『[Cisco ASA Compatibility](#)』 [英語] を参照してください。

ASA の要件

- ユニットの管理ネットワークに追加する前に、一意の IP アドレスを各ユニットに提供します。
 - ASA への接続および管理 IP アドレスの設定に関する詳細については、「使用する前に」の章を参照してください。
 - 制御ユニット（通常は最初にクラスタに追加されたユニット）で使用される IP アドレスを除き、これらの管理 IP アドレスは一時的に使用されるだけです。
 - データユニットがクラスタに参加すると、管理インターフェイス設定はマスターユニットからの複製に置き換えられます。

サイト間クラスタリング用の Data Center Interconnect のサイジング

次の計算と同等の帯域幅をクラスタ制御リンク トラフィック用に Data Center Interconnect (DCI) に確保する必要があります。

$$\frac{\text{\# of cluster members per site}}{2} \times \text{cluster control link size per member}$$

メンバーの数が各サイトで異なる場合、計算には大きい方の値を使用します。DCIの最小帯域幅は、1つのメンバーに対するクラスタ制御リンクのサイズ未満にすることはできません。

次に例を示します。

- 4 サイトの 2 メンバーの場合。
 - 合計 4 クラスタ メンバー
 - 各サイト 2 メンバー
 - メンバーあたり 5 Gbps クラスタ制御リンク

予約する DCI 帯域幅 = 5 Gbps (2/2 x 5 Gbps)。

- 3 サイトの 6 メンバーの場合、サイズは増加します。
 - 合計 6 クラスタ メンバー
 - サイト 1 は 3 メンバー、サイト 2 は 2 メンバー、サイト 3 は 1 メンバー
 - メンバーあたり 10 Gbps クラスタ制御リンク

予約する DCI 帯域幅 = 15 Gbps (3/2 x 10 Gbps)。

- 2 サイトの 2 メンバーの場合。
 - 合計 2 クラスタ メンバー

- 各サイト 1 メンバー
- メンバーあたり 10 Gbps クラスタ制御リンク

予約する DCI 帯域幅 = 10 Gbps (1/2 x 10 Gbps = 5 Gbps、ただし最小帯域幅がクラスタ制御リンク (10 Gbps) のサイズ未満になってはなりません)。

その他の要件

ターミナルサーバーを使用して、すべてのクラスタメンバーユニットのコンソールポートにアクセスすることをお勧めします。初期設定および継続的な管理 (ユニットがダウンしたときなど) では、ターミナルサーバーがリモート管理に役立ちます。

ASA クラスタリングのガイドライン

コンテキスト モード

モードは、各メンバーユニット上で一致している必要があります。

ファイアウォール モード

シングルモードの場合、ファイアウォールモードがすべてのユニットで一致している必要があります。

フェールオーバー

フェールオーバーは、クラスタリングではサポートされません。

IPv6

クラスタ制御リンクは、IPv4 のみを使用してサポートされます。

スイッチ

- 接続されているスイッチが、クラスタ データ インターフェイスとクラスタ制御リンク インターフェイスの両方の MTU と一致していることを確認します。クラスタ制御リンク インターフェイスの MTU は、データ インターフェイスの MTU より 100 バイト以上大きく設定する必要があります。そのため、スイッチを接続するクラスタ制御リンクを適切に設定してください。クラスタ制御リンクのトラフィックにはデータパケット転送が含まれるため、クラスタ制御リンクはデータパケット全体のサイズに加えてクラスタトラフィックのオーバーヘッドにも対応する必要があります。
- Cisco IOS XR システムでデフォルト以外の MTU を設定する場合は、クラスタデバイスの MTU よりも 14 バイト大きい IOS XR インターフェイスの MTU を設定します。そうしないと、**mtu-ignore** オプションを使用しない限り、OSPF 隣接関係ピアリングの試行が失敗する可能性があります。クラスタデバイス MTU は、IOS XR IPv4 MTU と一致させる必要があります。この調整は、Cisco Catalyst および Cisco Nexus スイッチでは必要ありません。

- クラスタ制御リンク インターフェイスのスイッチでは、クラスタ ユニットに接続されるスイッチポートに対してスパンニングツリー PortFast をイネーブルにすることもできます。このようにすると、新規ユニットの参加プロセスを高速化できます。
- スイッチでは、EtherChannel ロードバランシング アルゴリズム **source-dest-ip** または **source-dest-ip-port** (Cisco Nexus OS および Cisco IOS-XE の **port-channel load-balance** コマンドを参照) を使用することをお勧めします。クラスタのデバイスにトラフィックを不均一に配分する場合があるので、ロードバランシング アルゴリズムでは **vlan** キーワードを使用しないでください。クラスタデバイスのデフォルトのロードバランシング アルゴリズムは変更しないでください。
- スイッチの EtherChannel ロードバランシング アルゴリズムを変更すると、スイッチの EtherChannel インターフェイスは一時的にトラフィックの転送を停止し、スパンニングツリー プロトコルが再始動します。トラフィックが再び流れ出すまでに、少し時間がかかります。
- 一部のスイッチは、LACP でのダイナミック ポート プライオリティをサポートしていません (アクティブおよびスタンバイ リンク)。ダイナミック ポート プライオリティを無効化することで、スパンド EtherChannel との互換性を高めることができます。
- クラスタ制御リンク パスのスイッチでは、L4 チェックサムを検証しないようにする必要があります。クラスタ制御リンク経路でダイレクトされたトラフィックには、正しい L4 チェックサムが設定されていません。L4 チェックサムを検証するスイッチにより、トラフィックがドロップされる可能性があります。
- ポートチャネルバンドルのダウンタイムは、設定されているキープアライブ インターバルを超えてはなりません。
- Supervisor 2T EtherChannel では、デフォルトのハッシュ配信アルゴリズムは適応型です。VSS 設計での非対称トラフィックを避けるには、クラスタデバイスに接続されているポートチャネルでのハッシュ アルゴリズムを固定に変更します。

```
router(config)# port-channel id hash-distribution fixed
```

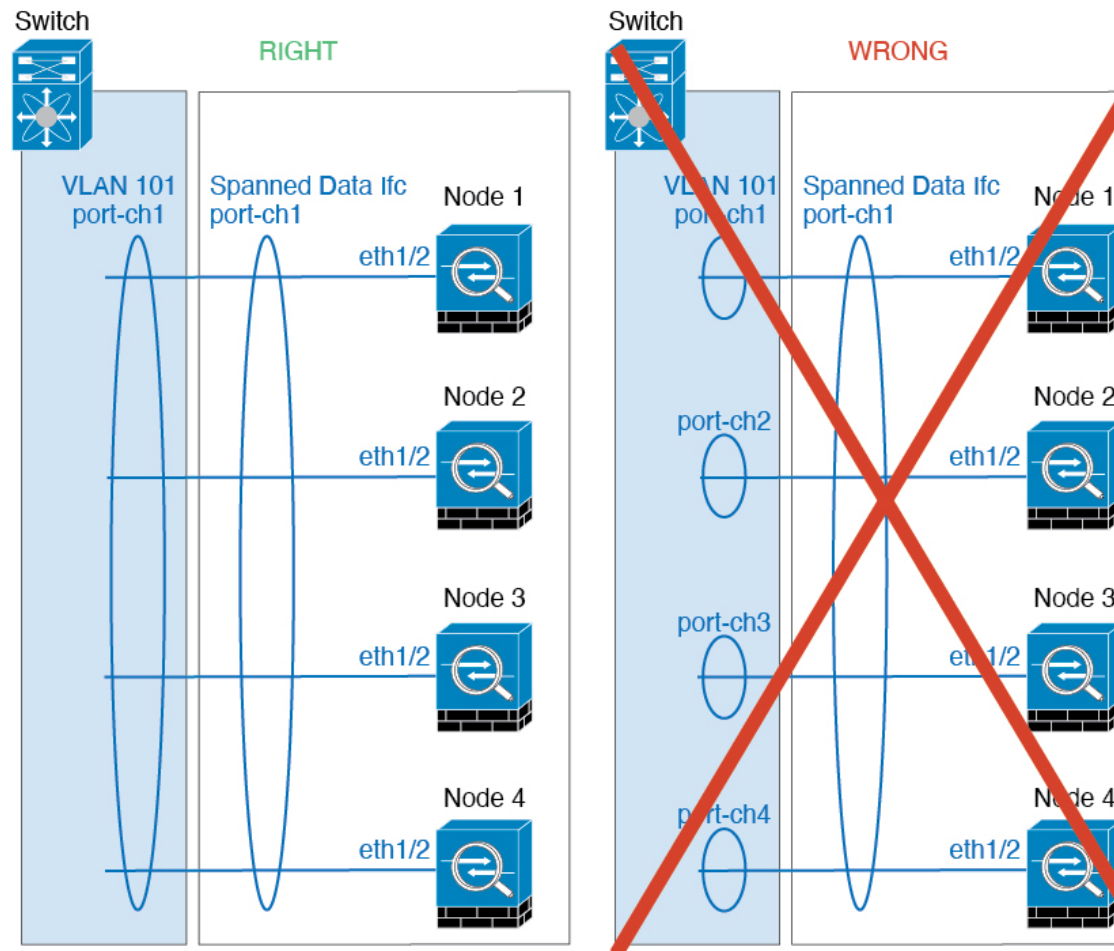
 アルゴリズムをグローバルに変更しないでください。VSS ピア リンクに対しては適応型アルゴリズムを使用できます。
- Cisco Nexus スイッチのクラスタに接続されたすべての EtherChannel インターフェイスで、LACP グレースフル コンバージェンス機能をディセーブルにする必要があります。

EtherChannel

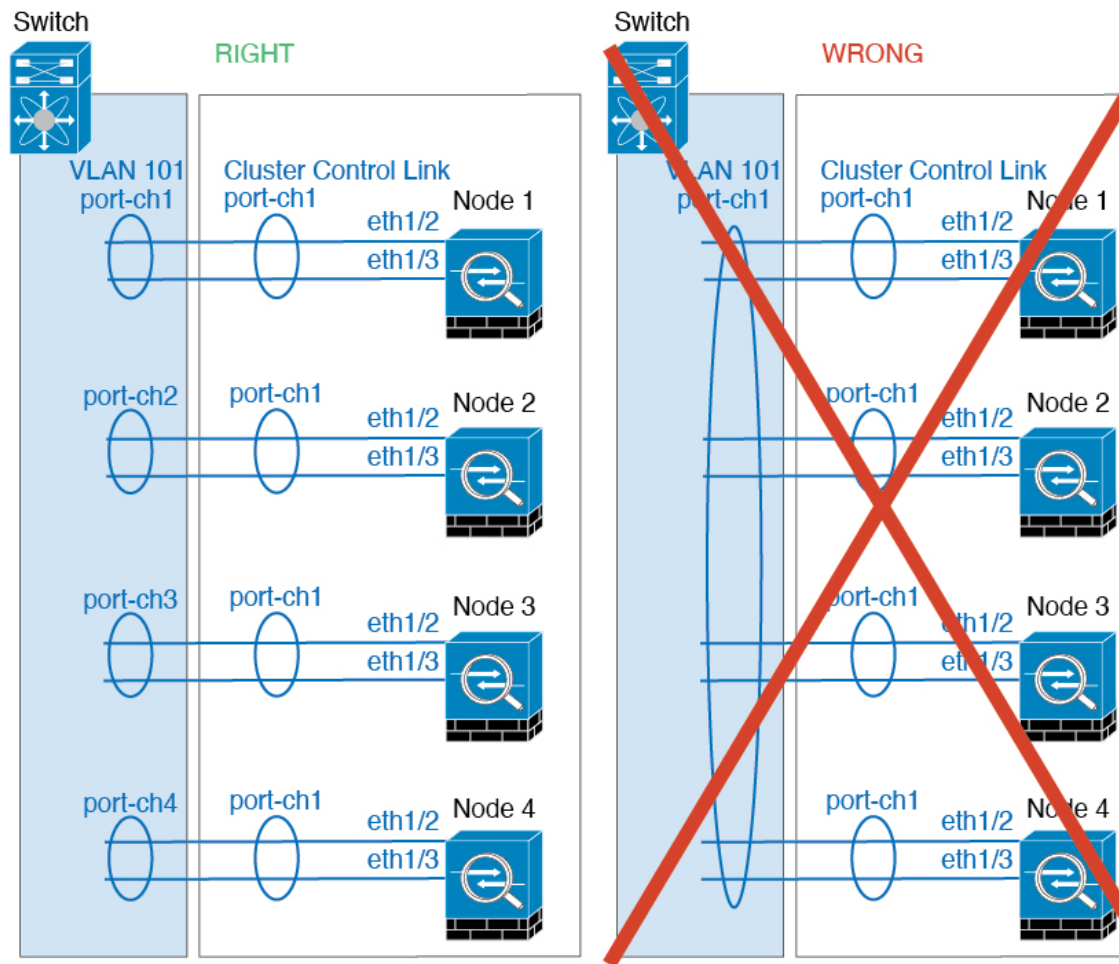
- 15.1(1)S2 より前の Catalyst 3750-X Cisco IOS ソフトウェア バージョンでは、クラスタ ユニットはスイッチ スタックに EtherChannel を接続することをサポートしていませんでした。デフォルトのスイッチ設定では、クラスタユニット EtherChannel がクロススタックに接続されている場合、制御ユニットのスイッチの電源がオフになると、残りのスイッチに接続されている EtherChannel は起動しません。互換性を高めるため、**stack-mac persistent timer** コマンドを設定して、十分なリロード時間を確保できる大きな値、たとえば 8 分、

0（無制限）などを設定します。または、15.1(1)S2 など、より安定したスイッチ ソフトウェアバージョンにアップグレードできます。

- スパンド EtherChannel とデバイスローカル EtherChannel のコンフィギュレーション：スパンド EtherChannel と デバイス ローカル EtherChannel に対してスイッチを適切に設定します。
- スパンド EtherChannel：クラスタ ユニット スパンド EtherChannel（クラスタのすべてのメンバに広がる）の場合は、複数のインターフェイスが結合されてスイッチ上の単一の EtherChannel となります。各インターフェイスがスイッチ上の同じチャネルグループ内にあることを確認してください。



- デバイス ローカル EtherChannel：クラスタ ユニット デバイス ローカル EtherChannel（クラスタ制御リンク用に設定された EtherChannel もこれに含まれます）は、それぞれ独立した EtherChannel としてスイッチ上で設定してください。スイッチ上で複数のクラスタ ユニット EtherChannel を結合して 1 つの EtherChannel としないでください。



サイト間のガイドライン

サイト間クラスタリングについては、次のガイドラインを参照してください。

- クラスタ制御リンクの遅延が、ラウンドトリップ時間 (RTT) 20 ms 未満である必要があります。
- クラスタ制御リンクは、順序の異常やパケットのドロップがない信頼性の高いものである必要があります。たとえば、専用リンクを使用する必要があります。
- 接続の再分散を設定しないでください。異なるサイトのクラスタメンバには接続を再分散できません。
- ASA は専用リンクであるため、データセンター相互接続 (DCI) で使用されている場合でも、クラスタ制御リンクで転送されるデータトラフィックを暗号化しません。オーバーレイトランスポート仮想化 (OTV) を使用する場合、またはローカル管理ドメインの外部でクラスタ制御リンクを拡張する場合は、OTE を介した 802.1AE MacSec などの境界ルータで暗号化を設定できます。

- クラスタの実装では、着信接続用の複数のサイトでメンバが区別されません。したがって、特定の接続に対する接続のルールが複数のサイトにまたがる場合があります。これは想定されている動作です。ただし、ディレクタローカリゼーションを有効にすると、ローカルディレクタのルールは（サイト ID に従って）常に接続オーナーと同じサイトから選択されます。また、元のオーナーに障害が発生すると、ローカルディレクタが同じサイトで新しいオーナーを選択します（注：サイト間でトラフィックが非対称で、元のオーナーに障害が発生した後もリモートサイトから継続的にトラフィックが発生する場合、リモートサイトのノードが再ホスティングウィンドウ内でデータパケットを受信する場合にはこのリモートサイトのノードが新しいオーナーとなることがあります）。
- ディレクタローカリゼーションでは、次のトラフィックタイプのローカリゼーションをサポートしていません。NAT または PAT のトラフィック、SCTP がインスペクションを行うトラフィック、オーナーのフラグメンテーションクエリ。
- トランスペアレントモードの場合、内部ルータと外部ルータのペア間にクラスタを配置すると（AKA ノースサウス挿入）、両方の内部ルータが同じ MAC アドレスを共有し、両方の外部ルータが同じ MAC アドレスを共有する必要があります。サイト 1 のクラスタメンバがサイト 2 のメンバに接続を転送するとき、宛先 MAC アドレスは維持されます。MAC アドレスがサイト 1 のルータと同じである場合にのみ、パケットはサイト 2 のルータに到達します。
- トランスペアレントモードの場合、内部ネットワーク間のファイアウォール用に各サイトのデータネットワークとゲートウェイルータ間にクラスタを配置すると（AKA イーストウェスト挿入）、各ゲートウェイルータは、HSRP などの First Hop Redundancy Protocol (FHRP) を使用して、各サイトで同じ仮想 IP および MAC アドレスの宛先を提供します。データ VLAN は、オーバーレイ トランスポート仮想化 (OTV) または同様のものを使用してサイト全体にわたって拡張されます。ローカルゲートウェイルータ宛てのトラフィックが DCI 経由で他のサイトに送信されないようにするには、フィルタを作成する必要があります。ゲートウェイルータが 1 つのサイトで到達不能になった場合、トラフィックが正常に他のサイトのゲートウェイに到達できるようにフィルタを削除する必要があります。
- トランスペアレントモードでは、クラスタが HSRP ルータに接続されている場合、ルータの HSRP MAC アドレスを静的 MAC アドレステーブルエントリとして ASA に追加する必要があります（ブリッジグループのスタティック MAC アドレスの追加を参照）。隣接ルータで HSRP が使用される場合、HSRP IP アドレス宛てのトラフィックは HSRP MAC アドレスに送信されますが、リターントラフィックは特定のルータのインターフェイスの MAC アドレスから HSRP ペアで送信されます。したがって、ASA MAC アドレステーブルは通常、HSRP IP アドレスの ASA ARP テ이블エントリが期限切れになり、ASA が ARP 要求を送信して応答を受信した場合にのみ更新されます。ASA の ARP テブルエントリはデフォルトで 14400 秒後に期限切れになりますが、MAC アドレステーブルエントリはデフォルトで 300 秒後に期限切れになるため、MAC アドレステーブルの期限切れトラフィックのドロップを回避するために静的 MAC アドレスエントリが必要です。
- スパンド EtherChannel を使用したルーテッドモードでは、サイト固有の MAC アドレスを設定します。OTV または同様のものを使用してサイト全体にデータ VLAN を拡張します。グローバル MAC アドレス宛てのトラフィックが DCI 経由で他のサイトに送信されないようにするには、フィルタを作成する必要があります。クラスタが 1 つのサイトで到達不能

になった場合、トラフィックが他のサイトのクラスタノードに正常に到達できるようにフィルタを削除する必要があります。ダイナミックルーティングは、サイト間クラスタが拡張セグメントのファースト ホップ ルータとして機能する場合はサポートされません。

その他のガイドライン

- 大々的なトポロジ変更が発生する場合（EtherChannel インターフェイスの追加または削除、ASA 上でのインターフェイスまたはスイッチの有効化または無効化、VSS、vPC、StackWise、または StackWise Virtual を形成するための追加スイッチの追加など）、ヘルスチェック機能や無効なインターフェイスのインターフェイス モニタリングを無効にする必要があります。トポロジの変更が完了して、コンフィギュレーション変更がすべてのユニットに同期されたら、インターフェイスのヘルスチェック機能を再度有効にできます。
- ユニットを既存のクラスタに追加したときや、ユニットをリロードしたときは、一時的に、限定的なパケット/接続ドロップが発生します。これは予定どおりの動作です。場合によっては、ドロップされたパケットが原因で接続がハングすることがあります。たとえば、FTP 接続の FIN/ACK パケットがドロップされると、FTP クライアントがハングします。この場合は、FTP 接続を再確立する必要があります。
- スパンド EtherChannel に接続された Windows 2003 Server を使用している場合、syslog サーバー ポートがダウンし、サーバーが ICMP エラー メッセージを調整しないと、多数の ICMP メッセージが ASA クラスタに送信されます。このようなメッセージにより、ASA クラスタの一部のユニットで CPU 使用率が高くなり、パフォーマンスに影響する可能性があります。ICMP エラー メッセージを調節することを推奨します。
- クラスタ内のすべてのユニットに変更が複製されるまでには時間がかかります。たとえば、オブジェクトグループを使用するアクセスコントロールルール（展開時に複数のルールに分割される）を追加するなどの大きな変更を行うと、変更の完了に必要な時間がクラスタユニットが成功メッセージで応答できるタイムアウトを超える可能性があります。この場合、「failed to replicate command」というメッセージが表示されることがあります。このメッセージは無視できます。

ASA クラスタリングのデフォルト

- スパンド EtherChannel を使用するときは、cLACP システム ID は自動生成され、システムプライオリティはデフォルトで 1 です。
- クラスタのヘルスチェック機能は、デフォルトで有効になり、ホールド時間は 3 秒です。デフォルトでは、すべてのインターフェイスでインターネットヘルス モニタリングが有効になっています。
- 失敗したクラスタ制御リンクのクラスタ再結合機能が 5 分おきに無制限に試行されます。
- 失敗したデータインターフェイスのクラスタ自動再結合機能は、5 分後と、2 に設定された増加間隔で合計で 3 回試行されます。
- 接続再分散は、デフォルトでは無効になっています。接続再分散を有効にした場合の、デフォルトの負荷情報交換間隔は 5 秒です。

- HTTP トラフィックでは、5 秒間の接続複製遅延がデフォルトで有効になっています。

ASA クラスタリングの設定

クラスタリングを設定するには、次のタスクを実行します。



- (注) クラスタリングを有効または無効にするには、コンソール接続（CLI の場合）または ASDM 接続を使用します。

ユニットのケーブル接続およびインターフェイスの設定

クラスタリングを設定する前に、クラスタ制御リンク ネットワーク、管理ネットワーク、およびデータ ネットワークをケーブルで接続します。次に、インターフェイスを設定します。

クラスタ インターフェイスについて

データインターフェイスは、スパンド EtherChannel。また、各ユニットの、少なくとも 1 つのハードウェア インターフェイスをクラスタ制御リンク専用とする必要があります。

クラスタ制御リンク

各ユニットの、少なくとも 1 つのハードウェアインターフェイスをクラスタ制御リンク専用とする必要があります。可能な場合は、クラスタ制御リンクに EtherChannel を使用することを推奨します。

クラスタ制御リンク トラフィックの概要

クラスタ制御リンク トラフィックには、制御とデータの両方のトラフィックが含まれます。

制御トラフィックには次のものが含まれます。

- 制御ノードの選択。
- 設定の複製。
- ヘルス モニタリング。

データ トラフィックには次のものが含まれます。

- 状態の複製。
- 接続所有権クエリおよびデータ パケット転送。

クラスタ制御リンク インターフェイスとネットワーク

次の例外を除き、クラスタ制御リンクには任意のデータ インターフェイスを使用できます。

- VLAN サブインターフェイスをクラスタ制御リンクとして使用することはできません。
- 管理 x/x インターフェイスをクラスタ制御リンクとして使用することはできません（単独か EtherChannel かにかかわらず）。

EtherChannel インターフェイスを使用できます。

各クラスタ制御リンクは、同じサブネット上の IP アドレスを持ちます。このサブネットは、他のすべてのトラフィックからは隔離し、ASA クラスタ制御リンク インターフェイスだけが含まれるようにしてください。

2 メンバー クラスタの場合、ASA と ASA の間をクラスタ制御リンクで直接接続しないでください。インターフェイスを直接接続した場合、一方のユニットで障害が発生すると、クラスタ制御リンクが機能せず、他の正常なユニットも動作しなくなります。スイッチを介してクラスタ制御リンクを接続した場合は、正常なユニットについてはクラスタ制御リンクは動作を維持します。

クラスタ制御リンクのサイジング

可能であれば、各シャーシの予想されるスループットに合わせてクラスタ制御リンクをサイジングする必要があります。そうすれば、クラスタ制御リンクが最悪のシナリオを処理できます。

クラスタ制御リンク トラフィックの内容は主に、状態アップデートや転送されたパケットです。クラスタ制御リンクでのトラフィックの量は常に変化します。転送されるトラフィックの量は、ロードバランシングの有効性、または中央集中型機能のための十分なトラフィックがあるかどうかによって決まります。次に例を示します。

- NAT では接続のロードバランシングが低下するので、すべてのリターン トラフィックを正しいユニットに再分散する必要があります。
- ネットワークアクセスに対する AAA は一元的な機能であるため、すべてのトラフィックが制御ユニットに転送されます。
- メンバーシップが変更されると、クラスタは大量の接続の再分散を必要とするため、一時的にクラスタ制御リンクの帯域幅を大量に使用します。

クラスタ制御リンクの帯域幅を大きくすると、メンバーシップが変更されたときの収束が高速になり、スループットのボトルネックを回避できます。

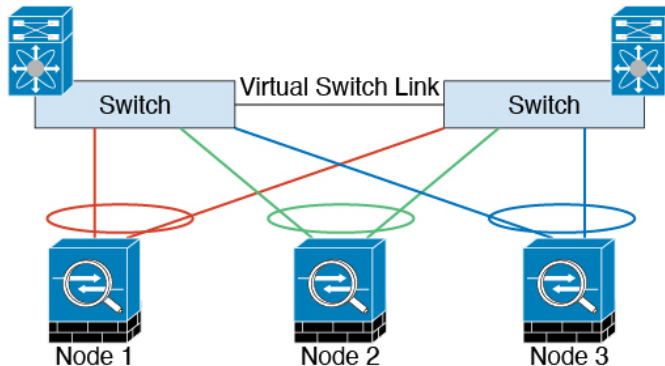


(注) クラスタに大量の非対称（再分散された）トラフィックがある場合は、クラスタ制御リンクのサイズを大きくする必要があります。

クラスタ制御リンクの冗長性

クラスタ制御リンクには EtherChannel を使用することを推奨します。冗長性を実現しながら、EtherChannel 内の複数のリンクにトラフィックを渡すことができます。

次の図は、仮想スイッチングシステム（VSS）、仮想ポートチャンネル（vPC）、StackWise、または StackWise Virtual 環境でクラスタ制御リンクとして EtherChannel を使用する方法を示します。EtherChannel のすべてのリンクがアクティブです。スイッチが冗長システムの一部である場合は、同じ EtherChannel 内のファイアウォールインターフェイスをそれぞれ、冗長システム内の異なるスイッチに接続できます。スイッチインターフェイスは同じ EtherChannel ポートチャンネルインターフェイスのメンバです。複数の個別のスイッチが単一のスイッチのように動作するからです。この EtherChannel は、スパンド EtherChannel ではなく、デバイスローカルであることに注意してください。



クラスタ制御リンクの信頼性

クラスタ制御リンクの機能を保証するには、ユニット間のラウンドトリップ時間（RTT）が 20 ms 未満になるようにします。この最大遅延により、異なる地理的サイトにインストールされたクラスタメンバとの互換性が向上します。遅延を調べるには、ユニット間のクラスタ制御リンクで ping を実行します。

クラスタ制御リンクは、順序の異常やパケットのドロップがない信頼性の高いものである必要があります。たとえば、サイト間の導入の場合、専用リンクを使用する必要があります。

クラスタ制御リンクの障害

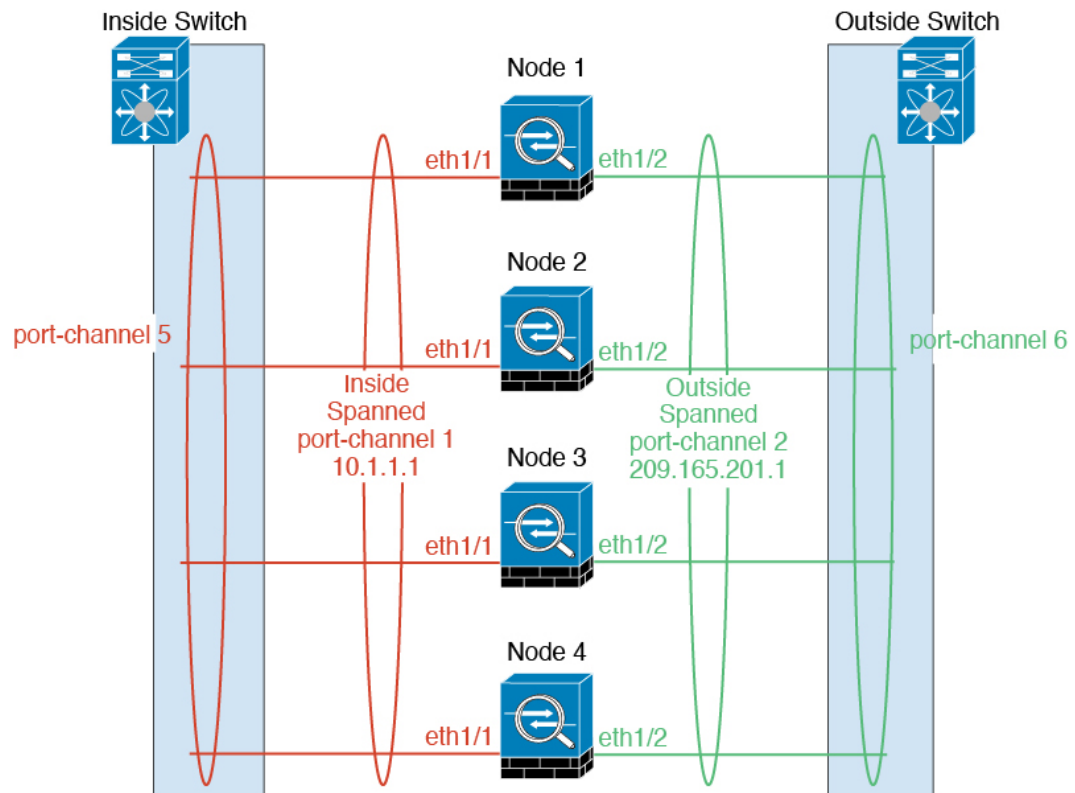
ユニットのクラスタ制御リンク回線プロトコルがダウンした場合、クラスタリングはディセーブルになります。データ インターフェイスはシャットダウンされます。クラスタ制御リンクの修復後、クラスタリングを再度イネーブルにして手動でクラスタに再参加する必要があります。



- (注) ASA が非アクティブになると、すべてのデータ インターフェイスがシャットダウンされます。管理専用インターフェイスのみがトラフィックを送受信できます。管理インターフェイスは、そのユニットがクラスタ IP プールから受け取った IP アドレスを使用して引き続き稼働状態となります。ただし、リロードする場合、クラスタでユニットがまだ非アクティブになっていると、管理インターフェイスはアクセスできません（制御ユニットと同じメイン IP アドレスを使用するため）。それ以降のコンフィギュレーション作業には、コンソールポートを使用する必要があります。

スパンド EtherChannel

シャシーあたり1つ以上のインターフェイスをグループ化して、クラスタのすべてのシャシーに広がる EtherChannel とすることができます。EtherChannel によって、チャンネル内の使用可能なすべてのアクティブインターフェイスのトラフィックが集約されます。スパンド EtherChannel は、ルーテッドとトランスペアレントのどちらのファイアウォールモードでも設定できます。ルーテッドモードでは、EtherChannel は単一の IP アドレスを持つルーテッドインターフェイスとして設定されます。トランスペアレントモードでは、IP アドレスはブリッジグループメンバのインターフェイスではなく BVI に割り当てられます。EtherChannel は初めから、ロードバランシング機能を基本的動作の一部として備えています。



最大スループットのガイドライン

最大スループットを実現するには、次のことを推奨します。

- 使用するロードバランシングハッシュアルゴリズムは「対称」であるようにします。つまり、どちらの方向からのパケットも同じハッシュを持たせて、スパンド EtherChannel 内の同じ ASA に送信します。送信元と宛先の IP アドレス（デフォルト）または送信元と宛先のポートをハッシュアルゴリズムとして使用することを推奨します。
- ASA をスイッチに接続するときは、同じタイプのラインカードを使用します。すべてのパケットに同じハッシュアルゴリズムが適用されるようにするためです。

ロード バランシング

EtherChannel リンクは、送信元または宛先 IP アドレス、TCP ポートおよび UDP ポート番号に基づいて、専用のハッシュ アルゴリズムを使用して選択されます。



- (注) ASA では、デフォルトのロードバランシング アルゴリズムを変更しないでください。スイッチでは、アルゴリズム **source-dest-ip** または **source-dest-ip-port** (Cisco Nexus OS または Cisco IOS の **port-channel load-balance** コマンドを参照) を使用することをお勧めします。クラスタ内の ASA へのトラフィックが均等に分散されなくなることがあるため、ロードバランシング アルゴリズムでは、**vlan** キーワードを使用しないでください。

EtherChannel 内のリンク数はロード バランシングに影響を及ぼします。

対称ロード バランシングは常に可能とは限りません。NAT を設定する場合は、フォワード パケットとリターン パケットとで IP アドレスやポートが異なります。リターン トラフィックはハッシュに基づいて別のユニットに送信されるため、クラスタはほとんどのリターン トラフィックを正しいユニットにリダイレクトする必要があります。

EtherChannel の冗長性

EtherChannel には、冗長性機能が組み込まれています。これは、すべてのリンクの回線プロトコルステータスをモニターします。リンクの1つで障害が発生すると、トラフィックは残りのリンク間で再分散されます。EtherChannel のすべてのリンクが特定のユニット上で停止したが、他方のユニットがまだアクティブである場合は、そのユニットはクラスタから削除されます。

冗長スイッチシステムへの接続

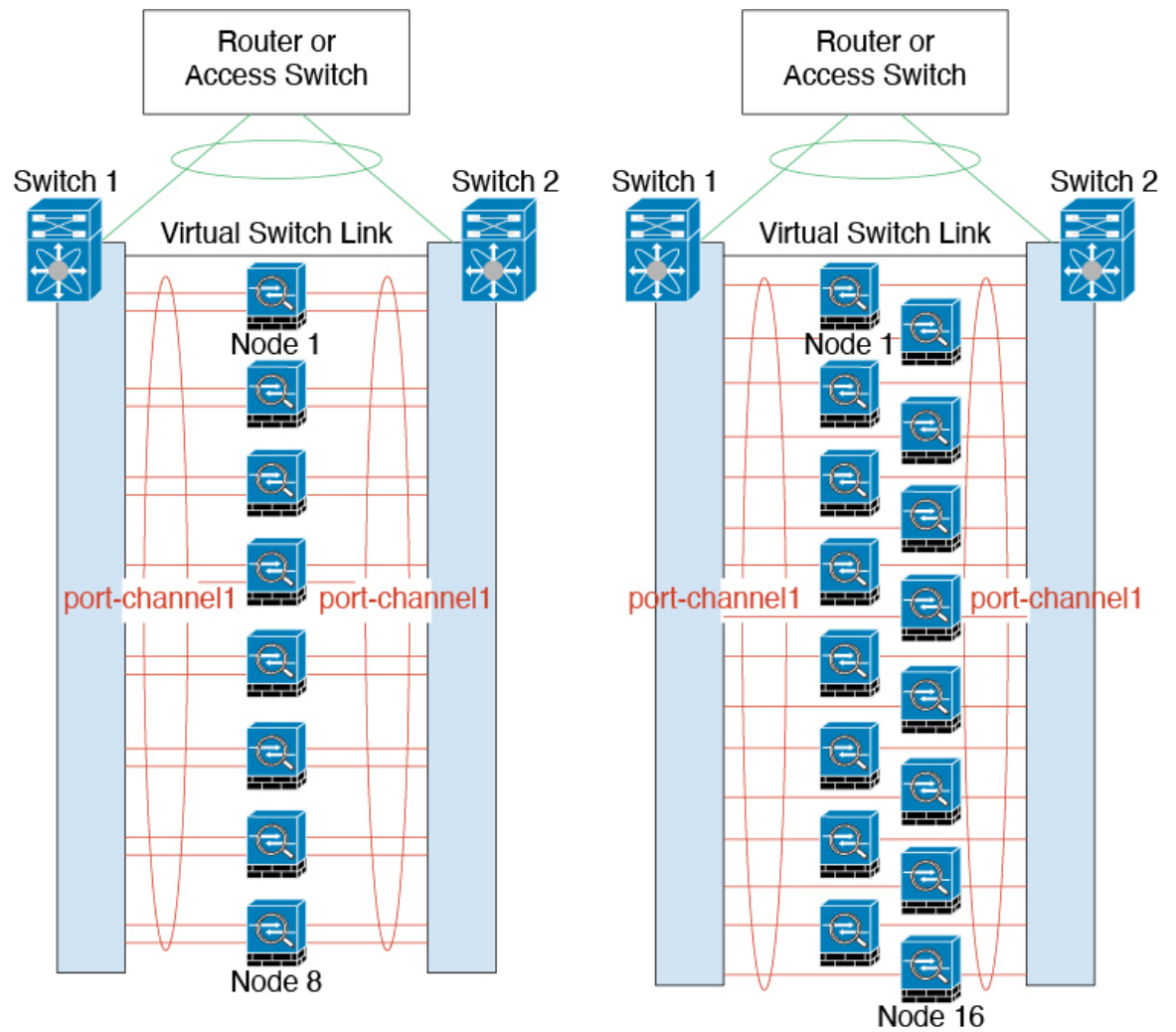
1 つの ASA につき複数のインターフェイスを、スパンド EtherChannel に入れることができます。1 つの ASA につき複数のインターフェイスが特に役立つのは、VSS、vPC、StackWise、または StackWise Virtual の両方のスイッチに接続するときです。

スイッチによっては、スパンド EtherChannel に最大 32 個のアクティブリンクを設定できます。この機能では、vPC 内の両方のスイッチが、それぞれ 16 個のアクティブリンクの EtherChannel をサポートする必要があります (例: Cisco Nexus 7000 と F2 シリーズ 10 ギガビットイーサネット モジュール)。

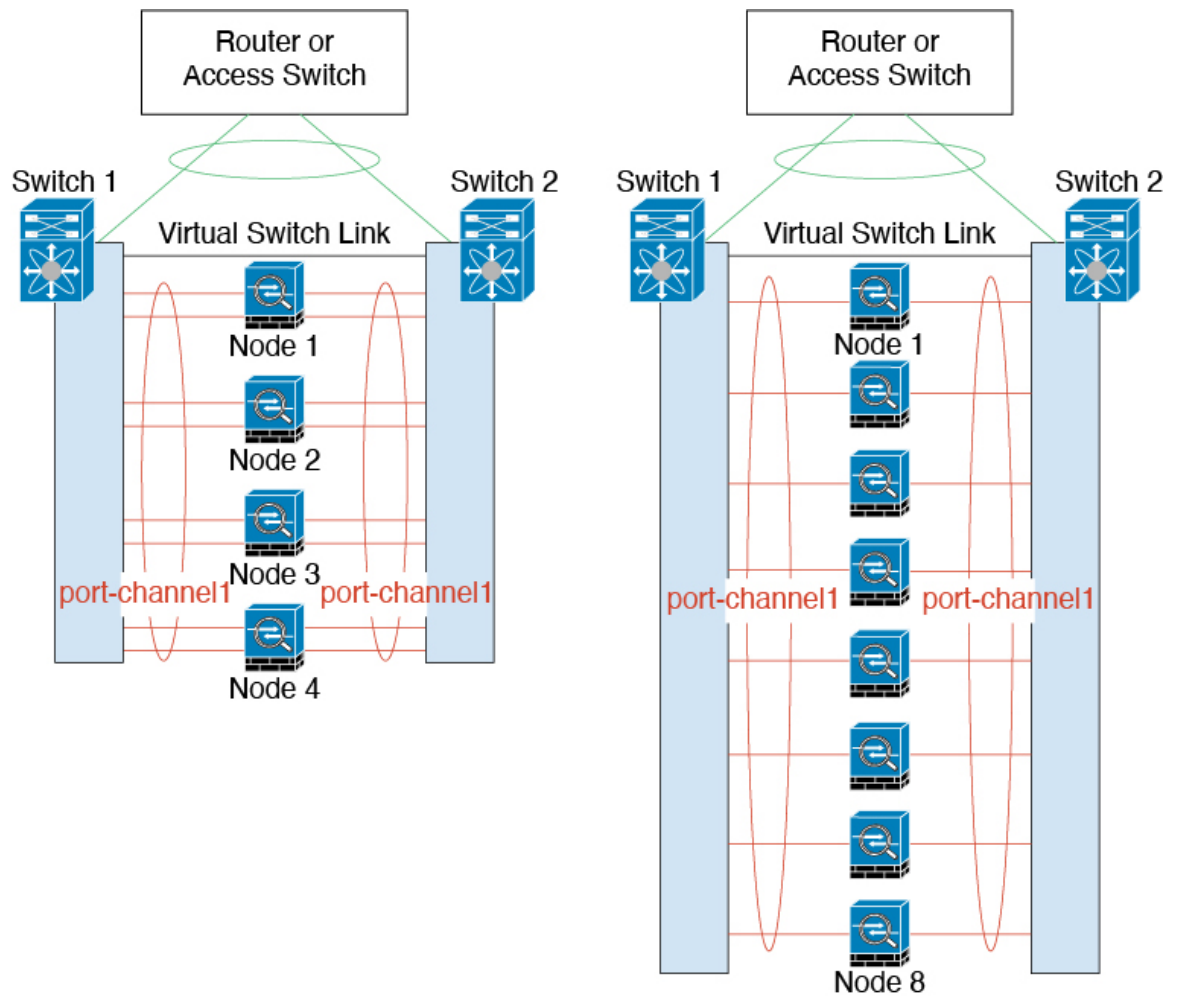
EtherChannel で 8 個のアクティブリンクをサポートするスイッチの場合、冗長システムで 2 台のスイッチに接続すると、スパンド EtherChannel に最大 16 個のアクティブリンクを設定できます。

スパンド EtherChannel で 8 個より多くのアクティブリンクを使用する場合は、スタンバイリンクも使用できません。9 ~ 32 個のアクティブリンクをサポートするには、スタンバイリンクの使用を可能にする cLACP ダイナミック ポートプライオリティをディセーブルにする必要があります。それでも、必要であれば、たとえば 1 台のスイッチに接続するときに、8 個のアクティブリンクと 8 個のスタンバイリンクを使用できます。

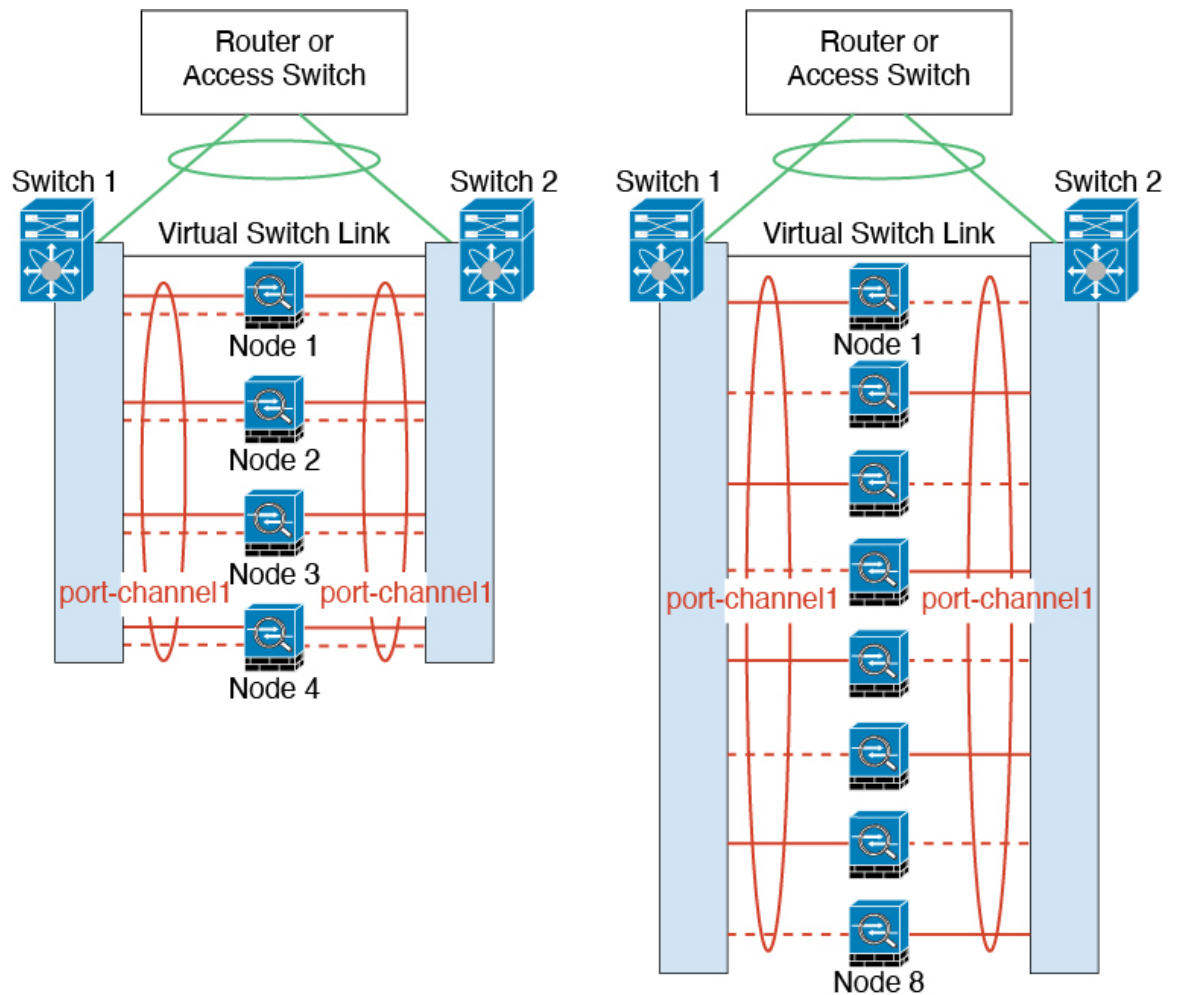
次の図では、8 ノードクラスタおよび 16 ノードクラスタでの 32 アクティブリンクのスパンド EtherChannel を示します。



次の図では、4 ノードクラスタおよび 8 ノードクラスタでの 16 アクティブリンクのスパンド EtherChannel を示します。



次の図では、4 ノードクラスターおよび 8 ノードクラスターでの従来の 8 アクティブ/8 スタンバイリンクのスパンド EtherChannel を示します。アクティブリンクは実線で、非アクティブリンクは点線で示しています。cLACP ロードバランシングは、EtherChannel のリンクのうち最良の 8 本を自動的に選択してアクティブにできます。つまり、cLACP は、リンクレベルでのロードバランシング実現に役立ちます。



クラスタユニットのケーブル接続とアップストリームおよびダウンストリーム機器の設定

クラスタリングを設定する前に、クラスタ制御リンク ネットワーク、管理ネットワーク、およびデータ ネットワークをケーブルで接続します。

手順

クラスタ制御リンク ネットワーク、管理ネットワーク、およびデータ ネットワークをケーブルで接続します。

(注) クラスタに参加するようにユニットを設定する前に、少なくとも、アクティブなクラスタ制御リンク ネットワークが必要です。

アップストリームとダウンストリームの機器も設定する必要があります。たとえば、EtherChannelを使用する場合は、EtherChannelのアップストリーム/ダウンストリーム機器を設定する必要があります。

各ユニットでのクラスタ インターフェイス モードの設定

クラスタリングを有効にする前に、スパンドEtherChannelを使用するようにファイアウォールを変換する必要があります。クラスタリングによって使用できるインターフェイスの種類が制限されるため、このプロセスでは、既存の設定に互換性のないインターフェイスがあるかどうかを確認し、サポートされていないインターフェイスを設定できないようにします。

始める前に

- モードの設定は、クラスタに追加する各 ASA で個別に行う必要があります。
- 管理専用インターフェイスはいつでも、個別インターフェイス（推奨）として設定できます。管理インターフェイスは、個別インターフェイスとすることができます（トランスペアレント ファイアウォール モードのときでも）。
- 管理インターフェイスを個別インターフェイスとして設定すると、管理インターフェイスに対してダイナミックルーティングをイネーブルにできません。スタティックルートを使用する必要があります。

手順

- ステップ 1** 互換性のないコンフィギュレーションを表示し、強制的にインターフェイスモードにして後でコンフィギュレーションを修正できるようにします。このコマンドではモードは変更されません。

cluster interface-mode spanned check-details

例：

```
ciscoasa(config)# cluster interface-mode spanned check-details
```

- ステップ 2** クラスタリング用にインターフェイス モードを設定します。

cluster interface-mode spanned force

例：

```
ciscoasa(config)# cluster interface-mode spanned force
```

デフォルト設定はありません。明示的にモードを選択する必要があります。モードを設定していない場合は、クラスタリングをイネーブルにできません。

force オプションを指定すると、互換性のないコンフィギュレーションの検査は行わずにモードが変更されます。コンフィギュレーションの問題がある場合は、モードを変更した後に手動で解決する必要があります。インターフェイス コンフィギュレーションの修正ができるのはモードの設定後に限られるので、**force** オプションを使用することを推奨します。このようにすれば、最低でも、既存のコンフィギュレーションの状態から開始できます。さらにガイドランスが必要な場合は、モードを設定した後で **check-details** オプションを再実行します。

force オプションを指定しないと、互換性のないコンフィギュレーションがある場合は、コンフィギュレーションをクリアしてリロードするように求められるので、コンソールポートに接続して管理アクセスを再設定する必要があります。コンフィギュレーションに互換性の問題がない場合は（まれなケース）、モードが変更され、コンフィギュレーションは維持されます。コンフィギュレーションをクリアしたくない場合は、**n** を入力してコマンドを終了します。

インターフェイス モードを解除するには、**no cluster interface-mode** コマンドを入力します。

制御ユニットでのインターフェイスの設定

クラスタリングを有効にする前に、現在 IP アドレスが設定されているインターフェイスをクラスタ対応に変更する必要があります。他のインターフェイスについては、クラスタリングをイネーブルにする前またはした後で設定できます。完全なコンフィギュレーションが新しいクラスタメンバと同期するように、すべてのインターフェイスを事前に設定することを推奨します。

ここでは、クラスタリング互換となるようにインターフェイスを設定する方法について説明します。

管理インターフェイスを個別インターフェイスとして設定する

個別インターフェイスは通常のルーテッドインターフェイスであり、それぞれが専用の IP アドレスを IP アドレス プールから取得します。メイン クラスタ IP アドレスは、そのクラスタのための固定アドレスであり、常に現在のプライマリ ユニットに属します。

管理インターフェイスを個別インターフェイスとして設定することを推奨します。個別管理インターフェイスならば、必要に応じて各ユニットに直接接続できますが、スバンド EtherChannel インターフェイスでは、現在のプライマリ ユニットへの接続しかできません。

始める前に

- マルチ コンテキスト モードの場合は、この手順を各コンテキストで実行します。まだコンテキスト コンフィギュレーション モードに入っていない場合は、**changeto context name** コマンドを入力します。
- (オプション) インターフェイスをデバイスローカル EtherChannel インターフェイスとして設定する、およびサブインターフェイスを設定する作業を必要に応じて行います。
 - EtherChannel の場合、この EtherChannel はユニットに対してローカルであり、スバンド EtherChannel ではありません。

手順

- ステップ 1** ローカル IP アドレス（IPv4 と IPv6 の一方または両方）のプールを設定します。このアドレスの 1 つが、このインターフェイス用に各クラスタ ユニットに割り当てられます。

(IPv4)

ip local pool *poolname first-address — last-address [mask mask]*

(IPv6)

ipv6 local pool *poolname ipv6-address/prefix-length number_of_addresses*

例：

```
ciscoasa(config)# ip local pool ins 192.168.1.2-192.168.1.9
ciscoasa(config-if)# ipv6 local pool insipv6 2001:DB8:45:1002/64 8
```

最低でも、クラスタ内のユニット数と同じ数のアドレスが含まれるようにしてください。クラスタを拡張する予定の場合は、アドレスを増やします。現在のプライマリユニットに属するメインクラスタ IP アドレスは、このプールの一部ではありません。必ず、同じネットワークの IP アドレスの 1 つをメインクラスタ IP アドレス用に確保してください。

各ユニットに割り当てられるローカルアドレスを、事前に正確に特定することはできません。各ユニットで使用されているアドレスを表示するには、**show ip[v6] local pool poolname** コマンドを入力します。各クラスタメンバには、クラスタに参加したときにメンバ ID が割り当てられます。この ID によって、プールから使用されるローカル IP が決定します。

- ステップ 2** 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。

interface *interface_id*

例：

```
ciscoasa(config)# interface management 1/1
```

- ステップ 3** インターフェイスを管理専用モードに設定してトラフィックが通過しないようにします。

management-only

デフォルトでは、管理タイプのインターフェイスは管理専用として設定されます。トランスペアレントモードでは、このコマンドは管理タイプのインターフェイスに対して常にイネーブルになります。

- ステップ 4** インターフェイスの名前を指定します。

nameif *name*

例：

```
ciscoasa(config-if)# nameif management
```

name は最大 48 文字のテキスト文字列です。大文字と小文字は区別されません。名前を変更するには、このコマンドで新しい値を再入力します。

ステップ 5 メインクラスタの IP アドレスを設定し、クラスタ プールを指定します。

(IPv4)

ip address *ip_address* [*mask*] **cluster-pool** *poolname*

(IPv6)

ipv6 address *ipv6-address/prefix-length* **cluster-pool** *poolname*

例 :

```
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0 cluster-pool ins
ciscoasa(config-if)# ipv6 address 2001:DB8:45:1002::99/64 cluster-pool insipv6
```

この IP アドレスは、クラスタ プールアドレスと同じネットワーク上に存在している必要がありますが、プールに含まれてはなりません。IPv4 アドレスと IPv6 アドレスの一方または両方を設定できます。

DHCP、PPPoE、および IPv6 自動設定はサポートされません。IP アドレスを手動で設定する必要があります。

ステップ 6 セキュリティ レベルを設定します。 *number* には、0 (最低) ~ 100 (最高) の整数を指定します。

security-level *number*

例 :

```
ciscoasa(config-if)# security-level 100
```

ステップ 7 インターフェイスをイネーブルにします。

no shutdown

例

次の例では、イーサネット 1/3 およびイーサネット 1/4 インターフェイスをデバイス ローカル EtherChannel として設定してから、この EtherChannel を個別インターフェイスとして設定します。

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8:45:1002/64 8
```

```
interface ethernet 1/3
channel-group 1 mode active
no shutdown
```

```
interface ethernet 1/4
```

```
channel-group 1 mode active
no shutdown

interface port-channel 1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8:45:1002::99/64 cluster-pool mgmtipv6
security-level 100
management-only
```

スパンド EtherChannel の設定

スパンド EtherChannel は、クラスタ内のすべての ASA に広がるものであり、EtherChannel の動作の一部としてロード バランシングを行うことができます。

始める前に

- スパンド EtherChannel インターフェイス モードにする必要があります。
- マルチ コンテキスト モードの場合は、この手順をシステム実行スペースで開始します。まだシステム コンフィギュレーション モードに入っていない場合は、**changeto system** コマンドを入力します。
- トランスペアレント モードの場合は、ブリッジ グループを設定します。[ブリッジ仮想インターフェイス \(BVI\) の設定](#)を参照してください。
- EtherChannel には最大および最小のリンク数を指定しないでください。EtherChannel の最大および最小のリンク数の指定 (**lacp max-bundle** コマンドと **port-channel min-bundle** コマンド) は、ASA とスイッチのどちらにおいても行わないことを推奨します。これらを使用する必要がある場合は、次の点に注意してください。
 - ASA 上で設定されるリンクの最大数は、クラスタ全体のアクティブ ポートの合計数です。スイッチ上で設定された最大リンク数の値が、ASA での値を超えていないことを確認してください。
 - ASA 上で設定される最小リンク数は、ポートチャネル インターフェイスを起動するための最小アクティブポート数 (ユニットあたり) です。スイッチ上では、最小リンク数はクラスタ全体の最小リンク数であるため、この値は ASA での値とは一致しません。
- デフォルトのロードバランシング アルゴリズムを変更しないでください (**port-channel load-balance** コマンドを参照)。スイッチでは、アルゴリズム **source-dest-ip** または **source-dest-ip-port** (Cisco Nexus OS および Cisco IOS の **port-channel load-balance** コマンドを参照) を使用することをお勧めします。クラスタ内の ASA へのトラフィックが均等に分散されなくなることがあるため、ロードバランシング アルゴリズムでは、**vlan** キーワードを使用しないでください。
- **lacp port-priority** コマンドと **lacp system-priority** コマンドは、スパンド EtherChannel には使用されません。

- スパンド EtherChannel を使用している場合、クラスタリングが完全にイネーブルになるまで、ポートチャンネルインターフェイスは起動しません。この要件により、クラスタのアクティブではないユニットにトラフィックが転送されるのが防がれます。

手順

ステップ 1 チャンネルグループに追加するインターフェイスを指定します。

interface *physical_interface*

例 :

```
ciscoasa(config)# interface ethernet 1/1
```

physical_interface ID には、タイプ、スロット、およびポート番号 (type slot/port) が含まれます。チャンネルグループのこの最初のインターフェイスによって、グループ内の他のすべてのインターフェイスのタイプと速度が決まります。

ステップ 2 EtherChannel にこのインターフェイスを割り当てます。

channel-group *channel_id* **mode active** [**vss-id** {**1** | **2**}]

例 :

```
ciscoasa(config-if)# channel-group 1 mode active
```

channel_id は 1 ~ 48 です。このチャンネル ID のポートチャンネル インターフェイスがコンフィギュレーションにまだ存在しない場合は、自動的に追加されます。

interface port-channel *channel_id*

active モードだけがスパンド EtherChannel に対してサポートされます。

VSS、vPC、StackWise、または StackWise Virtual の 2 台のスイッチに ASA を接続する場合は、このインターフェイスをどのスイッチに接続するかを指定するために **vss-id** キーワードを設定します (1 または 2)。また、ステップ 6 で **port-channel span-cluster vss-load-balance** コマンドをポートチャンネルインターフェイスに対して使用する必要があります。

ステップ 3 インターフェイスをイネーブルにします。

no shutdown

ステップ 4 (オプション) EtherChannel にさらにインターフェイスを追加するには、上記のプロセスを繰り返します。

例 :

```
ciscoasa(config)# interface ethernet 1/2
ciscoasa(config-if)# channel-group 1 mode active
ciscoasa(config-if)# no shutdown
```

ユニットごとに複数のインターフェイスが EtherChannel に含まれていると、VSS、vPC、StackWise、または StackWise Virtual のスイッチに接続する場合に役立ちます。デフォルトでは、クラスタの全メンバで最大 16 個のアクティブ インターフェイスのうち、スパンド EtherChannel が使用できるのは 8 個だけであることを注意してください。残りの 8 インターフェイスはリンク障害時のためのスタンバイです。8 個より多くのアクティブ インターフェイスを使用するには（ただしスタンバイ インターフェイスではなく）、**clacp static-port-priority** コマンドを使用してダイナミック ポート プライオリティをディセーブルにします。ダイナミック ポート プライオリティをディセーブルにすると、クラスタ全体で最大 32 個のアクティブ リンクを使用できます。たとえば、16 台の ASA から成るクラスタの場合は、各 ASA で最大 2 個のインターフェイスを使用でき、スパンド EtherChannel の合計は 32 インターフェイスとなります。

ステップ 5 ポートチャネル インターフェイスを指定します。

```
interface port-channel channel_id
```

例：

```
ciscoasa(config)# interface port-channel 1
```

このインターフェイスは、チャネルグループにインターフェイスを追加したときに自動的に作成されたものです。

ステップ 6 この EtherChannel をスパンド EtherChannel として設定します。

```
port-channel span-cluster [vss-load-balance]
```

例：

```
ciscoasa(config-if)# port-channel span-cluster
```

ASA を VSS、vPC、StackWise、または StackWise Virtual の 2 台のスイッチに接続する場合は、**vss-load-balance** キーワードを使用して VSS ロードバランシングをイネーブルにする必要があります。この機能を使用すると、ASA と VSS（または vPC、StackWise、StackWise Virtual vPC）ペアとの間の物理リンク接続の負荷が確実に分散されます。ロードバランシングをイネーブルにする前に、各メンバー インターフェイスに対して **channel-group** コマンドの **vss-id** キーワードを設定する必要があります（ステップ 2 を参照）。

ステップ 7 （オプション）ポートチャネル インターフェイスのイーサネット プロパティを設定します。この設定は、個別インターフェイスに対して設定されたプロパティよりも優先されます。

これらのパラメータはチャネルグループのすべてのインターフェイスで一致している必要があるため、この方法はこれらのパラメータを設定するショートカットになります。

ステップ 8 （オプション）この EtherChannel 上に VLAN サブインターフェイスを作成する予定の場合は、この時点で作成します。

例：

```
ciscoasa(config)# interface port-channel 1.10
```

```
ciscoasa(config-if)# vlan 10
```

この手順の残りの部分は、サブインターフェイスに適用されます。

- ステップ 9** (マルチコンテキストモード) コンテキストにインターフェイスを割り当てます。その後で、次のとおりに入力します。

```
changeto context name
interface port-channel channel_id
```

例 :

```
ciscoasa(config)# context admin
ciscoasa(config)# allocate-interface port-channell
ciscoasa(config)# changeto context admin
ciscoasa(config-if)# interface port-channel 1
```

マルチコンテキストモードの場合は、インターフェイスコンフィギュレーションの残りの部分は各コンテキスト内で行われます。

- ステップ 10** インターフェイスの名前を指定します。

```
nameif name
```

例 :

```
ciscoasa(config-if)# nameif inside
```

name は最大 48 文字のテキスト文字列です。大文字と小文字は区別されません。名前を変更するには、このコマンドで新しい値を再入力します。

- ステップ 11** ファイアウォールモードに応じて、次のいずれかを実行します。

- ルーテッドモード : IPv4 アドレスと IPv6 アドレスの一方または両方を設定します。

(IPv4)

```
ip address ip_address [mask]
```

(IPv6)

```
ipv6 address ipv6-prefix/prefix-length
```

例 :

```
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
ciscoasa(config-if)# ipv6 address 2001:DB8::1001/32
```

DHCP、PPPoE、および IPv6 自動設定はサポートされません。ポイントツーポイント接続の場合、31 ビットのサブネットマスク (255.255.255.254) を指定できます。この場合、ネットワークまたはブロードキャストアドレス用の IP アドレスは予約されません。

- トランスペアレントモード : インターフェイスをブリッジグループに割り当てます。

```
bridge-group number
```

例 :

```
ciscoasa(config-if)# bridge-group 1
```

number は、1 ~ 100 の整数です。ブリッジグループには最大 64 個のインターフェイスを割り当てることができます。同一インターフェイスを複数のブリッジグループに割り当てることはできません。BVI のコンフィギュレーションには IP アドレスが含まれていることに注意してください。

ステップ 12 セキュリティ レベルを設定します。

security-level *number*

例 :

```
ciscoasa(config-if)# security-level 50
```

number には、0 (最下位) ~ 100 (最上位) の整数を指定します。

ステップ 13 潜在的なネットワークの接続問題を回避するために、スパンド EtherChannel のグローバル MAC アドレスを設定します。

mac-address *mac_address*

例 :

```
ciscoasa(config-if)# mac-address 000C.F142.4CDE
```

MAC アドレスが手動設定されている場合、その MAC アドレスは現在の制御ユニットに留まります。MAC アドレスを設定していない場合に、制御ユニットが変更された場合、新しい制御ユニットはインターフェイスに新しい MAC アドレスを使用します。これにより、一時的なネットワークの停止が発生する可能性があります。

マルチコンテキストモードでは、コンテキスト間でインターフェイスを共有する場合は、MAC アドレスの自動生成を有効にして、手動で MAC アドレスを設定しなくてすむようにします。非共有インターフェイスの場合は、このコマンドを使用して MAC アドレスを手動で設定する必要がありますことに注意してください。

mac_address は、H.H.H 形式で指定します。H は 16 ビットの 16 進数です。たとえば、MAC アドレス 00-0C-F1-42-4C-DE は、000C.F142.4CDE と入力します。

自動生成された MAC アドレスも使用する場合、手動で割り当てる MAC アドレスの最初の 2 バイトには A2 を使用できません。

ステップ 14 (ルーテッドモード) サイト間クラスタリングの場合、サイトごとにサイト固有の MAC アドレスおよび IP アドレスを設定します。

mac-address *mac_address site-id number site-ip ip_address*

例 :

```
ciscoasa(config-if)# mac-address aaaa.1111.1234
```

```
ciscoasa(config-if)# mac-address aaaa.1111.aaaa site-id 1 site-ip 10.9.9.1
ciscoasa(config-if)# mac-address aaaa.1111.bbbb site-id 2 site-ip 10.9.9.2
ciscoasa(config-if)# mac-address aaaa.1111.cccc site-id 3 site-ip 10.9.9.3
ciscoasa(config-if)# mac-address aaaa.1111.dddd site-id 4 site-ip 10.9.9.4
```

サイト固有の IP アドレスは、グローバル IP アドレスと同じサブネット上にある必要があります。ユニットで使用するサイト固有の MAC アドレスおよび IP アドレスは、各ユニットのブートストラップコンフィギュレーションに指定したサイト ID によって異なります。

ブートストラップコンフィギュレーションの作成

クラスタ内の各ノードがクラスタに参加するには、ブートストラップ設定が必要です。

制御ノードのブートストラップの設定

クラスタ内の各ノードがクラスタに参加するには、ブートストラップ設定が必要です。一般的には、クラスタに参加するように最初に設定したノードが制御ノードとなります。クラスタリングをイネーブルにした後で、選定期間が経過すると、クラスタの制御ノードが選定されます。最初はクラスタ内に1つのノードしかないため、そのノードが制御ノードになります。クラスタに追加する後続のノードはデータノードになります。

始める前に

- コンフィギュレーションをバックアップします。後でクラスタから脱退する必要があるときに備えて、コンフィギュレーションを復元できるようにしておくためです。
- マルチ コンテキスト モードの場合、システム実行スペースで次の手順を実行します。コンテキストからシステム実行スペースに切り替えるには、**changeto system** コマンドを入力します。
- クラスタリングをイネーブルまたはディセーブルにするには、コンソールポートを使用する必要があります。Telnet または SSH を使用することはできません。
- 稼働中のクラスタにノードを追加すると、一時的に、限定的なパケット/接続ドロップが発生することがありますが、これは想定内の動作です。
- クラスタ制御リンクのサイズをあらかじめ決定しておきます。[クラスタ制御リンクのサイジング \(16 ページ\)](#) を参照してください。

手順

- ステップ 1** クラスタに参加する前に、クラスタ制御リンク インターフェイスをイネーブルにします。後でクラスタリングをイネーブルにするときに、このインターフェイスをクラスタ制御リンクとして識別します。

十分な数のインターフェイスがある場合は、複数のクラスタ制御リンクインターフェイスを結合して 1 つの EtherChannel とすることを推奨します。この EtherChannel は ASA に対してローカルであり、スパンド EtherChannel ではありません。

クラスタ制御リンクインターフェイスコンフィギュレーションは、制御ノードからデータノードには複製されませんが、同じコンフィギュレーションを各ノードで使用する必要があります。このコンフィギュレーションは複製されないため、クラスタ制御リンクインターフェイスの設定は各ノードで個別に行う必要があります。

- VLAN サブインターフェイスをクラスタ制御リンクとして使用することはできません。
- 管理 x/x インターフェイスをクラスタ制御リンクとして使用することはできません（単独か EtherChannel かにかかわらず）。

- a) 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。

interface *interface_id*

例 :

```
ciscoasa(config)# interface ethernet 1/6
```

- b) （任意、EtherChannel の場合）EtherChannel にこの物理インターフェイスを割り当てます。

channel-group *channel_id* **mode on**

例 :

```
ciscoasa(config-if)# channel-group 1 mode on
```

channel_id は 1 ~ 48 です。このチャンネル ID のポートチャンネルインターフェイスがコンフィギュレーションにまだ存在しない場合は、自動的に追加されます。

interface port-channel *channel_id*

クラスタ制御リンクでの不要なトラフィックを削減できるように、クラスタ制御リンクのメンバーインターフェイスに対しては On モードを使用することを推奨します。クラスタ制御リンクは LACP トラフィックのオーバーヘッドを必要としません。これは隔離された、安定したネットワークであるからです。注：データ EtherChannel を Active モードに設定することをお勧めします。

- c) インターフェイスをイネーブルにします。

no shutdown

必要があるのはインターフェイスのイネーブル化だけです。インターフェイスの名前などのパラメータを設定しないでください。

- d) （EtherChannel の場合）EtherChannel に追加するインターフェイスごとに繰り返します。

例 :

```
ciscoasa(config)# interface ethernet 1/7
ciscoasa(config-if)# channel-group 1 mode on
```

```
ciscoasa(config-if)# no shutdown
```

- ステップ 2** クラスタ制御リンクインターフェイスの最大伝送ノードを指定します。データインターフェイスの最大 MTU より少なくとも 100 バイト高い値を指定します。

mtu cluster bytes

例：

```
ciscoasa(config)# mtu cluster 9198
```

MTU を 1400 ～ 9198 バイトの間で設定します。デフォルトの MTU は 1500 バイトです。クラスタ制御リンクの MTU を最大値。クラスタ制御リンクのトラフィックにはデータパケット転送が含まれるため、クラスタ制御リンクはデータパケット全体のサイズに加えてクラスタトラフィックのオーバーヘッドにも対応する必要があります。

たとえば、最大 MTU は 9198 バイトであるため、データインターフェイスの最大 MTU は 9098 になり、クラスタ制御リンクは 9198 に設定できます。

このコマンドはグローバル コンフィギュレーション コマンドですが、ノード間で複製されないブートストラップ コンフィギュレーションの一部でもあります。

- ステップ 3** クラスタに名前を付け、クラスタ コンフィギュレーション モードにします。

cluster group name

例：

```
ciscoasa(config)# cluster group pod1
```

名前は 1 ～ 38 文字の ASCII 文字列であることが必要です。ノードごとに設定できるクラスタグループは 1 つだけです。クラスタのすべてのメンバが同じ名前を使用する必要があります。

- ステップ 4** クラスタのこのメンバの名前を指定します。

local-unit unit_name

1 ～ 38 文字の一意の ASCII 文字列を使用します。各ノードには一意の名前が必要です。クラスタ内の他のノードと同じ名前を付けることはできません。

例：

```
ciscoasa(cfg-cluster)# local-unit node1
```

- ステップ 5** クラスタ制御リンク インターフェイス (EtherChannel を推奨) を指定します。

cluster-interface interface_id ip ip_address mask

例：

```
ciscoasa(cfg-cluster)# cluster-interface port-channel2 ip 192.168.1.1 255.255.255.0
INFO: Non-cluster interface config is cleared on Port-Channel2
```

サブインターフェイスと管理インターフェイスは許可されません。

IP アドレスには IPv4 アドレスを指定します。IPv6 は、このインターフェイスではサポートされません。このインターフェイスには、**nameif** を設定することはできません。

ノードごとに、同じネットワーク上の異なる IP アドレスを指定します。

- ステップ 6** サイト間クラスタリングを使用している場合、このノードのサイト ID を設定し、サイト固有の MAC アドレスが使用されるようにします。

site-id number

例 :

```
ciscoasa(cfg-cluster)# site-id 1
```

number には、1 ~ 8 の範囲内の値を入力します。

- ステップ 7** 制御ノードの選択に対するこのノードのプライオリティを設定します。

priority priority_number

例 :

```
ciscoasa(cfg-cluster)# priority 1
```

プライオリティは 1 ~ 100 であり、1 が最高のプライオリティです。

- ステップ 8** (オプション) クラスタ制御リンクの制御トラフィックの認証キーを設定します。

key shared_secret

例 :

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

共有秘密は、1 ~ 63 文字の ASCII 文字列です。共有秘密は、キーを生成するために使用されます。このコマンドは、データパストラフィック (接続状態アップデートや転送されるパケットなど) には影響しません。データパストラフィックは、常にクリアテキストとして送信されます。

- ステップ 9** (オプション) LACP のダイナミック ポートプライオリティをディセーブルにします。

clacp static-port-priority

一部のスイッチはダイナミック ポートプライオリティをサポートしていないため、このコマンドはスイッチの互換性を高めます。さらに、このコマンドは、9 ~ 32 のアクティブ スパンド EtherChannel メンバーのサポートをイネーブルにします。このコマンドを使用しないと、サポートされるのは 8 個のアクティブ メンバと 8 個のスタンバイ メンバのみです。このコマンドをイネーブルにした場合、スタンバイメンバは使用できません。すべてのメンバがアクティブです。

- ステップ 10** (オプション) cLACP システム ID およびシステムのプライオリティを手動で指定します。

clacp system-mac {*mac_address* | **auto**} [**system-priority number**]

例 :

```
ciscoasa(cfg-cluster)# clacp system-mac 000a.0000.aaaa
```

スバンド EtherChannel を使用するときは、ASA は cLACP を使用してネイバー スイッチとの間で EtherChannel のネゴシエーションを行います。cLACP ネゴシエーションの際に、同じクラスタ内の ASA は互いに連携するため、スイッチには 1 つの (仮想) デバイスであるかのように見えます。cLACP ネゴシエーションのパラメータの 1 つであるシステム ID は、MAC アドレスの形式をとります。クラスタ内のすべての ASA が同じシステム ID を使用します。これは制御ノードによって自動生成され (デフォルト)、すべてのセカンダリノードに複製されます。あるいは、このコマンドに *H.H.H* の形式で手動で指定することもできます。H は 16 ビットの 16 進数です。(たとえば、MAC アドレス 00-0A-00-00-AA-AA は、000A.0000.AAAA と入力します)。トラブルシューティングの目的で、たとえば、識別が容易な MAC アドレスを使用できるように、手動で MAC アドレスを設定することがあります。一般的には、自動生成された MAC アドレスを使用します。

システムプライオリティ (1 ~ 65535) は、どのノードがバンドルの決定を行うかを定めるために使用されます。デフォルトでは、ASA はプライオリティ 1 (最高のプライオリティ) を使用します。このプライオリティは、スイッチのプライオリティよりも高いことが必要です。

このコマンドは、ブートストラップ設定の一部ではなく、制御ノードからデータノードに複製されます。ただし、クラスタリングをイネーブルにした後は、この値は変更できません。

ステップ 11 クラスタリングをイネーブルにします。

enable [**noconfirm**]

例 :

```
ciscoasa(cfg-cluster)# enable
INFO: Clustering is not compatible with following commands:
policy-map global_policy
  class inspection_default
    inspect skinny
policy-map global_policy
  class inspection_default
    inspect sip
Would you like to remove these commands? [Y]es/[N]o:Y

INFO: Removing incompatible commands from running configuration...
Cryptochecksum (changed): f16b7fc2 a742727e e40bc0b0 cd169999
INFO: Done
```

enable コマンドが入力されると、ASA は実行コンフィギュレーションをスキャンして、クラスタリングに対応していない機能の非互換コマンドの有無を調べます。デフォルトコンフィギュレーションにあるコマンドも、これに該当することがあります。互換性のないコマンドを削除するように求められます。応答として **No** を入力した場合は、クラスタリングはイネーブルになりません。確認を省略し、互換性のないコマンドを自動的に削除するには、**noconfirm** キーワードを使用します。

最初にイネーブルにしたノードについては、制御ノード選定が発生します。これまでは最初のノードがクラスタの唯一のメンバーである必要があるため、これが制御ノードになります。この期間中にコンフィギュレーション変更を実行しないでください。

クラスタリングをディセーブルにするには、**no enable** コマンドを入力します。

(注) クラスタリングをディセーブルにした場合は、すべてのデータインターフェイスがシャットダウンされ、管理専用インターフェイスだけがアクティブになります。

例

次の例では、管理インターフェイスを設定し、クラスタ制御リンク用のデバイスローカル EtherChannel を設定し、その後で、「node1」という名前の ASA のクラスタリングをイネーブルにします。これは最初にクラスタに追加されるノードであるため、制御ノードになります。

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/32 8
interface management 1/1
  nameif management
  ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
  ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
  security-level 100
  management-only
  no shutdown

interface ethernet 1/6
  channel-group 1 mode on
  no shutdown

interface ethernet 1/7
  channel-group 1 mode on
  no shutdown

cluster group pod1
  local-unit node1
  cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
  priority 1
  key chuntheunavoidable
  enable noconfirm
```

データノードのブートストラップの設定

データノードを設定するには、次の手順に従います。

始める前に

- クラスタリングをイネーブルまたはディセーブルにするには、コンソールポートを使用する必要があります。Telnet または SSH を使用することはできません。

- コンフィギュレーションをバックアップします。後でクラスタから脱退する必要があるときに備えて、コンフィギュレーションを復元できるようにしておくためです。
- マルチ コンテキスト モードでは、システム実行スペースで次の手順を実行します。コンテキストからシステム実行スペースに切り替えるには、**changeto system** コマンドを入力します。
- コンフィギュレーション内に、クラスタリング用として設定されていないインターフェイスがある場合は（たとえば、デフォルト コンフィギュレーションの管理 1/1 インターフェイス）、データノードとしてクラスタに参加させることができます（現在の選定で制御ノードになる可能性はありません）。
- 稼働中のクラスタにノードを追加すると、一時的に、限定的なパケット/接続ドロップが発生することがありますが、これは想定内の動作です。

手順

ステップ 1 制御ノードに設定したものと同一クラスタ制御リンクインターフェイスを設定します。

例：

```
ciscoasa(config)# interface ethernet 1/6
ciscoasa(config-if)# channel-group 1 mode on
ciscoasa(config-if)# no shutdown
ciscoasa(config)# interface ethernet 1/7
ciscoasa(config-if)# channel-group 1 mode on
ciscoasa(config-if)# no shutdown
```

ステップ 2 制御ノードに設定したものと同一 MTU を指定します。

例：

```
ciscoasa(config)# mtu cluster 9198
```

ステップ 3 制御ノードに設定したものと同一クラスタ名を指定します。

例：

```
ciscoasa(config)# cluster group pod1
```

ステップ 4 クラスタのこのメンバに一意の文字列で名前を指定します。

local-unit *unit_name*

例：

```
ciscoasa(cfg-cluster)# local-unit node2
```

1 ～ 38 文字の ASCII 文字列を指定します。

各ノードには一意の名前が必要です。クラスタ内の他のノードと同じ名前を付けることはできません。

- ステップ 5** 制御ノードに設定したものと同じクラスタ制御リンクインターフェイスを指定しますが、ノードごとに同じネットワーク上の異なる IP アドレスを指定します。

cluster-interface *interface_id ip ip_address mask*

例 :

```
ciscoasa(cfg-cluster)# cluster-interface port-channel2 ip 192.168.1.2 255.255.255.0
INFO: Non-cluster interface config is cleared on Port-Channel2
```

IP アドレスには IPv4 アドレスを指定します。IPv6 は、このインターフェイスではサポートされません。このインターフェイスには、**nameif** を設定することはできません。

- ステップ 6** サイト間クラスタリングを使用している場合、このノードのサイト ID を設定し、サイト固有の MAC アドレスが使用されるようにします。

site-id *number*

例 :

```
ciscoasa(cfg-cluster)# site-id 1
```

number は 1 ~ 8 です。

- ステップ 7** 制御ノードの選定に対するこのノードのプライオリティを設定します。通常は、制御ノードより高い値にします。

priority *priority_number*

例 :

```
ciscoasa(cfg-cluster)# priority 2
```

プライオリティを 1 ~ 100 に設定します。1 が最高のプライオリティです。

- ステップ 8** 制御ノードに設定したものと同一認証キーを設定します。

例 :

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

- ステップ 9** クラスタリングをイネーブルにします。

enable as-slave

enable as-slave コマンドを使用することによって、設定に関するすべての非互換性（主にまだクラスタリング用に設定されていないインターフェイスの存在）を回避できます。このコマンドを実行すると、クラスタに参加させるデータノードが現在の選定において制御ノードとなる可能性をなくすことができます。データノードのコンフィギュレーションは、制御ノードから同期されたコンフィギュレーションによって上書きされます。

クラスタリングをディセーブルにするには、**no enable** コマンドを入力します。

(注) クラスタリングをディセーブルにした場合は、すべてのデータインターフェイスがシャットダウンされ、管理インターフェイスだけがアクティブになります。

例

次の例には、データノード **node2** の設定が含まれています。

```
interface ethernet 1/6

channel-group 1 mode on
no shutdown

interface ethernet 1/7

channel-group 1 mode on
no shutdown

cluster group pod1

local-unit node2
cluster-interface port-channel1 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

クラスタリング動作のカスタマイズ

クラスタリングヘルスモニタリング、TCP接続複製の遅延、フローのモビリティ、他の最適化をカスタマイズできます。

制御ノードで次の手順を実行します。

ASA クラスタの基本パラメータの設定

制御ノード上のクラスタ設定をカスタマイズできます。

始める前に

- マルチコンテキストモードでは、制御ノード上のシステム実行スペースで次の手順を実行します。コンテキストからシステム実行スペースに切り替えるには、**changeto system** コマンドを入力します。

手順

ステップ 1 クラスタの設定モードを開始します。

cluster group *name*

ステップ 2 (任意) データノードから制御ノードへのコンソール複製を有効にします。

console-replicate

この機能はデフォルトで無効に設定されています。ASA は、特定の重大イベントが発生したときに、メッセージを直接コンソールに出力します。コンソール複製を有効にすると、データノードから制御ノードにコンソールメッセージが送信されるので、モニタする必要があるのはクラスタのコンソールポート 1 つだけです。

ステップ 3 クラスタリング イベントの最小トレース レベルを設定します。

trace-level *level*

必要に応じて最小レベルを設定します。

- **critical** : クリティカル イベント (重大度 = 1)
- **warning** : 警告 (重大度 = 2)
- **informational** : 情報イベント (重大度 = 3)
- **debug** : デバッグ イベント (重大度 = 4)

のヘルス モニタリングおよび自動再結合の設定

この手順では、ノードとインターフェイスのヘルスモニタリングを設定します。

たとえば、管理インターフェイスなど、必須以外のインターフェイスのヘルスモニタリングをディisableにすることができます。任意のポート チャネル ID、冗長 ID、単一の物理インターフェイス ID、をモニターできます。ヘルス モニタリングは VLAN サブインターフェイス、または VNI や BVI などの仮想インターフェイスでは実行されません。クラスタ制御リンクのモニタリングは設定できません。このリンクは常にモニターされています。

手順

ステップ 1 クラスタの設定モードを開始します。

cluster group *name*

例 :

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)#
```

ステップ 2 クラスタノードのヘルスチェック機能をカスタマイズします。

health-check [**holdtime** *timeout*] [**vss-enabled**]

ノードのヘルスを確認するため、ASA のクラスタノードはクラスタ制御リンクで他のノードにハートビート メッセージを送信します。ノードが保留時間内にピアノードからハートビートメッセージを受信しない場合、そのピアノードは応答不能またはデッド状態と見なされます。

- **holdtime timeout** : ノードのハートビートステータスメッセージの時間間隔を指定します。指定できる範囲は .3 ~ 45 秒で、デフォルトは 3 秒です。
- **vss-enabled** : クラスタ制御リンクのすべての EtherChannel インターフェイスでハートビートメッセージをフラグディングして、少なくとも 1 台のスイッチがそれを受信できるようにします。EtherChannel としてクラスタ制御リンクを設定し（推奨）、VSS、vPC、StackWise、または vPC StackWise Virtual ペアに接続している場合、**vss-enabled** オプションをイネーブルにする必要がある場合があります。一部のスイッチでは、冗長システムの 1 つのノードがシャットダウンまたは起動すると、そのスイッチに接続された EtherChannel メンバーインターフェイスが ASA に対してアップ状態であるように見えますが、これらのインターフェイスはスイッチ側のトラフィックを通していません。ASA holdtime timeout を低い値（0.8 秒など）に設定した場合、ASA が誤ってクラスタから削除される可能性があります。ASA はキープアライブメッセージをこれらのいずれかの EtherChannel インターフェイスに送信します。

何らかのトポロジ変更（たとえばデータインターフェイスの追加/削除、ASA、またはスイッチ上のインターフェイスの有効化/無効化、VSS、vPC、StackWise、または StackWise Virtual を形成するスイッチの追加）を行うときには、ヘルスチェック機能を無効にし、無効化したインターフェイスのモニタリングも無効にしてください（**no health-check monitor-interface**）。トポロジの変更が完了して、設定の変更がすべてのノードに同期されたら、ヘルスチェック機能を再度有効にできます。

例 :

```
ciscoasa(cfg-cluster)# health-check holdtime 5
```

ステップ 3 インターフェイスでインターフェイスヘルスチェックを無効化します。

no health-check monitor-interface interface_id

インターフェイスのヘルスチェックはリンク障害をモニターします。特定の論理インターフェイスのすべての物理ポートが、特定のノード上では障害が発生したが、別のノード上の同じ論理インターフェイスでアクティブポートがある場合、そのノードはクラスタから削除されます。ASA がメンバーをクラスタから削除するまでの時間は、インターフェイスのタイプと、そのノードが確立済みメンバーであるか、またはクラスタに参加しようとしているかによって異なります。デフォルトでは、ヘルスチェックはすべてのインターフェイスでイネーブルになっています。このコマンドの **no** 形式を使用してディセーブル（無効）にすることができます。たとえば、管理インターフェイスなど、必須以外のインターフェイスのヘルスモニタリングをディセーブルにすることができます。

- **interface_id** : ポートチャネル ID と冗長 ID、または単一の物理インターフェイス ID のモニタリングを無効にします。ヘルスモニタリングは VLAN サブインターフェイス、または VNI や BVI などの仮想インターフェイスでは実行されません。クラスタ制御リンクのモニタリングは設定できません。このリンクは常にモニターされています。

何らかのトポロジ変更（たとえばデータ インターフェイスの追加/削除、ASA、またはスイッチ上のインターフェイスの有効化/無効化、VSS、vPC、StackWise、または StackWise Virtual を形成するスイッチの追加）を行うときには、ヘルスチェック機能を無効にし（**no health-check**）、無効化したインターフェイスのモニタリングも無効にしてください。トポロジの変更が完了して、設定の変更がすべてのノードに同期されたら、ヘルスチェック機能を再度有効にできます。

例：

```
ciscoasa(cfg-cluster)# no health-check monitor-interface management1/1
```

ステップ 4 ヘルス チェック失敗後の自動再結合クラスタ設定をカスタマイズします。

health-check {data-interface | cluster-interface | system} auto-rejoin [unlimited | auto_rejoin_max] auto_rejoin_interval auto_rejoin_interval_variation

- **system**：内部エラー時の自動再結合の設定を行います。内部の障害には、アプリケーション同期のタイムアウト、矛盾したアプリケーション ステータスなどがあります。
- **unlimited**：（**cluster-interface** のデフォルト）再結合の試行回数を制限しません。
- **auto-rejoin-max**：再結合の試行回数を 0 ～ 65535 の範囲の値に設定します。0 は自動再結合を無効化します。**data-interface** と **system** のデフォルトは 3 です。
- **auto_rejoin_interval**：再結合試行の間隔を 2 ～ 60 の範囲の分単位で定義します。デフォルト値は 5 分です。クラスタへの再参加をノードが試行する最大合計時間は、最後の障害発生時から 14400 分（10 日）に制限されます。
- **auto_rejoin_interval_variation**：間隔を増加させるかどうかを定義します。1 ～ 3 の範囲で値を設定します（**1**：変更なし、**2**：直前の間隔の 2 倍、**3**：直前の間隔の 3 倍）。たとえば、間隔を 5 分に設定し、変分を 2 に設定した場合は、最初の試行が 5 分後、2 回目の試行が 10 分後（2 x 5）、3 階目の試行が 20 分後（2 x 10）となります。デフォルト値は、クラスタ インターフェイスの場合は **1**、データ インターフェイスおよびシステムの場合は **2** です。

例：

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 10 3 3
```

ステップ 5 ASA がインターフェイスを障害が発生していると思なし、クラスタからノードが削除されるまでのデバウンス時間を設定します。

health-check monitor-interface debounce-time ms

例：

```
ciscoasa(cfg-cluster)# health-check monitor-interface debounce-time 300
```

デバウンス時間は 300 ～ 9000 ms の範囲の値を設定します。デフォルトは 500 ms です。値を小さくすると、インターフェイスの障害をより迅速に検出できます。デバウンス時間を短くす

ると、誤検出の可能性が高くなることに注意してください。インターフェイスのステータス更新が発生すると、ASA はインターフェイスを障害としてマークし、クラスタからノードを削除するまで指定されたミリ秒数待機します。EtherChannel がダウン状態からアップ状態に移行する場合（スイッチがリロードされた、スイッチで EtherChannel が有効になったなど）、デバウンス時間がより長くなり、ポートのバンドルにおいて別のクラスタノードの方が高速なため、クラスタノードでインターフェイスの障害が表示されることを妨げることがあります。

ステップ 6 （任意） トラフィック負荷のモニタリングを設定します。

load-monitor [frequency seconds] [intervals intervals]

- **seconds**: モニタリングメッセージ間の時間を、10～360 秒の範囲で設定します。 **frequency** デフォルトは 20 秒です。
- 間隔（*interval*）： ASA がデータを保持する間隔の数を 1～60 の範囲で設定します。 **intervals** デフォルトは 30 です。

クラスタメンバのトラフィック負荷をモニターできます。対象には、合計接続数、CPU とメモリの使用率、バッファドロップなどが含まれます。負荷が高すぎる場合、残りのノードが負荷を処理できる場合は、ノードのクラスタリングを手動で無効にするか、外部スイッチのロードバランシングを調整するかを選択できます。この機能は、デフォルトでイネーブルにされています。たとえば、各シャーシに 3 つのセキュリティモジュールが搭載された Firepower 9300 のシャーシ間クラスタリングの場合、シャーシ内の 2 つのセキュリティモジュールがクラスタを離れると、そのシャーシに対する同じ量のトラフィックが残りのモジュールに送信され、過負荷になる可能性があります。トラフィックの負荷を定期的にモニターできます。負荷が高すぎる場合は、ノードでクラスタリングを手動で無効にすることを選択できます。

トラフィック負荷を表示するには、**show cluster info load-monitor** コマンドを使用します。

例：

```
ciscoasa(cfg-cluster)# load-monitor frequency 50 intervals 25
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0   B
1   A_1
Information from all units with 50 second interval:
Unit  Connections  Buffer Drops  Memory Used  CPU Used
Average from last 1 interval:
0     0             0             14           25
1     0             0             16           20
Average from last 25 interval:
0     0             0             12           28
1     0             0             13           27
```

例

次の例では、ヘルスチェック保留時間を .3 秒に設定し、VSS を有効にし、管理に使用されるイーサネット 1/2 インターフェイスのモニタリングを無効にし、データインターフェイスの自動再結合の試行回数を 2 分から開始して前回の間隔の 3 倍増加させる計

4 回に設定し、クラスタ制御リンクの自動再結合の試行回数を 2 分おきの計 6 回に設定しています。

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)# health-check holdtime .3 vss-enabled
ciscoasa(cfg-cluster)# no health-check monitor-interface ethernet1/2
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 4 2 3
ciscoasa(cfg-cluster)# health-check cluster-interface auto-rejoin 6 2 1
```

接続の再分散およびクラスタ TCP 複製の遅延の設定

接続の再分散を設定できます。詳細については、[新しい TCP 接続のクラスタ全体での再分散 \(110 ページ\)](#) を参照してください。

TCP 接続のクラスタ複製の遅延を有効化して、ディレクタ/バックアップ フロー作成の遅延による存続期間が短いフローに関連する「不要な作業」を排除できます。ディレクタ/バックアップ フローが作成される前にノードが失敗する場合は、それらのフローを回復することはできません。同様に、フローを作成する前にトラフィックが別のノードに再調整される場合、流れを回復することはできません。TCP のランダム化を無効化するトラフィックの TCP の複製の遅延を有効化しないようにする必要があります。

手順

ステップ 1 TCP 接続のクラスタ複製の遅延を有効化します。

```
cluster replication delay seconds { http | match tcp { host ip_address | ip_address mask | any | any4 | any6 } [{ eq | lt | gt } port] { host ip_address | ip_address mask | any | any4 | any6 } [{ eq | lt | gt } port] }
```

例 :

```
ciscoasa(config)# cluster replication delay 15 match tcp any any eq ftp
ciscoasa(config)# cluster replication delay 15 http
```

1 ~ 15 の範囲で秒数を設定します。**http** 遅延はデフォルトで 5 秒間有効になります。

マルチ コンテキスト モードで、コンテキスト内でこの設定を行います。

ステップ 2 クラスタの設定モードを開始します。

```
cluster group name
```

ステップ 3 (オプション) TCP トラフィックの接続の再分散を有効化します。

```
conn-rebalance [frequency seconds]
```

例 :

```
ciscoasa(cfg-cluster)# conn-rebalance frequency 60
```

このコマンドは、デフォルトでディセーブルになっています。有効化されている場合は、ASA は負荷情報を定期的に交換し、新しい接続の負荷を高負荷のデバイスから低負荷のデバイスに移動します。負荷情報を交換する間隔を、1 ~ 360 秒の範囲内で指定します。デフォルトは 5 秒です。

サイト間トポロジに対しては接続の再分散を設定しないでください。異なるサイトのクラスタメンバには接続を再分散できません。

サイト間機能の設定

サイト間クラスタリングの場合、冗長性と安定性を高めるために、設定をカスタマイズできます。

ディレクタ ローカリゼーションの有効化

データセンターのサイト間クラスタリングのパフォーマンスを向上させ、ラウンドトリップ時間を短縮するために、ディレクターローカリゼーションをイネーブルにすることができます。通常、新しい接続は特定のサイト内のクラスタメンバーによってロードバランスされ、所有されています。しかし、ASA は任意のサイトのメンバーにディレクタロールを割り当てます。ディレクタローカリゼーションにより、所有者と同じサイトのローカルディレクタ、どのサイトにも存在可能なグローバルディレクタという追加のディレクタロールが有効になります。所有者とディレクタが同一サイトに存在すると、パフォーマンスが向上します。また、元の所有者が失敗した場合、ローカルなディレクタは同じサイトで新しい接続の所有者を選択します。グローバルなディレクタは、クラスタメンバーが別のサイトで所有される接続のパケットを受信する場合に使用されます。

始める前に

- ブートストラップ設定でクラスタメンバーのサイト ID を設定します。
- 次のトラフィックタイプは、ローカリゼーションをサポートしていません：NAT および PAT トラフィック、SCTP 検査されたトラフィック、フラグメンテーション所有クエリ。

手順

ステップ 1 クラスタの設定モードを開始します。

cluster group *name*

例：

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

ステップ 2 ディレクタローカリゼーションをイネーブルにします。

director-localization

サイト冗長性の有効化

サイトの障害からフローを保護するために、サイトの冗長性を有効にできます。接続バックアップオーナーがオーナーと同じサイトにある場合は、サイトの障害からフローを保護するために、追加のバックアップオーナーが別のサイトから選択されます。

始める前に

- ブートストラップ設定でクラスタメンバーのサイト ID を設定します。

手順

ステップ 1 クラスタの設定モードを開始します。

cluster group *name*

例 :

```
ciscoasa(config)# cluster group cluster1  
ciscoasa(cfg-cluster)#
```

ステップ 2 サイトの冗長性を有効にします。

site-redundancy

サイトごとの Gratuitous ARP の設定

ASA では、Gratuitous ARP (GARP) パケットを生成してスイッチングインフラストラクチャを常に最新の状態に保つようになりました。各サイトの優先順位値が最も高いメンバーによって、グローバル MAC/IP アドレスの GARP トラフィックが定期的に生成されます。

クラスタから送信されたサイトごとの MAC および IP アドレスとパケットがサイト固有の MAC アドレスおよび IP アドレスを使用するのに対し、クラスタで受信したパケットは、グローバル MAC アドレスおよび IP アドレスを使用します。トラフィックがグローバル MAC アドレスから定期的に生成されない場合、グローバル MAC アドレスのスイッチで MAC アドレスのタイムアウトが発生する可能性があります。タイムアウト後にグローバル MAC アドレスへのトラフィックがスイッチングインフラストラクチャ全体にわたりフラッドされ、これによりパフォーマンスおよびセキュリティ上の問題が発生することがあります。

各スパンド EtherChannel のユニットおよびサイト MAC アドレスごとにサイト ID を設定すると、GARP がデフォルトで有効になります。GARP 間隔をカスタマイズするか、または GARP を無効にすることができます。

始める前に

- ブートストラップ設定でクラスタ メンバーのサイト ID を設定します。
- 制御ユニット設定では、スパンド EtherChannel のサイトごとの MAC アドレスを設定します。

手順

ステップ 1 クラスタの設定モードを開始します。

cluster group name

例 :

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

ステップ 2 GARP 間隔をカスタマイズします。

site-periodic-garp interval seconds

- *seconds* : GARP 生成の間隔を 1 ~ 1000000 秒間の秒単位で設定します。デフォルトは 290 秒です。

GARP を無効にするには、**no site-periodic-garp interval** を入力します。

クラスタ フロー モビリティの設定

LISP のトラフィックを検査して、サーバーがサイト間を移動する時にフロー モビリティを有効にできます。

LISP インспекションについて

LISP トラフィックを検査することで、サイト間のフローのモビリティを有効にできます。

LISP について

VMware vMotion などのデータセンター仮想マシンのモビリティによって、サーバはクライアントへの接続を維持すると同時に、データセンター間を移動できます。このようなデータセンター サーバ モビリティをサポートするには、サーバの移動時にサーバへの入力ルートが更新できる必要があります。Cisco Locator/ID Separation Protocol (LISP) のアーキテクチャは、デバイス ID、つまりエンドポイント ID (EID) をその場所、つまりルーティング ロケータ (RLOC) から 2 つの異なるナンバリング スペースに分離し、サーバの移行をクライアントに対して透過的にします。たとえば、サーバが新しい場所に移動し、クライアントがサーバにトラフィックを送信すると、ルータは新しい場所にトラフィックをリダイレクトします。

LISP では、LISP の出力トンネルルータ (ETR)、入力トンネルルータ (ITR)、ファーストホップルータ、マップリゾルバ (MR)、およびマップサーバ (MS) などのある一定のロー

ルにおいてルータとサーバが必要です。サーバが別のルータに接続されていることをサーバのファーストホップルータが感知すると、そのルータは他のすべてのルータとデータベースを更新し、クライアントに接続されている ITR がトラフィックを代行受信してカプセル化し、新しいサーバの場所に送信できるようにします。

ASA LISP のサポート

ASA は LISP 自体を実行しませんが、場所の変更に関する LISP トラフィックを検査し、シームレスなクラスタリング操作のためにこの情報を使用できます。LISP の統合を行わない場合、サーバが新しいサイトに移動すると、トラフィックは元のフローオーナーの代わりに、新しいサイトで ASA クラスタメンバーになります。新しい ASA が古いサイトの ASA にトラフィックを転送した後、古い ASA は、サーバに到達するためにトラフィックを新しいサイトに送り返す必要があります。このトラフィックフローは最適ではなく、「トロンボーン」または「ヘアピニング」と呼ばれます。

LISP 統合により、ASA クラスタメンバーは、最初のホップルータと ETR または ITR 間でやり取りされる LISP トラフィックを検査し、フローの所有者を新しいサイトに変更できます。

LISP のガイドライン

- ASA クラスタメンバーは、サイトのファーストホップルータと ITR または ETR の間に存在する必要があります。ASA クラスタ自体を拡張セグメントのファーストホップルータにすることはできません。
- 完全分散されたフローのみがサポートされます。一元化されたフロー、半分散されたフロー、または個々のノードに属しているフローは新しいオーナーには移動されません。半分散されたフローには SIP などのアプリケーションが含まれており、親フローとそのすべての子フローが同じ ASA によって所有されます。
- クラスタはレイヤ 3 および 4 のフロー状態を移動させるだけです。一部のアプリケーションデータが失われる可能性があります。
- 短時間のフローまたはビジネスに不可欠でないフローの場合、オーナーの移動は有用でない可能性があります。インスペクションポリシーを設定するときに、この機能でサポートされるトラフィックのタイプを制御できます。また、フローモビリティを不可欠なトラフィックに制限する必要があります。

ASA LISP の実装

この機能には、複数の相互に関係する設定が含まれています（それらについてはすべてこの章で説明します）。

1. （任意）ホストまたはサーバ IP アドレスに基づく検査対象 EID の制限：ファーストホップルータは、ASA クラスタが関与していないホストまたはネットワークに EID 通知メッセージを送信する場合があります。このため、クラスタに関連するサーバまたはネットワークのみに EID を制限できます。たとえば、クラスタが 2 つのサイトのみに関与しているが、LISP が 3 つのサイトで実行されている場合は、クラスタに関与している 2 つのサイトに対してのみ EID を含める必要があります。

2. LISP トラフィック インспекション：ASA は、ファーストホップルータと ITR または ETR の間で送信される EID 通知メッセージにおいて、UDP ポート 4342 上の LISP トラフィックを検査します。ASA は、EID とサイト ID を関連付ける EID テーブルを保持します。たとえば、最初のホップルータの送信元 IP アドレスと ITR または ETR の宛先アドレスをもつ LISP トラフィックを検査する必要があります。LISP トラフィックにはディレクタが割り当てられておらず、LISP トラフィック自体はクラスタ状態の共有に参加しないことに注意してください。
3. 指定されたトラフィックでのフロー モビリティを有効にするサービス ポリシー：ビジネスクリティカルなトラフィックでフロー モビリティを有効にする必要があります。たとえば、フロー モビリティを、HTTPS トラフィックのみに制限したり、特定のサーバとの間でやり取りされるトラフィックのみに制限したりできます。
4. サイト ID：ASA は、各クラスタノードのサイト ID を使用して新しいオーナーを特定します。
5. フロー モビリティを有効にするクラスタレベルの設定：クラスタ レベルでもフロー モビリティを有効にする必要があります。このオン/オフの切り替えを使用することで、特定のクラスのトラフィックまたはアプリケーションに対してフロー モビリティを簡単に有効または無効にできます。

LISP インспекションの設定

LISP のトラフィックを検査して、サーバーがサイト間を移動する時にフロー モビリティを有効にできます。

始める前に

- [制御ノードのブートストラップの設定 \(32 ページ\)](#) および [データノードのブートストラップの設定 \(37 ページ\)](#) に従って、各クラスタユニットをサイト ID に割り当てます。
- LISP のトラフィックはデフォルト インспекション トラフィック クラスに含まれないため、この手順の一部として LISP のトラフィック用に別のクラスを設定する必要があります。

手順

ステップ 1 (任意) LISP インспекション マップを設定して、IP アドレスに基づいて検査済みの EID を制限し、LISP の事前共有キーを設定します。

- a) 拡張 ACL を作成します。宛先 IP アドレスのみが EID 組み込みアドレスと照合されます。

```
access list eid_acl_name extended permit ip source_address mask destination_address mask
```

IPv4 ACL および IPv6 ACL のどちらにも対応しています。厳密な **access-list extended** の構文については、コマンド リファレンスを参照してください。

- b) LISP インспекション マップを作成し、パラメータ モードに移行します。

policy-map type inspect lisp inspect_map_name**parameters**

- c) 作成した ACL を識別して、許可された EID を定義します。

allowed-eid access-list eid_acl_name

ファースト ホップ ルータまたは ITR/ETR は、ASA クラスタが関与していないホストまたはネットワークに EID 通知メッセージを送信することがあります。このため、クラスタに関連するサーバーまたはネットワークのみに EID を制限できます。たとえば、クラスタが 2 つのサイトのみに関与しているが、LISP が 3 つのサイトで実行されている場合は、クラスタに関与している 2 つのサイトに対してのみ EID を含める必要があります。

- d) 必要に応じて、事前共有キーを入力します。

validate-key key

例 :

```
ciscoasa(config)# access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0
255.255.255.0
ciscoasa(config)# policy-map type inspect lisp LISP_EID_INSPECT
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# allowed-eid access-list TRACKED_EID_LISP
ciscoasa(config-pmap-p)# validate-key MadMaxShinyandChrome
```

ステップ 2 ファースト ホップ ルータとポート 4342 の ITR または ETR の間の UDP トラフィック の LISP インспекションの設定。

- a) 拡張 ACL を設定して LISP のトラフィックを特定します。

access list inspect_acl_name extended permit udp source_address mask destination_address mask eq 4342

UDP ポート 4342 を指定する必要があります。IPv4 ACL および IPv6 ACL のどちらにも対応しています。厳密な **access-list extended** の構文については、コマンドリファレンスを参照してください。

- b) ACL のクラス マップを作成します。

class-map inspect_class_name**match access-list inspect_acl_name**

- c) ポリシー マップ、クラス マップを指定し、オプションの LISP インспекション マップを使用してインспекションを有効化し、サービスポリシーをインターフェイスに適用します（新規であれば）。

policy-map policy_map_name**class inspect_class_name****inspect lisp [inspect_map_name]****service-policy policy_map_name {global | interface ifc_name}**

既存のサービス ポリシーある場合は、既存のポリシー マップ名を指定します。デフォルトで、ASAには**global_policy**と呼ばれるグローバルポリシーが含まれているため、グローバルポリシーの名前を指定します。ポリシーをグローバルに適用しない場合は、インターフェイスごとに1つのサービスポリシーを作成することもできます。LISP インспекションは、双方向にトラフィックに適用するため、送信元と宛先の両方のインターフェイスにサービスポリシーを適用する必要はありません。トラフィックが両方向のクラス マップに一致する場合、ポリシーマップを適用するインターフェイスに入るまたは存在するトラフィックのすべてが影響を受けます。

例：

```
ciscoasa(config)# access-list LISP_ACL extended permit udp host 192.168.50.89 host
192.168.10.8 eq 4342
ciscoasa(config)# class-map LISP_CLASS
ciscoasa(config-cmap)# match access-list LISP_ACL
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class LISP_CLASS
ciscoasa(config-pmap-c)# inspect lisp LISP_EID_INSPECT
ciscoasa(config)# service-policy INSIDE_POLICY interface inside
```

ASAは、ファーストホップルータと ITR または ETR の間で送信される EID 通知メッセージの LISP トラフィックを検査します。ASA は、EID とサイト ID を関連付ける EID テーブルを保持します。

ステップ 3 トラフィック クラスのフロー モビリティを有効化します。

- a) 拡張 ACL を設定して、サーバーがサイトを変更するときに、最適なサイトに再割り当てするビジネスクリティカルなトラフィックを特定します。

access list *flow_acl_name* **extended permit udp** *source_address mask destination_address mask eq port*

IPv4 ACL および IPv6 ACL のどちらにも対応しています。厳密な **access-list extended** の構文については、コマンドリファレンスを参照してください。フロー モビリティは、ビジネスクリティカルなトラフィックに対してイネーブルにする必要があります。たとえば、フロー モビリティを HTTPS トラフィックのみ、または特定のサーバーへのトラフィックのみに制限できます。

- b) ACL のクラス マップを作成します。

class-map *flow_map_name*
match access-list *flow_acl_name*

- c) LISP インспекションを有効化した同じポリシー マップ、フロー クラス マップを指定して、フロー モビリティを有効にします。

policy-map *policy_map_name*
class *flow_map_name*
cluster flow-mobility **lisp**

例：

```

ciscoasa(config)# access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0
255.255.255.0 eq https
ciscoasa(config)# class-map IMPORTANT-FLOWS-MAP
ciscoasa(config)# match access-list IMPORTANT-FLOWS
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class IMPORTANT-FLOWS-MAP
ciscoasa(config-pmap-c)# cluster flow-mobility lisp

```

ステップ 4 クラスタ グループ コンフィギュレーション モードに移行し、クラスタのフローのモビリティを有効化します。

cluster group name

flow-mobility lisp

このオン/オフの切り替えにより、フロー モビリティの有効化や無効化を簡単に行えます。

例

次に例を示します。

- EID を 10.10.10.0/24 ネットワーク上の EID に制限します。
- 192.168.50.89 (内部) にある LISP ルータと 192.168.10.8 (別の ASA インターフェイス上) にある ITR または ETR ルータの間の LISP トラフィック (UDP 4342) を検査します。
- HTTPS を使用して 10.10.10.0/24 のサーバーに送信されるすべての内部トラフィックに対してフロー モビリティを有効化します。
- クラスタに対してフロー モビリティをイネーブルにします。

```

access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
policy-map type inspect lisp LISP_EID_INSPECT
  parameters
    allowed-eid access-list TRACKED_EID_LISP
    validate-key MadMaxShinyandChrome
!
access-list LISP_ACL extended permit udp host 192.168.50.89 host 192.168.10.8 eq 4342
class-map LISP_CLASS
  match access-list LISP_ACL
policy-map INSIDE_POLICY
  class LISP_CLASS
    inspect lisp LISP_EID_INSPECT
service-policy INSIDE_POLICY interface inside
!
access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0 eq https
class-map IMPORTANT-FLOWS-MAP
  match access-list IMPORTANT-FLOWS
policy-map INSIDE_POLICY
  class IMPORTANT-FLOWS-MAP
    cluster flow-mobility lisp
!
cluster group cluster1

```

```
flow-mobility lisp
```

クラスタノードの管理

クラスタを導入した後は、コンフィギュレーションを変更し、クラスタノードを管理できます。

非アクティブノードになる

クラスタの非アクティブなメンバーになるには、クラスタリングコンフィギュレーションは変更せずに、そのノード上でクラスタリングをディセーブルにします。



- (注) ASAが(手動で、またはヘルスチェックエラーにより)非アクティブになると、すべてのデータインターフェイスがシャットダウンされます。管理専用インターフェイスのみがトラフィックを送受信できます。トラフィックフローを再開させるには、クラスタリングを再びイネーブルにします。または、そのノードをクラスタから完全に削除します。管理インターフェイスは、そのノードがクラスタ IP プールから受け取った IP アドレスを使用して引き続き稼働状態となります。ただし、リロードしてもノードがクラスタ内でまだアクティブではない場合(クラスタリングが無効な状態で設定を保存した場合など)、管理インターフェイスは無効になります。それ以降のコンフィギュレーション作業には、コンソールポートを使用する必要があります。

始める前に

- コンソールポートを使用する必要があります。クラスタリングのイネーブルまたはディセーブルを、リモート CLI 接続から行うことはできません。
- マルチ コンテキスト モードの場合は、この手順をシステム実行スペースで実行します。まだシステム コンフィギュレーション モードに入っていない場合は、**changeto system** コマンドを入力します。

手順

ステップ 1 クラスタの設定モードを開始します。

```
cluster group name
```

例 :

```
ciscoasa(config)# cluster group pod1
```

ステップ 2 クラスタリングをディセーブルにします。

no enable

このノードが制御ノードであった場合は、新しい制御ノードの選定が実行され、別のメンバーが制御ノードになります。

クラスタ コンフィギュレーションは維持されるので、後でクラスタリングを再度イネーブルにできます。

ノードの非アクティブ化

ログインしているノード以外のメンバを非アクティブにするには、次のステップを実行します。



- (注) ASA が非アクティブになると、すべてのデータインターフェイスがシャットダウンされます。管理専用インターフェイスのみがトラフィックを送受信できます。トラフィックフローを再開するには、クラスタリングを再度有効にします。管理インターフェイスは、そのノードがクラスタ IP プールから受け取った IP アドレスを使用して引き続き稼働状態となります。ただし、リロードしてもノードがクラスタ内でまだアクティブではない場合（クラスタリングが無効な状態で設定を保存した場合など）、管理インターフェイスは無効になります。それ以降のコンフィギュレーション作業には、コンソールポートを使用する必要があります。

始める前に

マルチ コンテキスト モードの場合は、この手順をシステム実行スペースで実行します。まだシステム コンフィギュレーション モードに入っていない場合は、**changeto system** コマンドを入力します。

手順

クラスタからノードを削除します。

cluster remove unit *node_name*

ブートストラップコンフィギュレーションは変更されず、制御ノードから最後に同期されたコンフィギュレーションもそのままであるので、コンフィギュレーションを失わずに後でそのノードを再度追加できます。制御ノードを削除するためにデータノードでこのコマンドを入力した場合は、新しい制御ノードが選定されます。

メンバ名を一覧表示するには、**cluster remove unit ?** と入力するか、**show cluster info** コマンドを入力します。

例 :

```
ciscoasa(config)# cluster remove unit ?
```

```
Current active units in the cluster:
asa2

ciscoasa(config)# cluster remove unit asa2
WARNING: Clustering will be disabled on unit asa2. To bring it back
to the cluster please logon to that unit and re-enable clustering
```

クラスタへの再参加

ノードがクラスタから削除された場合（たとえば、障害が発生したインターフェイスの場合、またはメンバーを手動で非アクティブにした場合）は、クラスタに手動で再参加する必要があります。

始める前に

- クラスタリングを再イネーブルにするには、コンソール ポートを使用する必要があります。他のインターフェイスはシャットダウンされます。
- マルチ コンテキスト モードの場合は、この手順をシステム実行スペースで実行します。まだシステム コンフィギュレーション モードに入っていない場合は、**changeto system** コマンドを入力します。
- クラスタへの再参加を試行する前に、障害が解決されていることを確認します。

手順

ステップ 1 コンソールで、クラスタ コンフィギュレーション モードを開始します。

```
cluster group name
```

例 :

```
ciscoasa(config)# cluster group pod1
```

ステップ 2 クラスタリングをイネーブルにします。

```
enable
```

クラスタからの脱退

クラスタから完全に脱退するには、クラスタ ブートストラップ コンフィギュレーション全体を削除する必要があります。各ノードの現在のコンフィギュレーションは（アクティブユニットから同期されて）同じであるため、クラスタから脱退すると、クラスタリング前のコンフィ

ギューレションをバックアップから復元するか、IPアドレスの競合を避けるためコンフィギュレーションを消去して初めからやり直すことも必要になります。

始める前に

コンソールポートを使用する必要があります。クラスタのコンフィギュレーションを削除すると、管理インターフェイスとクラスタ制御リンクを含むすべてのインターフェイスがシャットダウンされます。さらに、クラスタリングのイネーブルまたはディセーブルを、リモートCLI接続から行うことはできません。

手順

ステップ 1 データノードの場合、クラスタリングを次のように無効化します。

cluster group cluster_name no enable

例 :

```
ciscoasa(config)# cluster group cluster1  
ciscoasa(cfg-cluster)# no enable
```

クラスタリングがデータノード上でイネーブルになっている間は、コンフィギュレーション変更を行うことはできません。

ステップ 2 クラスタ コンフィギュレーションをクリアします。

clear configure cluster

ASAは、管理インターフェイスとクラスタ制御リンクを含むすべてのインターフェイスをシャットダウンします。

ステップ 3 クラスタ インターフェイス モードをディセーブルにします。

no cluster interface-mode

モードはコンフィギュレーションには保存されないため、手動でリセットする必要があります。

ステップ 4 バックアップコンフィギュレーションがある場合、実行コンフィギュレーションにバックアップコンフィギュレーションをコピーします。

copy backup_cfg running-config

例 :

```
ciscoasa(config)# copy backup_cluster.cfg running-config  
  
Source filename [backup_cluster.cfg]?  
  
Destination filename [running-config]?  
ciscoasa(config)#
```

ステップ5 コンフィギュレーションをスタートアップに保存します。

write memory

ステップ6 バックアップ コンフィギュレーションがない場合は、管理アクセスを再設定します。たとえば、インターフェイス IP アドレスを変更し、正しいホスト名を復元します。

制御ノードの変更



注意 制御ノードを変更する最良の方法は、制御ノードでクラスタリングを無効にし、新しい制御ユニットの選択を待ってから、クラスタリングを再度有効にする方法です。制御ノードにするノードを厳密に指定する必要がある場合は、この項の手順を使用します。ただし、中央集中型機能の場合は、この手順を使用して制御ノード変更を強制するとすべての接続がドロップされるので、新しい制御ノード上で接続を再確立する必要があります。

制御ノードを変更するには、次の手順を実行します。

始める前に

マルチ コンテキスト モードの場合は、この手順をシステム実行スペースで実行します。まだシステム コンフィギュレーション モードに入っていない場合は、**changeto system** コマンドを入力します。

手順

新しいノードを制御ノードとして設定します。

cluster master unit*node_name*

例：

```
ciscoasa(config)# cluster master unit asa2
```

メイン クラスタ IP アドレスへの再接続が必要になります。

メンバー名を表示するには、**cluster master unit ?**（現在のノードを除くすべての名前が表示される）と入力するか、**show cluster info** コマンドを入力します。

クラスタ全体でのコマンドの実行

コマンドをクラスタ内のすべてのノードに、または特定のノードに送信するには、次の手順を実行します。**show** コマンドをすべてのノードに送信すると、すべての出力が収集されて現在

のノードのコンソールに表示されます。その他のコマンド、たとえば **capture** や **copy** も、クラスタ全体での実行を活用できます。

手順

すべてのノードにコマンドを送信します。ノード名を指定した場合は、特定のノードに送信します。

cluster exec [unit node_name] コマンド

例：

```
ciscoasa# cluster exec show xlate
```

ノード名を表示するには、**cluster exec unit ?**（現在のノードを除くすべての名前が表示される）と入力するか、**show cluster info** コマンドを入力します。

例

同じキャプチャファイルをクラスタ内のすべてのノードから同時に TFTP サーバにコピーするには、制御ノードで次のコマンドを入力します。

```
ciscoasa# cluster exec copy /pcap capture: tftp://10.1.1.56/capture1.pcap
```

複数の PCAP ファイル（各ノードから 1 つずつ）が TFTP サーバにコピーされます。宛先のキャプチャファイル名には自動的にノード名が付加され、**capture1_asa1.pcap**、**capture1_asa2.pcap** などとなります。この例では、**asa1** と **asa2** はクラスタノード名です。

次の例では、**cluster exec show port-channel summary** コマンドの出力に、クラスタの各ノードの EtherChannel 情報が表示されています。

```
ciscoasa# cluster exec show port-channel summary
master(LOCAL):*****
Number of channel-groups in use: 2
Group Port-channel Protocol Span-cluster Ports
-----+-----+-----+-----+-----
1      Po1           LACP           Yes  Gi0/0 (P)
2      Po2           LACP           Yes  Gi0/1 (P)
slave:*****
Number of channel-groups in use: 2
Group Port-channel Protocol Span-cluster Ports
-----+-----+-----+-----+-----
1      Po1           LACP           Yes  Gi0/0 (P)
2      Po2           LACP           Yes  Gi0/1 (P)
```

ASA クラスタのモニタリング

クラスタの状態と接続をモニターおよびトラブルシューティングできます。

クラスタ ステータスのモニタリング

クラスタの状態のモニタリングについては、次のコマンドを参照してください。

- **show cluster info [health [details]]**

キーワードを指定しないで **show cluster info** コマンドを実行すると、クラスタ内のすべてのメンバのステータスが表示されます。

show cluster info health コマンドは、インターフェイス、ノードおよびクラスタ全体の現在の状態を表示します。**details** キーワードは、ハートビートメッセージの失敗数を表示します。

show cluster info コマンドについては次の出力を参照してください。

```
ciscoasa# show cluster info
Cluster stbu: On
  This is "C" in state SLAVE
    ID      : 0
    Site ID : 1
      Version : 9.4(1)
    Serial No.: P3000000025
    CCL IP    : 10.0.0.3
    CCL MAC   : 000b.fcf8.c192
    Last join : 17:08:59 UTC Sep 26 2011
    Last leave: N/A
  Other members in the cluster:
    Unit "D" in state SLAVE
      ID      : 1
      Site ID : 1
        Version : 9.4(1)
      Serial No.: P3000000001
      CCL IP    : 10.0.0.4
      CCL MAC   : 000b.fcf8.c162
      Last join : 19:13:11 UTC Sep 23 2011
      Last leave: N/A
    Unit "A" in state MASTER
      ID      : 2
      Site ID : 2
        Version : 9.4(1)
      Serial No.: JAB0815R0JY
      CCL IP    : 10.0.0.1
      CCL MAC   : 000f.f775.541e
      Last join : 19:13:20 UTC Sep 23 2011
      Last leave: N/A
    Unit "B" in state SLAVE
      ID      : 3
      Site ID : 2
        Version : 9.4(1)
      Serial No.: P3000000191
      CCL IP    : 10.0.0.2
      CCL MAC   : 000b.fcf8.c61e
      Last join : 19:13:50 UTC Sep 23 2011
```

Last leave: 19:13:36 UTC Sep 23 2011

• show cluster info auto-join

時間遅延後にクラスタノードがクラスタに自動的に再参加するかどうか、および障害状態（ライセンスの待機やシャーシのヘルスチェック障害など）がクリアされたかどうかを示します。ノードが永続的に無効になっている場合、またはノードがすでにクラスタ内にある場合、このコマンドでは出力が表示されません。

show cluster info auto-join コマンドについては次の出力を参照してください。

```
ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster in 253 seconds.
Quit reason: Received control message DISABLE

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Master has application down that slave has up.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Chassis-blade health check failed.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Service chain application became down.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Unit is kicked out from cluster because of Application health check failure.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license entitlement: ent1)

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license export control flag)
```

• show cluster info transport {asp | cp [detail]}

次のトランスポート関連の統計情報を表示します。

- **asp** : データプレーンのトランスポート統計情報。
- **cp** : コントロールプレーンのトランスポート統計情報。

detail キーワードを入力すると、クラスタで信頼性の高いトランスポートプロトコルの使用状況が表示され、バッファがコントロールプレーンでいっぱいになったときにパケットドロップの問題を特定できます。**show cluster info transport cp detail** コマンドについては次の出力を参照してください。

```
ciscoasa# show cluster info transport cp detail
Member ID to name mapping:
 0 - unit-1-1   2 - unit-4-1   3 - unit-2-1

Legend:
 U   - unreliable messages
 UE  - unreliable messages error
```

SN - sequence number
 ESN - expecting sequence number
 R - reliable messages
 RE - reliable messages error
 RDC - reliable message deliveries confirmed
 RA - reliable ack packets received
 RFR - reliable fast retransmits
 RTR - reliable timer-based retransmits
 RDP - reliable message dropped
 RDPR - reliable message drops reported
 RI - reliable message with old sequence number
 RO - reliable message with out of order sequence number
 ROW - reliable message with out of window sequence number
 ROB - out of order reliable messages buffered
 RAS - reliable ack packets sent

This unit as a sender

```

-----
      all      0      2      3
U    123301    3867966  3230662  3850381
UE   0         0         0         0
SN   1656a4ce acb26fe  5f839f76  7b680831
R    733840    1042168  852285   867311
RE   0         0         0         0
RDC  699789    934969   740874   756490
RA   385525    281198   204021   205384
RFR  27626     56397    0         0
RTR  34051     107199   111411   110821
RDP  0         0         0         0
RDPR 0         0         0         0
  
```

This unit as a receiver of broadcast messages

```

-----
      0      2      3
U    111847    121862   120029
R    7503     665700   749288
ESN  5d75b4b3 6d81d23  365ddd50
RI   630      34278    40291
RO   0        582      850
ROW  0        566      850
ROB  0        16        0
RAS  1571     123289   142256
  
```

This unit as a receiver of unicast messages

```

-----
      0      2      3
U    1         3308122  4370233
R    513846    879979   1009492
ESN  4458903a 6d841a84  7b4e7fa7
RI   66024     108924   102114
RO   0         0         0
ROW  0         0         0
ROB  0         0         0
RAS  130258    218924   228303
  
```

Gated Tx Buffered Message Statistics

```

-----
current sequence number: 0

total:                    0
current:                  0
high watermark:          0

delivered:                0
  
```

```

deliver failures:          0

buffer full drops:        0
message truncate drops:   0

gate close ref count:     0

num of supported clients:45

MRT Tx of broadcast messages
=====
Message high watermark: 3%
Total messages buffered at high watermark: 5677
[Per-client message usage at high watermark]
-----
Client name                Total messages  Percentage
Cluster Redirect Client    4153            73%
Route Cluster Client       419             7%
RRI Cluster Client         1105            19%

Current MRT buffer usage: 0%
Total messages buffered in real-time: 1
[Per-client message usage in real-time]
Legend:
    F - MRT messages sending when buffer is full
    L - MRT messages sending when cluster node leave
    R - MRT messages sending in Rx thread
-----
Client name                Total messages  Percentage  F  L  R
VPN Clustering HA Client    1             100%      0  0  0

MRT Tx of unitcast messages(to member_id:0)
=====
Message high watermark: 31%
Total messages buffered at high watermark: 4059
[Per-client message usage at high watermark]
-----
Client name                Total messages  Percentage
Cluster Redirect Client    3731            91%
RRI Cluster Client         328             8%

Current MRT buffer usage: 29%
Total messages buffered in real-time: 3924
[Per-client message usage in real-time]
Legend:
    F - MRT messages sending when buffer is full
    L - MRT messages sending when cluster node leave
    R - MRT messages sending in Rx thread
-----
Client name                Total messages  Percentage  F  L  R
Cluster Redirect Client    3607            91%      0  0  0
RRI Cluster Client         317             8%      0  0  0

MRT Tx of unitcast messages(to member_id:2)
=====
Message high watermark: 14%
Total messages buffered at high watermark: 578
[Per-client message usage at high watermark]
-----
Client name                Total messages  Percentage
VPN Clustering HA Client    578            100%

Current MRT buffer usage: 0%
Total messages buffered in real-time: 0

```

```

MRT Tx of unicast messages (to member_id:3)
=====
Message high watermark: 12%
Total messages buffered at high watermark: 573
[Per-client message usage at high watermark]
-----
Client name                               Total messages  Percentage
VPN Clustering HA Client                  572             99%
Cluster VPN Unique ID Client              1                0%

Current MRT buffer usage: 0%
Total messages buffered in real-time: 0

```

- **show cluster history**

クラスタの履歴、およびクラスタノードが参加できなかった理由や、ノードがクラスタを離れた理由に関するエラーメッセージが表示されます。

クラスタ全体のパケットのキャプチャ

クラスタでのパケットのキャプチャについては、次のコマンドを参照してください。

cluster exec capture

クラスタ全体のトラブルシューティングをサポートするには、**cluster exec capture** コマンドを使用して制御ノード上でのクラスタ固有トラフィックのキャプチャを有効にします。これで、クラスタ内のすべてのデータノードでも自動的に有効になります。

クラスタリソースのモニタリング

クラスタリソースのモニタリングについては、次のコマンドを参照してください。

show cluster {cpu | memory | resource} [options]

クラスタ全体の集約データを表示します。使用可能な *options* はデータのタイプによって異なります。

クラスタ トラフィックのモニタリング

クラスタトラフィックのモニタリングについては、次のコマンドを参照してください。

- **show conn [detail]、cluster exec show conn**

show conn コマンドは、フローがディレクタ、バックアップ、またはフォワーダのどのフローであるかを示します。**cluster exec show conn** コマンドを任意のノードで使用すると、すべての接続が表示されます。このコマンドの表示からは、1つのフローのトラフィックがクラスタ内のさまざまなASAにどのように到達するかがわかります。クラスタのスループットは、ロードバランシングの効率とコンフィギュレーションによって異なります。このコマンドを利用すると、ある接続のトラフィックがクラスタ内をどのように流れるかが

簡単にわかります。また、ロードバランサがフローのパフォーマンスにどのように影響を与えるかを理解するのに役立ちます。

また、**show conn detail** コマンドはフローモビリティの影響を受けるフローを表示します。

次に、**show conn detail** コマンドの出力例を示します。

```
ciscoasa/ASA2/slave# show conn detail
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
      B - initial SYN from outside, b - TCP state-bypass or nailed,
      C - CTIQBE media, c - cluster centralized,
      D - DNS, d - dump, E - outside back connection, e - semi-distributed,
      F - outside FIN, f - inside FIN,
      G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
      i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
      k - Skinny media, L - LISP triggered flow owner mobility,
      M - SMTP data, m - SIP media, n - GUP
      O - outbound data, o - offloaded,
      P - inside back connection,
      Q - Diameter, q - SQL*Net data,
      R - outside acknowledged FIN,
      R - UDP SUNRPC, r - inside acknowledged FIN, S - awaiting inside SYN,
      s - awaiting outside SYN, T - SIP, t - SIP transient, U - up,
      V - VPN orphan, W - WAAS,
      w - secondary domain backup,
      X - inspected by service module,
      x - per session, Y - director stub flow, y - backup stub flow,
      Z - Scansafe redirection, z - forwarding stub flow
ESP outside: 10.1.227.1/53744 NP Identity Ifc: 10.1.226.1/30604, , flags c, idle 0s,
uptime
1m21s, timeout 30s, bytes 7544, cluster sent/rcvd bytes 0/0, owners (0,255) Traffic
received
at interface outside Locally received: 7544 (93 byte/s) Traffic received at interface
NP
Identity Ifc Locally received: 0 (0 byte/s) UDP outside: 10.1.227.1/500 NP Identity
Ifc:
10.1.226.1/500, flags -c, idle 1m22s, uptime 1m22s, timeout 2m0s, bytes 1580, cluster
sent/rcvd bytes 0/0, cluster sent/rcvd total bytes 0/0, owners (0,255) Traffic
received at
interface outside Locally received: 864 (10 byte/s) Traffic received at interface
NP Identity
Ifc Locally received: 716 (8 byte/s)
```

接続フローのトラブルシューティングを行うには、最初にすべてのノードの接続を一覧表示します。それには、任意のノードで **cluster exec show conn** コマンドを入力します。ディレクタ (Y)、バックアップ (y)、およびフォワーダ (z) のフラグを持つフローを探します。次の例には、3つのすべてのASAでの172.18.124.187:22から192.168.103.131:44727へのSSH接続が表示されています。ASA1にはzフラグがあり、この接続のフォワーダであることを表しています。ASA3にはYフラグがあり、この接続のディレクタであることを表しています。ASA2には特別なフラグはなく、これがオーナーであることを表しています。アウトバウンド方向では、この接続のパケットはASA2の内部インターフェイスに入り、外部インターフェイスから出ていきます。インバウンド方向では、この接続のパケットはASA1およびASA3の外部インターフェイスに入り、クラスタ制御リンクを介してASA2に転送され、次にASA2の内部インターフェイスから出ていきます。

```

ciscoasa/ASA1/master# cluster exec show conn
ASA1(LOCAL):*****
18 in use, 22 most used
Cluster stub connections: 0 in use, 5 most used
TCP outside 172.18.124.187:22 inside 192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags z

ASA2:*****
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
TCP outside 172.18.124.187:22 inside 192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags UIO

ASA3:*****
10 in use, 12 most used
Cluster stub connections: 2 in use, 29 most used
TCP outside 172.18.124.187:22 inside 192.168.103.131:44727, idle 0:00:03, bytes
0, flags Y

```

- **show cluster info [conn-distribution | packet-distribution | loadbalance | flow-mobility counters]**

show cluster info conn-distribution コマンドと **show cluster info packet-distribution** コマンドは、すべてのクラスタノードへのトラフィック分散を表示します。これらのコマンドは、外部ロードバランサを評価し、調整するのに役立ちます。

show cluster info loadbalance コマンドは、接続再分散の統計情報を表示します。

show cluster info flow-mobility counters コマンドは、EID およびフローの所有者の動作情報を表示します。**show cluster info flow-mobility counters** コマンドについては次の出力を参照してください。

```

ciscoasa# show cluster info flow-mobility counters
EID movement notification received : 4
EID movement notification processed : 4
Flow owner moving requested : 2

```

- **show cluster info load-monitor [details]**

この**show cluster info load-monitor** コマンドは、最後の間隔のクラスタメンバのトラフィック負荷と、設定された間隔の合計数（デフォルトでは30）を表示します。各間隔の各測定値を表示するには、**details** キーワードを使用します。

```

ciscoasa(cfg-cluster)# show cluster info load-monitor
ID Unit Name
0 B
1 A_1
Information from all units with 20 second interval:
Unit      Connections      Buffer Drops      Memory Used      CPU Used
Average from last 1 interval:
0          0                0                14                25
1          0                0                16                20
Average from last 30 interval:
0          0                0                12                28
1          0                0                13                27

```

```
ciscoasa(cfg-cluster)# show cluster info load-monitor details
```

```
ID Unit Name
```

```
0 B
```

```
1 A_1
```

```
Information from all units with 20 second interval
```

```
Connection count captured over 30 intervals:
```

```
Unit ID 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
Unit ID 1
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
Buffer drops captured over 30 intervals:
```

```
Unit ID 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
Unit ID 1
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```
0 0 0 0 0 0
```

```

0          0          0          0          0          0

```

```

Memory usage(%) captured over 30 intervals:

```

```

Unit ID 0

```

```

25          25          30          30          30          35
25          25          35          30          30          30
25          25          30          25          25          35
30          30          30          25          25          25
25          20          30          30          30          30

```

```

Unit ID 1

```

```

30          25          35          25          30          30
25          25          35          25          30          35
30          30          35          30          30          30
25          20          30          25          25          30
20          30          35          30          30          35

```

```

CPU usage(%) captured over 30 intervals:

```

```

Unit ID 0

```

```

25          25          30          30          30          35
25          25          35          30          30          30
25          25          30          25          25          35
30          30          30          25          25          25
25          20          30          30          30          30

```

```

Unit ID 1

```

```

30          25          35          25          30          30
25          25          35          25          30          35
30          30          35          30          30          30
25          20          30          25          25          30
20          30          35          30          30          35

```

• **show cluster** {**access-list** | **conn** | **traffic** | **user-identity** | **xlate**} [*options*]

クラスタ全体の集約データを表示します。使用可能な *options* はデータのタイプによって異なります。

show cluster access-list コマンドについては次の出力を参照してください。

```
ciscoasa# show cluster access-list
hitcnt display order: cluster-wide aggregated result, unit-A, unit-B, unit-C, unit-D
access-list cached ACL log flows: total 0, denied 0 (deny-flow-max 4096) alert-interval
  300
access-list 101; 122 elements; name hash: 0xe7d586b5
access-list 101 line 1 extended permit tcp 192.168.143.0 255.255.255.0 any eq www
(hitcnt=0, 0, 0, 0, 0) 0x207a2b7d
access-list 101 line 2 extended permit tcp any 192.168.143.0 255.255.255.0 (hitcnt=0,
  0, 0, 0, 0) 0xfe4f4947
access-list 101 line 3 extended permit tcp host 192.168.1.183 host 192.168.43.238
(hitcnt=1, 0, 0, 0, 1) 0x7b521307
access-list 101 line 4 extended permit tcp host 192.168.1.116 host 192.168.43.238
(hitcnt=0, 0, 0, 0, 0) 0x5795c069
access-list 101 line 5 extended permit tcp host 192.168.1.177 host 192.168.43.238
(hitcnt=1, 0, 0, 1, 0) 0x51bde7ee
access-list 101 line 6 extended permit tcp host 192.168.1.177 host 192.168.43.13
(hitcnt=0, 0, 0, 0, 0) 0x1e68697c
access-list 101 line 7 extended permit tcp host 192.168.1.177 host 192.168.43.132
(hitcnt=2, 0, 0, 1, 1) 0xc1ce5c49
access-list 101 line 8 extended permit tcp host 192.168.1.177 host 192.168.43.192
(hitcnt=3, 0, 1, 1, 1) 0xb6f59512
access-list 101 line 9 extended permit tcp host 192.168.1.177 host 192.168.43.44
(hitcnt=0, 0, 0, 0, 0) 0xdc104200
access-list 101 line 10 extended permit tcp host 192.168.1.112 host 192.168.43.44
(hitcnt=429, 109, 107, 109, 104)
0xce4f281d
access-list 101 line 11 extended permit tcp host 192.168.1.170 host 192.168.43.238
(hitcnt=3, 1, 0, 0, 2) 0x4143a818
access-list 101 line 12 extended permit tcp host 192.168.1.170 host 192.168.43.169
(hitcnt=2, 0, 1, 0, 1) 0xb18dfea4
access-list 101 line 13 extended permit tcp host 192.168.1.170 host 192.168.43.229
(hitcnt=1, 1, 0, 0, 0) 0x21557d71
access-list 101 line 14 extended permit tcp host 192.168.1.170 host 192.168.43.106
(hitcnt=0, 0, 0, 0, 0) 0x7316e016
access-list 101 line 15 extended permit tcp host 192.168.1.170 host 192.168.43.196
(hitcnt=0, 0, 0, 0, 0) 0x013fd5b8
access-list 101 line 16 extended permit tcp host 192.168.1.170 host 192.168.43.75
(hitcnt=0, 0, 0, 0, 0) 0x2c7dba0d
```

使用中の接続の、すべてのノードでの合計数を表示するには、次のとおりに入力します。

```
ciscoasa# show cluster conn count
Usage Summary In Cluster:*****
  200 in use (cluster-wide aggregated)
  cl2 (LOCAL):*****
  100 in use, 100 most used

  cl1:*****
  100 in use, 100 most used
```

• show asp cluster counter

このコマンドは、データパスのトラブルシューティングに役立ちます。

クラスタのルーティングのモニタリング

クラスタのルーティングについては、次のコマンドを参照してください。

- **show route cluster**
- **debug route cluster**

クラスタのルーティング情報を表示します。

- **show lisp eid**

EIDs と サイト ID を示す ASA EID テーブルを表示します。

cluster exec show lisp eid コマンドからの、次の出力を参照してください。

```
ciscoasa# cluster exec show lisp eid
L1 (LOCAL) :*****
  LISP EID      Site ID
  33.44.33.105    2
  33.44.33.201    2
  11.22.11.1      4
  11.22.11.2      4
L2:*****
  LISP EID      Site ID
  33.44.33.105    2
  33.44.33.201    2
  11.22.11.1      4
  11.22.11.2      4
```

- **show asp table classify domain inspect-lisp**

このコマンドは、トラブルシューティングに役立ちます。

クラスタリングのロギングの設定

クラスタリングのロギングの設定については、次のコマンドを参照してください。

logging device-id

クラスタ内の各ノードは、syslog メッセージを個別に生成します。**logging device-id** コマンドを使用すると、同一または異なるデバイス ID 付きで syslog メッセージを生成することができ、クラスタ内の同一または異なるノードからのメッセージのように見せることができます。

クラスタのインターフェイスのモニタリング

クラスタのインターフェイスのモニタリングについては、次のコマンドを参照してください。

- **show cluster interface-mode**

クラスタ インターフェイスのモードを表示します。

- **show port-channel**

ポートチャネルがスパンドかどうかに関する情報が含まれます。

- **show lacp cluster {system-mac | system-id}**

cLACP システム ID およびプライオリティを表示します。

- **debug lacp cluster [all | ccp | misc | protocol]**

cLACP のデバッグ メッセージを表示します。

- **show interface**

MAC アドレスを使用している場合、その使用状況を表示します。

```
ciscoasa# show interface port-channel1.3151
Interface Port-channel1.3151 "inside", is up, line protocol is up
Hardware is EtherChannel/LACP, BW 1000 Mbps, DLY 10 usec
VLAN identifier 3151
MAC address aaaa.1111.1234, MTU 1500
Site Specific MAC address aaaa.1111.aaaa
IP address 10.3.1.1, subnet mask 255.255.255.0
Traffic Statistics for "inside":
132269 packets input, 6483425 bytes
1062 packets output, 110448 bytes
98530 packets dropped
```

クラスタリングのデバッグ

クラスタリングのデバッグについては、次のコマンドを参照してください。

- **debug cluster [ccp | datapath | fsm | general | hc | license | rpc | transport]**

クラスタリングのデバッグ メッセージを表示します。

- **debug cluster flow-mobility**

クラスタリング フロー モビリティ 関連のイベントを表示します。

- **debug lisp eid-notify-intercept**

EID 通知メッセージ代行受信時のイベントを表示します。

- **show cluster info trace**

show cluster info trace コマンドは、トラブルシューティングのためのデバッグ情報を表示します。

show cluster info trace コマンドについては次の出力を参照してください。

```
ciscoasa# show cluster info trace
Feb 02 14:19:47.456 [DEBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
Feb 02 14:19:47.456 [DEBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
Feb 02 14:19:47.456 [DEBUG]Send CCP message to all: CCP_MSG_KEEPALIVE from 80-1 at
MASTER
```

ASA クラスタリングの例

以下の例には、一般的な導入での ASA のクラスタ関連のすべてのコンフィギュレーションが含まれます。

ASA およびスイッチのコンフィギュレーションの例

次のコンフィギュレーション例は、ASA とスイッチ間の次のインターフェイスを接続します。

ASA インターフェイス	スイッチ インターフェイス
イーサネット 1/2	GigabitEthernet 1/0/15
イーサネット 1/3	GigabitEthernet 1/0/16
イーサネット 1/4	GigabitEthernet 1/0/17
イーサネット 1/5	GigabitEthernet 1/0/18

ASA の設定

各ユニットのインターフェイス モード

```
cluster interface-mode spanned force
```

ASA1 制御ユニットのブートストラップ設定

```
interface Ethernet1/6
  channel-group 1 mode on
  no shutdown
!
interface Ethernet1/7
  channel-group 1 mode on
  no shutdown
!
interface Port-channel1
  description Clustering Interface
!
cluster group Moya
  local-unit A
  cluster-interface Port-channel1 ip 10.0.0.1 255.255.255.0
  priority 10
  key emphyri0
  enable noconfirm
```

ASA2 データユニットのブートストラップ設定

```
interface Ethernet1/6
```



```
channel-group 1 mode on
no shutdown
!
interface Ethernet1/7
channel-group 1 mode on
no shutdown
!
interface Port-channel1
description Clustering Interface
!
cluster group Moya
local-unit B
cluster-interface Port-channel1 ip 10.0.0.2 255.255.255.0
priority 11
key emphyri0
enable as-slave
```

制御ユニットのインターフェイス設定

```
ip local pool mgmt-pool 10.53.195.231-10.53.195.232

interface Ethernet1/2
channel-group 10 mode active
no shutdown
!
interface Ethernet1/3
channel-group 10 mode active
no shutdown
!
interface Ethernet1/4
channel-group 11 mode active
no shutdown
!
interface Ethernet1/5
channel-group 11 mode active
no shutdown
!
interface Management1/1
management-only
nameif management
ip address 10.53.195.230 cluster-pool mgmt-pool
security-level 100
no shutdown
!
interface Port-channel10
port-channel span-cluster
mac-address aaaa.bbbb.cccc
nameif inside
security-level 100
ip address 209.165.200.225 255.255.255.224
!
interface Port-channel11
port-channel span-cluster
mac-address aaaa.dddd.cccc
nameif outside
security-level 0
ip address 209.165.201.1 255.255.255.224
```

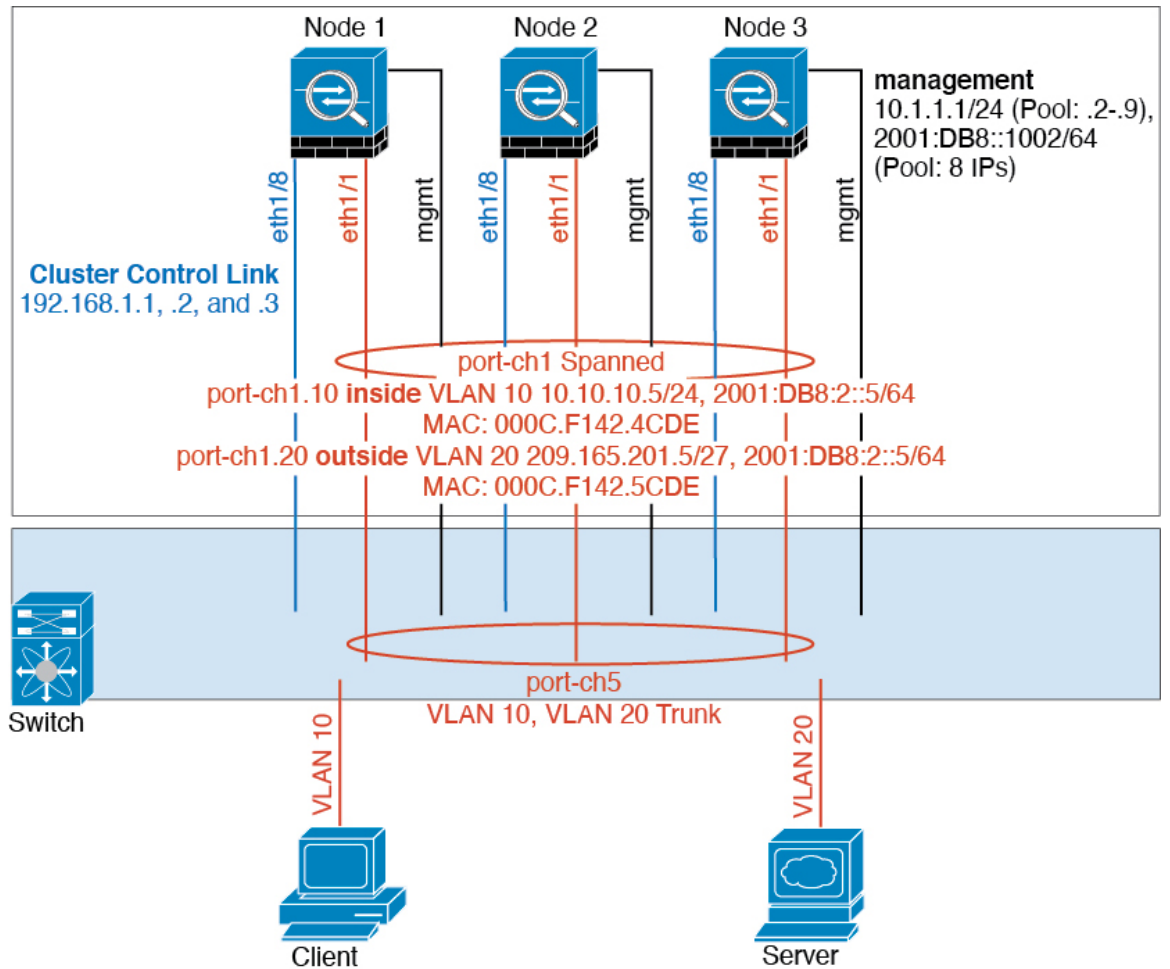
Cisco IOS スイッチのコンフィギュレーション

```
interface GigabitEthernet1/0/15
  switchport access vlan 201
  switchport mode access
  spanning-tree portfast
  channel-group 10 mode active
!
interface GigabitEthernet1/0/16
  switchport access vlan 201
  switchport mode access
  spanning-tree portfast
  channel-group 10 mode active
!
interface GigabitEthernet1/0/17
  switchport access vlan 401
  switchport mode access
  spanning-tree portfast
  channel-group 11 mode active
!
interface GigabitEthernet1/0/18
  switchport access vlan 401
  switchport mode access
  spanning-tree portfast
  channel-group 11 mode active

interface Port-channel10
  switchport access vlan 201
  switchport mode access

interface Port-channel11
  switchport access vlan 401
  switchport mode access
```

スティック上のファイアウォール



異なるセキュリティドメインからのデータトラフィックには、異なる VLAN が関連付けられます。たとえば内部ネットワーク用には VLAN 10、外部ネットワークには VLAN 20 とします。各 ASA は単一の物理ポートがあり、外部スイッチまたはルータに接続されます。トランキングがイネーブされているので、物理リンク上のすべてのパケットが 802.1q カプセル化されます。ASA は、VLAN 10 と VLAN 20 の間のファイアウォールです。

スパンド EtherChannel を使用するとき、スイッチ側ですべてのデータリンクがグループ化されて 1 つの EtherChannel となります。ASA が使用不可能になった場合は、スイッチは残りのユニット間でトラフィックを再分散します。

各ユニットのインターフェイスモード

```
cluster interface-mode spanned force
```

ユニット 1 制御ユニットのブートストラップ設定

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa1
cluster-interface ethernet1/8 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
enable noconfirm
```

ユニット 2 データユニットのブートストラップ設定

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa2
cluster-interface ethernet1/8 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

ユニット 3 データユニットのブートストラップ設定

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa3
cluster-interface ethernet1/8 ip 192.168.1.3 255.255.255.0
priority 3
key chuntheunavoidable
enable as-slave
```

制御ユニットのインターフェイス設定

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
security-level 100
management-only
no shutdown

interface ethernet1/1
channel-group 1 mode active
no shutdown
```

```

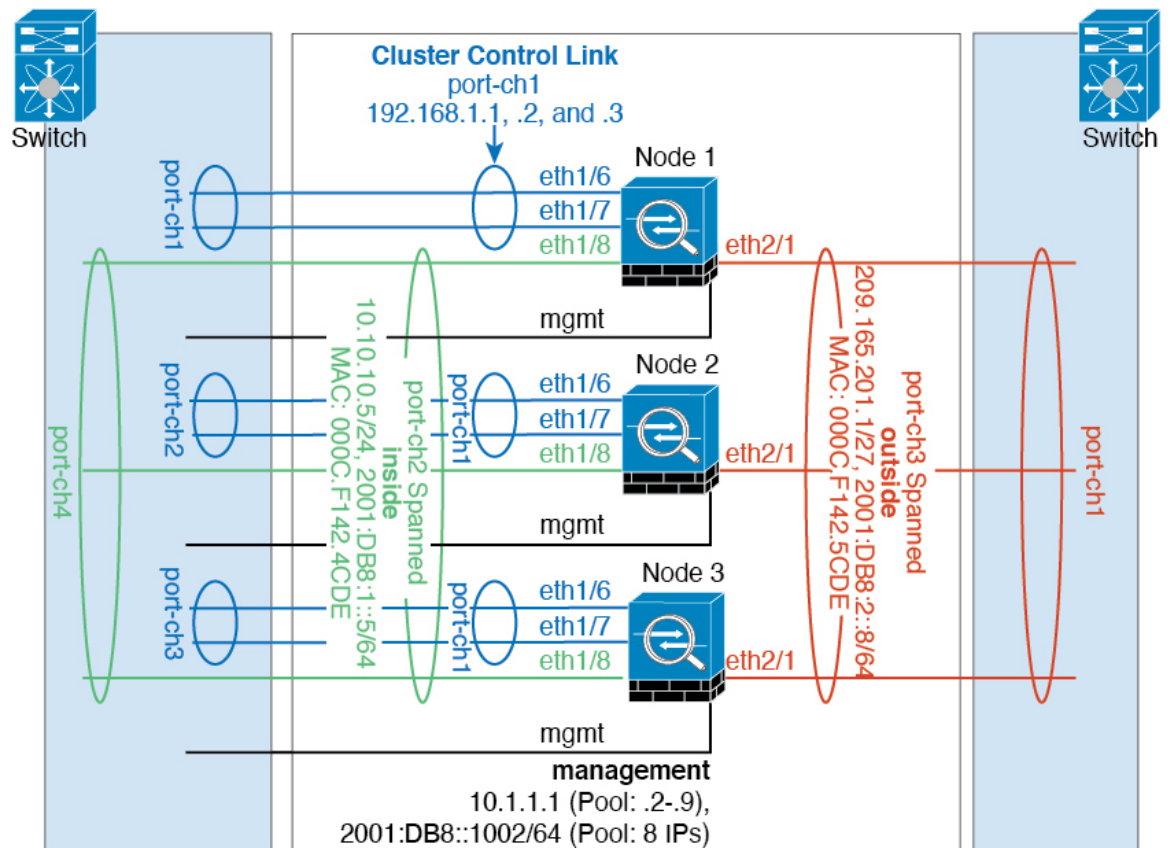
interface port-channel 1
port-channel span-cluster

interface port-channel 1.10
vlan 10
nameif inside
ip address 10.10.10.5 255.255.255.0
ipv6 address 2001:DB8:1::5/64
mac-address 000C.F142.4CDE

interface port-channel 1.20
vlan 20
nameif outside
ip address 209.165.201.1 255.255.255.224
ipv6 address 2001:DB8:2::8/64
mac-address 000C.F142.5CDE

```

トラフィックの分離



内部ネットワークと外部ネットワークの間で、トラフィックを物理的に分離できます。

上の図に示すように、左側に一方のスパンドEtherChannelがあり、内部スイッチに接続されています。他方は右側にあり、外部スイッチに接続されています。必要であれば、各EtherChannel上にVLANサブインターフェイスを作成することもできます。

各ユニットのインターフェイスモード

```
cluster interface-mode spanned force
```

ユニット 1 制御ユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asa1
cluster-interface port-channell1 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
enable noconfirm
```

ユニット 2 データユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asa2
cluster-interface port-channell1 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

ユニット 3 データユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL
```

```
cluster group cluster1
local-unit asa3
cluster-interface port-channel1 ip 192.168.1.3 255.255.255.0
priority 3
key chuntheunavoidable
enable as-slave
```

制御ユニットのインターフェイス設定

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
security-level 100
management-only
no shutdown

interface ethernet 1/8
channel-group 2 mode active
no shutdown

interface port-channel 2
port-channel span-cluster
nameif inside
ip address 10.10.10.5 255.255.255.0
ipv6 address 2001:DB8:1::5/64
mac-address 000C.F142.4CDE

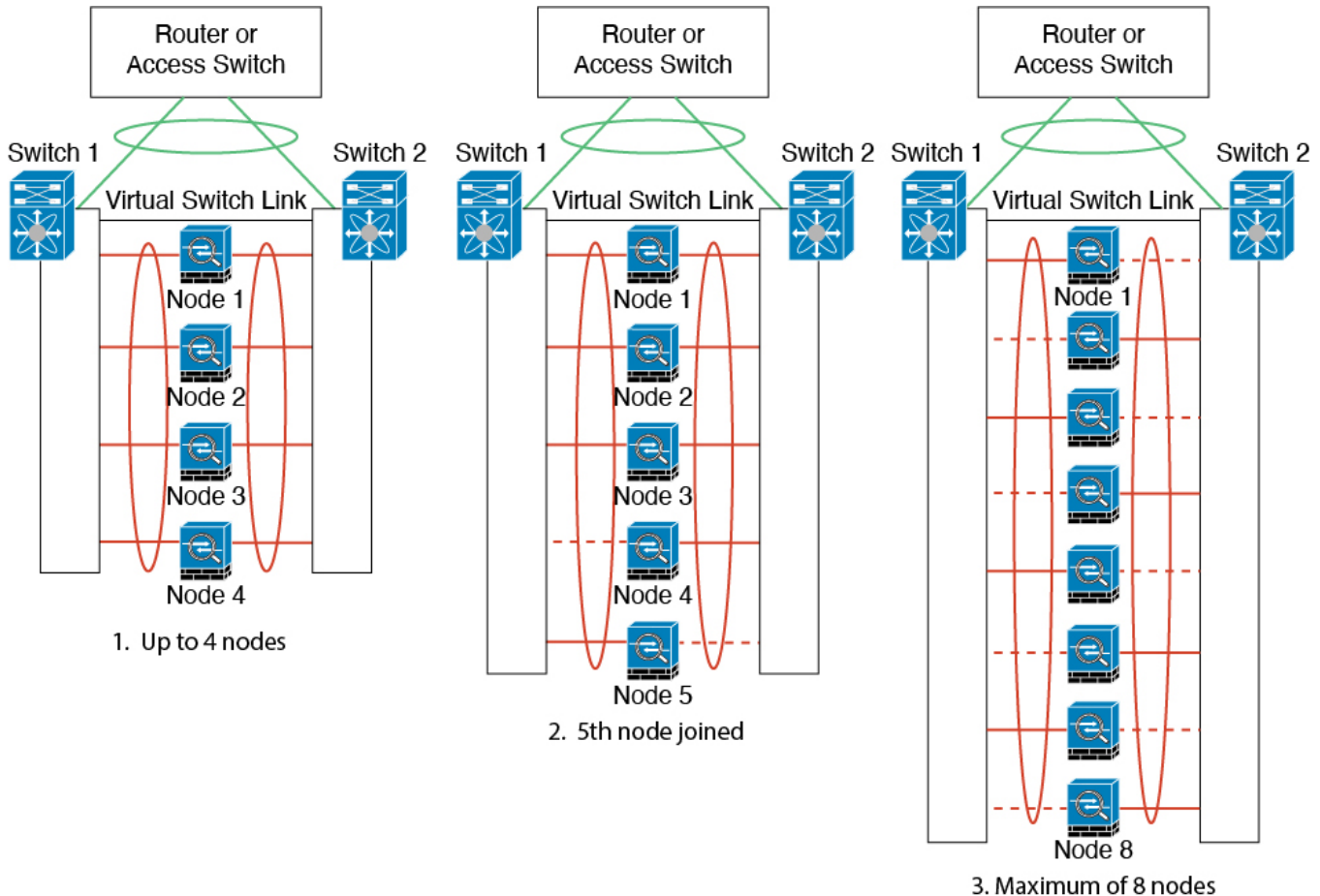
interface ethernet 2/1
channel-group 3 mode active
no shutdown

interface port-channel 3
port-channel span-cluster
nameif outside
ip address 209.165.201.1 255.255.255.224
ipv6 address 2001:DB8:2::8/64
mac-address 000C.F142.5CDE
```

スパンド EtherChannel とバックアップリンク（従来の 8 アクティブ/8 スタンバイ）

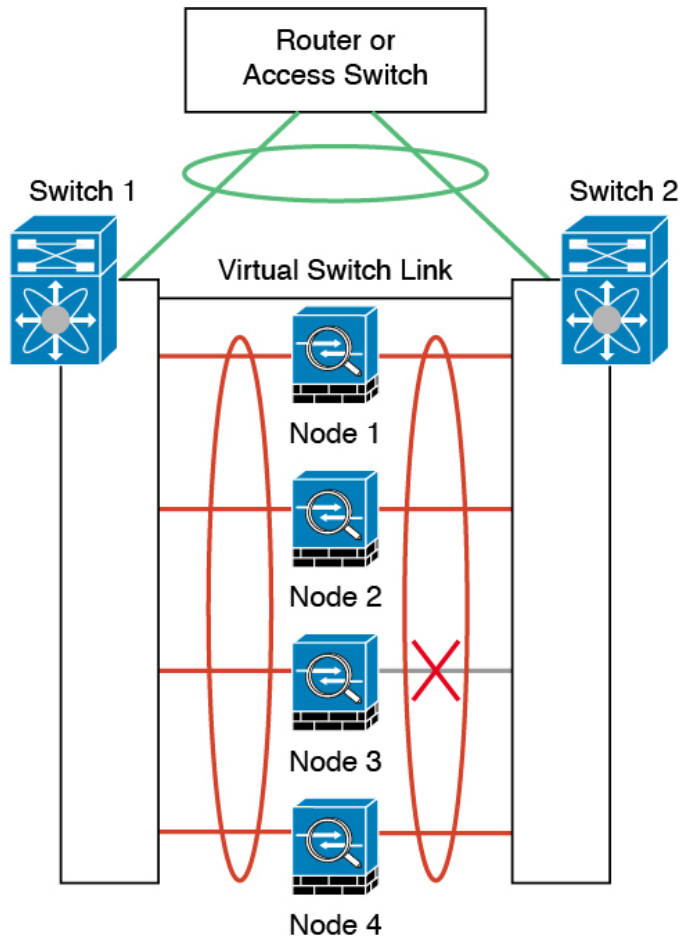
従来の EtherChannel のアクティブ ポートの最大数は、スイッチ側からの 8 に制限されます。8 ユニットから成るクラスタがあり、EtherChannel にユニットあたり 2 ポートを割り当てた場合は、合計 16 ポートのうち 8 ポートをスタンバイ モードにする必要があります。ASA は、どのリンクをアクティブまたはスタンバイにするかを、LACP を使用してネゴシエートします。VSS、vPC、StackWise、または StackWise Virtual を使用してマルチスイッチ EtherChannel をイネーブルにした場合は、スイッチ間の冗長性を実現できます。ASA では、すべての物理ポートが最初にスロット番号順、次にポート番号順に並べられます。次の図では、番号の小さいポートが「制御」ポートとなり（たとえば Ethernet 1/1）、他方が「データ」ポートとなります（た

たとえば Ethernet 1/2)。ハードウェア接続の対称性を保証する必要があります。つまり、すべての制御リンクは1台のスイッチが終端となり、すべてのデータリンクは別のスイッチが終端となっている必要があります（冗長スイッチシステムが使用されている場合）。次の図は、クラスタに参加するユニットが増えてリンクの総数が増加したときに、どのようになるかを示しています。

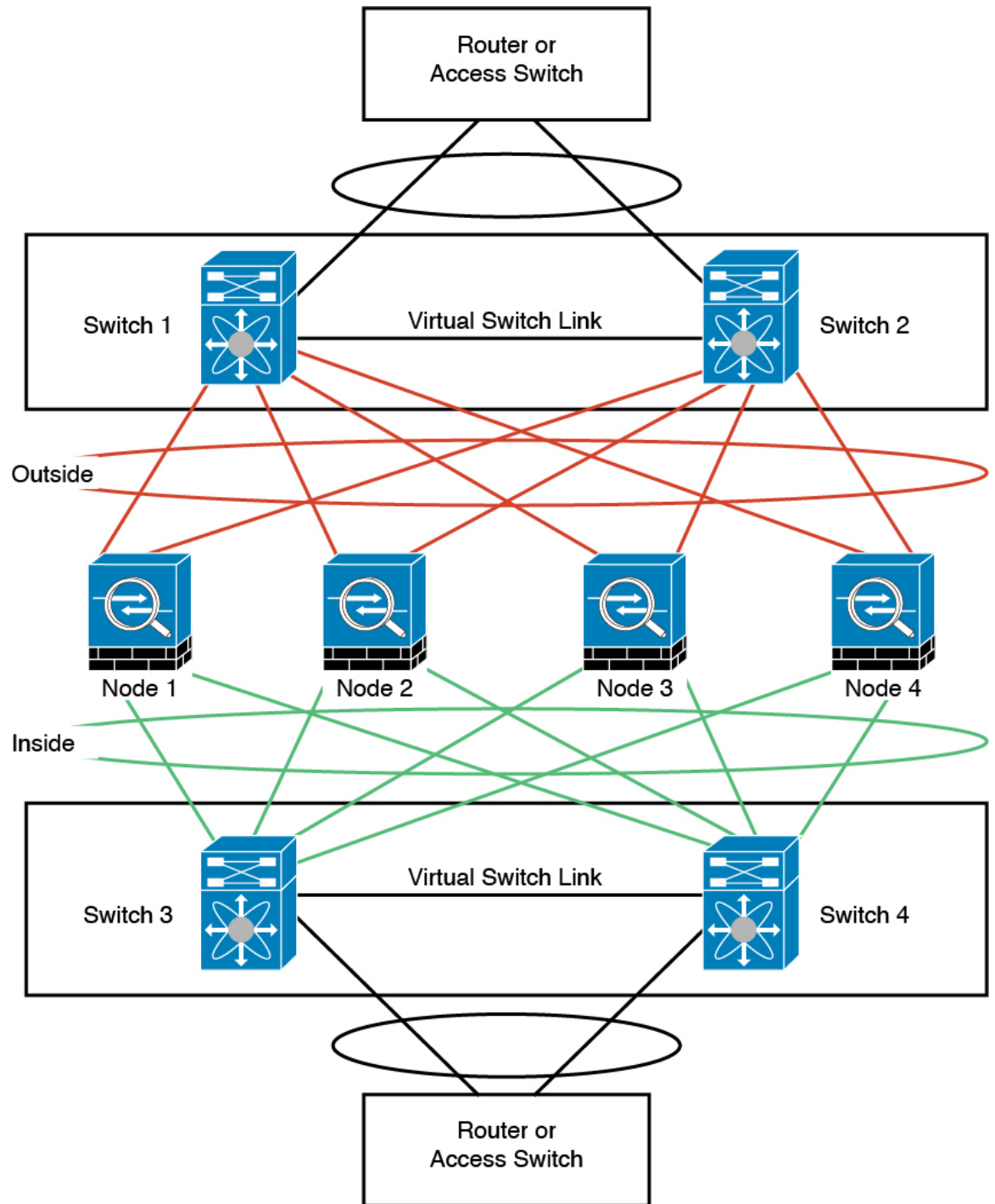


原則として、初めにチャンネル内のアクティブポート数を最大化し、そのうえで、アクティブな制御ポートとアクティブなデータポートの数のバランスを保ちます。5番目のユニットがクラスタに参加したときは、トラフィックがすべてのユニットに均等には分散されないことに注意してください。

リンクまたはデバイスの障害が発生したときも、同じ原則で処理されます。その結果、ロードバランシングが理想的な状態にはならないこともあります。次の図は、4ユニットのクラスタを示しています。このユニットの1つで、単一リンク障害が発生しています。



ネットワーク内に複数の EtherChannel を設定することも考えられます。次の図では、EtherChannel が内部に 1 つ、外部に 1 つあります。ASA は、一方の EtherChannel で制御とデータの両方のリンクが障害状態になった場合にクラスタから削除されます。これは、その ASA がすでに内部ネットワークへの接続を失っているにもかかわらず、外部ネットワークからトラフィックを受信するのを防ぐためです。



各ユニットのインターフェイスモード

```
cluster interface-mode spanned force
```

ユニット 1 制御ユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface ethernet 1/8
channel-group 1 mode on
no shutdown

interface ethernet 2/1
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asal
cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
enable noconfirm
```

ユニット 2 データユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface ethernet 1/8
channel-group 1 mode on
no shutdown

interface ethernet 2/1
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asa2
cluster-interface port-channel1 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

ユニット 3 データユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface ethernet 1/8
channel-group 1 mode on
no shutdown

interface ethernet 2/1
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asa3
cluster-interface port-channel1 ip 192.168.1.3 255.255.255.0
priority 3
key chuntheunavoidable
enable as-slave
```

ユニット 4 データユニットのブートストラップ設定

```
interface ethernet 1/6
channel-group 1 mode on
no shutdown

interface ethernet 1/7
channel-group 1 mode on
no shutdown

interface ethernet 1/8
channel-group 1 mode on
no shutdown

interface ethernet 2/1
channel-group 1 mode on
no shutdown

interface port-channel 1
description CCL

cluster group cluster1
local-unit asa4
cluster-interface port-channel1 ip 192.168.1.4 255.255.255.0
priority 4
key chuntheunavoidable
enable as-slave
```

制御ユニットのインターフェイス設定

```
ip local pool mgmt 10.1.1.2-10.1.1.9

interface management 1/1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
security-level 100
management-only

interface ethernet 2/6
channel-group 3 mode active vss-id 1
no shutdown

interface ethernet 2/7
channel-group 3 mode active vss-id 2
no shutdown

interface port-channel 3
port-channel span-cluster vss-load-balance
nameif inside
ip address 10.10.10.5 255.255.255.0
mac-address 000C.F142.4CDE

interface ethernet 2/8
channel-group 4 mode active vss-id 1
no shutdown

interface ethernet 2/9
channel-group 4 mode active vss-id 2
no shutdown

interface port-channel 4
port-channel span-cluster vss-load-balance
nameif outside
ip address 209.165.201.1 255.255.255.224
mac-address 000C.F142.5CDE
```

ルーテッドモードサイト間クラスタリングの OTV 設定

スパンド EtherChannel を使用したルーテッドモードに対するサイト間クラスタリングの成功は、OTV の適切な設定とモニタリングによって異なります。OTV は、DCI 全体にパケットを転送することで、重要な役割を果たします。OTV は、転送テーブルに MAC アドレスを学習するときのみ、DCI 全体にユニキャストパケットを転送します。MAC アドレスが OTV 転送テーブルに学習されていない場合、ユニキャストパケットはドロップされます。

OTV 設定の例

```
//Sample OTV config:
//3151 - Inside VLAN, 3152 - Outside VLAN, 202 - CCL VLAN
//aaa.1111.1234 - ASA inside interface global vMAC
//0050.56A8.3D22 - Server MAC

feature ospf
feature otv

mac access-list ALL_MACs
```

ルーテッドモード サイト間クラスタリングの OTV 設定

```

10 permit any any
mac access-list HSRP_VMAC
10 permit aaaa.1111.1234 0000.0000.0000 any
20 permit aaaa.2222.1234 0000.0000.0000 any
30 permit any aaaa.1111.1234 0000.0000.0000
40 permit any aaaa.2222.1234 0000.0000.0000
vlan access-map Local 10
  match mac address HSRP_VMAC
  action drop
vlan access-map Local 20
  match mac address ALL_MACs
  action forward
vlan filter Local vlan-list 3151-3152

//To block global MAC with ARP inspection:
arp access-list HSRP_VMAC_ARP
10 deny aaaa.1111.1234 0000.0000.0000 any
20 deny aaaa.2222.1234 0000.0000.0000 any
30 deny any aaaa.1111.1234 0000.0000.0000
40 deny any aaaa.2222.1234 0000.0000.0000
50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP 3151-3152

no ip igmp snooping optimise-multicast-flood
vlan 1,202,1111,2222,3151-3152

otv site-vlan 2222
mac-list GMAC_DENY seq 10 deny aaaa.aaaa.aaaa ffff.ffff.ffff
mac-list GMAC_DENY seq 20 deny aaaa.bbbb.bbbb ffff.ffff.ffff
mac-list GMAC_DENY seq 30 permit 0000.0000.0000 0000.0000.0000
route-map stop-GMAC permit 10
  match mac-list GMAC_DENY

interface Overlay1
  otv join-interface Ethernet8/1
  otv control-group 239.1.1.1
  otv data-group 232.1.1.0/28
  otv extend-vlan 202, 3151
  otv arp-nd timeout 60
  no shutdown

interface Ethernet8/1
  description uplink_to_OTV_cloud
  mtu 9198
  ip address 10.4.0.18/24
  ip igmp version 3
  no shutdown

interface Ethernet8/2

interface Ethernet8/3
  description back_to_default_vdc_e6/39
  switchport
  switchport mode trunk
  switchport trunk allowed vlan 202,2222,3151-3152
  mac packet-classify
  no shutdown

otv-isis default
  vpn Overlay1
  redistribute filter route-map stop-GMAC
otv site-identifier 0x2
//OTV flood not required for ARP inspection:

```

```
otv flood mac 0050.56A8.3D22 vlan 3151
```

サイト障害のために必要な OTV フィルタの変更

サイトがダウンした場合は、グローバル MAC アドレスをそれ以上ブロックしなくて済むように、フィルタを OTV から削除する必要があります。必要なくつかの追加設定があります。

機能しているサイトで OTV スイッチ上の ASA グローバル MAC アドレスに対するスタティック エントリを追加する必要があります。このエントリによって、反対側の OTV はオーバーレイ インターフェイスにこれらのエントリを追加できます。サーバとクライアントに ASA 用の ARP エントリがすでにある場合（これは既存の接続の場合です）、ARP は再送信されないのので、この手順が必要になります。したがって、OTV は転送テーブルに ASA グローバル MAC アドレスを学習する機会はありません。OTV には転送テーブル内にグローバル MAC アドレスがなく、OTV の設計ごとに OTV はオーバーレイ インターフェイスを介してユニキャスト パケットをフラッディングしないので、ユニキャスト パケットはサーバからのグローバル MAC アドレスにドロップされ、既存の接続は切断されます。

```
//OTV filter configs when one of the sites is down

mac-list GMAC_A seq 10 permit 0000.0000.0000 0000.0000.0000
route-map a-GMAC permit 10
  match mac-list GMAC_A

otv-isis default
  vpn Overlay1
    redistribute filter route-map a-GMAC

no vlan filter Local vlan-list 3151

//For ARP inspection, allow global MAC:
arp access-list HSRP_VMAC_ARP_Allow
  50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP_Allow 3151-3152

mac address-table static aaaa.1111.1234 vlan 3151 interface Ethernet8/3
//Static entry required only in the OTV in the functioning Site
```

他のサイトが復元した場合は、フィルタを再度追加して、OTV でこのスタティック エントリを削除する必要があります。グローバル MAC アドレスのオーバーレイ エントリをクリアするには、両方の OTV でダイナミック MAC アドレス テーブルをクリアすることが非常に重要です。

MAC アドレス テーブルのクリア

サイトがダウンし、グローバル MAC アドレスへのスタティック エントリが OTV に追加される場合は、他の OTV がオーバーレイ インターフェイスのグローバル MAC アドレスを学習できるようにする必要があります。他のサイトが起動したら、これらのエントリをクリアする必要があります。OTV の転送テーブルにこれらのエントリがないことを確認するために、MAC アドレス テーブルを必ず消去してください。

```
cluster-N7k6-OTV# show mac address-table
```

```

Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
G -    d867.d900.2e42 static - F F sup-eth1(R)
O 202 885a.92f6.44a5 dynamic - F F Overlay1
* 202 885a.92f6.4b8f dynamic 5 F F Eth8/3
O 3151 0050.5660.9412 dynamic - F F Overlay1
* 3151 aaaa.1111.1234 dynamic 50 F F Eth8/3

```

OTV ARP キャッシュのモニタリング

OTV は、OTV インターフェイス全体で学習した IP アドレスに対するプロキシ ARP への ARP キャッシュを維持します。

```

cluster-N7k6-OTV# show otv arp-nd-cache
OTV ARP/ND L3->L2 Address Mapping Cache

Overlay Interface Overlay1
VLAN MAC Address Layer-3 Address Age Expires In
3151 0050.5660.9412 10.0.0.2 1w0d 00:00:31
cluster-N7k6-OTV#

```

サイト間クラスタリングの例

次の例では、サポートされるクラスタ導入を示します。

サイト固有の MAC アドレスおよび IP アドレスを使用したスパンド EtherChannel ルーテッドモードの例

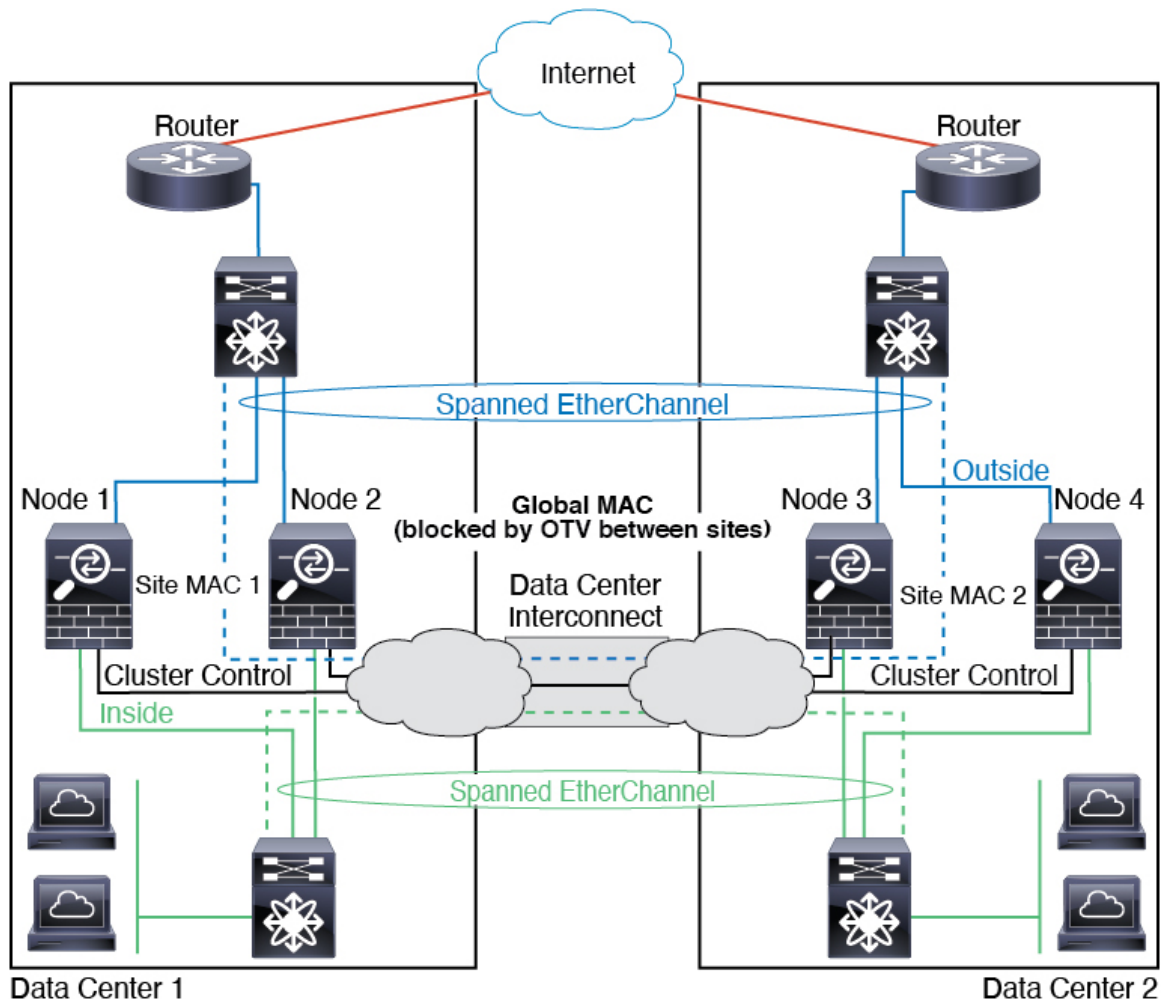
次の例では、各サイトのゲートウェイ ルータと内部ネットワーク間に配置された（イーストウェスト挿入）2つのデータセンターのそれぞれに2つのクラスタメンバーがある場合を示します。クラスタメンバーは、DCI経由のクラスタ制御リンクによって接続されています。各サイトのクラスタメンバーは、内部および外部両方のネットワークに対しスパンド EtherChannel を使用してローカルスイッチに接続します。各 EtherChannel は、クラスタ内のすべてのシャーシにスパンされます。

データ VLAN は、オーバーレイ トランスポート 仮想化 (OTV)（または同様のもの）を使用してサイトに拡張されます。トラフィックがクラスタ宛てである場合にトラフィックが DCI を通過して他のサイトに送信されないようにするには、グローバル MAC アドレスをブロックするフィルタを追加する必要があります。1つのサイトのクラスタノードが到達不能になった場合、トラフィックが他のサイトのクラスタノードに送信されるようにフィルタを削除する必要があります。Vacl を使用して、グローバルの MAC アドレスのフィルタリングする必要があります。F3 シリーズラインカードが搭載された Nexus などの一部のスイッチでは、グローバル MAC アドレスからの ARP パケットをブロックするために ARP インスペクションも使用する必要があります。ARP インスペクションでは、ASA でサイトの MAC アドレスとサイトの IP アドレスの両方を設定する必要があります。サイトの MAC アドレスのみを設定する場合は必ず ARP インスペクションを無効にしてください。

クラスタは、内部ネットワークのゲートウェイとして機能します。すべてのクラスタノード間で共有されるグローバルな仮想 MAC は、パケットを受信するためだけに使用されます。発信パケットは、各 DC クラスタからのサイト固有の MAC アドレスを使用します。この機能により、スイッチが2つの異なるポートで両方のサイトから同じグローバル MAC アドレスを学習してしまうのを防いでいます。MAC フラッピングが発生しないよう、サイト MAC アドレスのみを学習します。

この場合のシナリオは次のとおりです。

- クラスタから送信されるすべての出力パケットは、サイトの MAC アドレスを使用し、データセンターでローカライズされます。
- クラスタへのすべての入力パケットは、グローバル MAC アドレスを使用して送信されるため、両方のサイトにある任意のノードで受信できます。OTV のフィルタによって、データセンター内のトラフィックがローカライズされます。



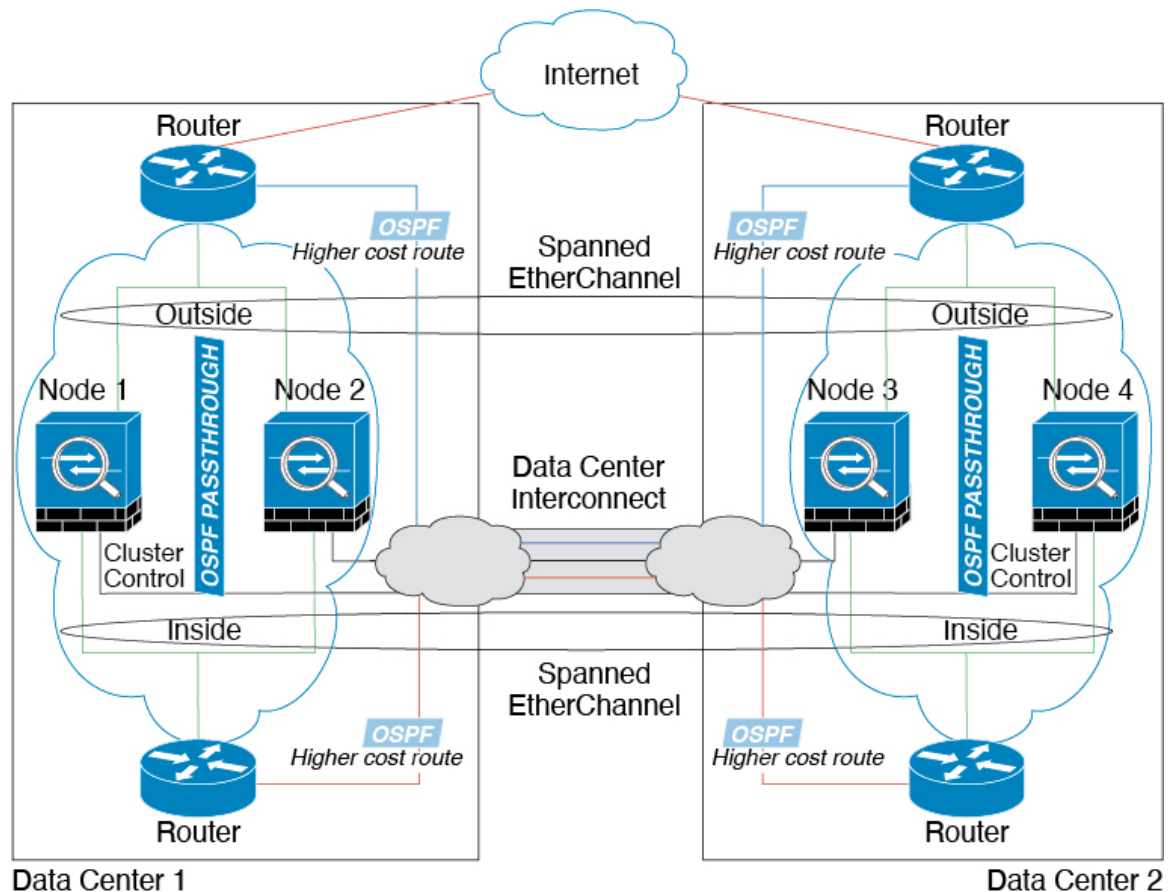
スパンド EtherChannel トランスペアレント モード ノースサウス サイト間の例

次の例では、内部ルータと外部ルータの間に配置された（ノースサウス挿入）2つのデータセンターのそれぞれに2つのクラスタメンバーがある場合を示します。クラスタメンバーは、DCI経由のクラスタ制御リンクによって接続されています。各サイトのクラスタメンバーは、内部および外部のスパンド EtherChannels を使用してローカルスイッチに接続します。各 EtherChannel は、クラスタ内のすべてのシャーシにスパンドされます。

各データセンターの内部ルータと外部ルータは OSPF を使用し、トランスペアレント ASA を通過します。MAC とは異なり、ルータの IP はすべてのルータで一意です。DCI に高コストルートを割り当てることにより、特定のサイトですべてのクラスタメンバーがダウンしない限り、トラフィックは各データセンター内に維持されます。クラスタが非対称型の接続を維持するため、ASA を通過する低コストのルートは、各サイトで同じブリッジグループを横断する必要があります。1つのサイトのすべてのクラスタメンバーに障害が発生した場合、トラフィックは各ルータから DCI 経由で他のサイトのクラスタメンバーに送られます。

各サイトのスイッチの実装には、次のものを含めることができます。

- サイト間 VSS、vPC、StackWise、StackWise Virtual：このシナリオでは、データセンター 1 に 1 台のスイッチをインストールし、データセンター 2 に別のスイッチをインストールします。1つのオプションとして、各データセンターのクラスタノードはローカルスイッチだけに接続し、冗長スイッチトラフィックは DCI を経由します。この場合、接続のほとんどの部分は各データセンターに対してローカルに維持されます。DCI が余分なトラフィックを処理できる場合、必要に応じて、各ノードを DCI 経由で両方のスイッチに接続できます。この場合、トラフィックは複数のデータセンターに分散されるため、DCI を非常に堅牢にするためには不可欠です。
- 各サイトのローカル VSS、vPC、StackWise、StackWise Virtual：スイッチの冗長性を高めるには、各サイトに2つの異なる冗長スイッチペアをインストールできます。この場合、クラスタノードは、両方のローカルスイッチだけに接続されたデータセンター1のシャーシ、およびそれらのローカルスイッチに接続されたデータセンター2のシャーシではスパンド EtherChannel を使用しますが、スパンド EtherChannel は基本的に「分離」しています。各ローカル冗長スイッチは、スパンド EtherChannel をサイトローカルの EtherChannel として認識します。

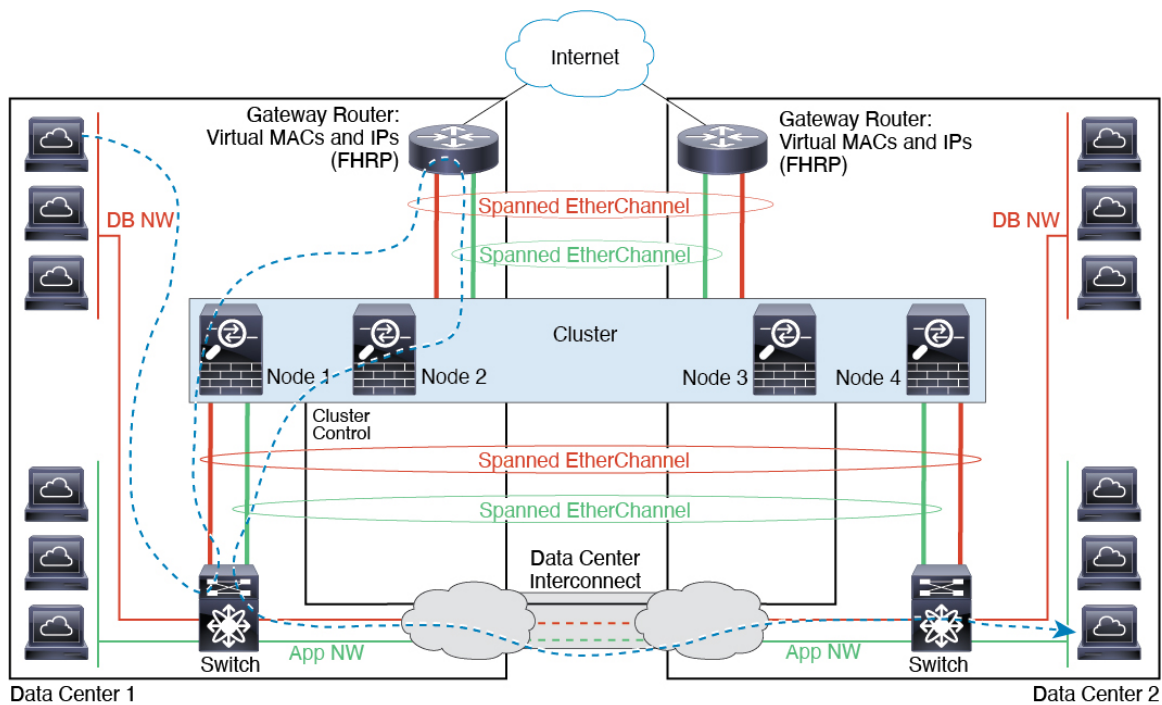


スパンド EtherChannel トランスペアレントモード イーストウェスト サイト間の例

次の例では、各サイトのゲートウェイ ルータと 2 つの内部ネットワーク（アプリケーション ネットワークと DB ネットワーク）間に配置された（イーストウェスト挿入）2 つのデータセンターのそれぞれに 2 つのクラスタ メンバーがある場合を示します。クラスタ メンバーは、DCI 経由のクラスタ制御リンクによって接続されています。各サイトのクラスタ メンバーは、内部および外部のアプリケーション ネットワークと DB ネットワークの両方にスパンド EtherChannels を使用してローカルスイッチに接続します。各 EtherChannel は、クラスタ内のすべてのシャーンにスパンされます。

各サイトのゲートウェイ ルータは、HSRP などの FHRP を使用して、各サイトで同じ宛先の仮想 MAC アドレスと IP アドレスを提供します。MAC アドレスの予期せぬフラッピングを避けるため、`mac-address-table static outside_interface mac_address` コマンドを使用して、ゲートウェイ ルータの実際の MAC アドレスを ASA MAC アドレステーブルに静的に追加することをお勧めします。これらのエントリがないと、サイト 1 のゲートウェイがサイト 2 のゲートウェイと通信する場合に、そのトラフィックが ASA を通過して、内部インターフェイスからサイト 2 に到達しようとして、問題が発生する可能性があります。データ VLAN は、オーバーレイ トランスポート 仮想化 (OTV)（または同様のもの）を使用してサイト間に拡張されます。トラフィックがゲートウェイ ルータ宛てである場合にトラフィックが DCI を通過して他のサイトに送信されないようにするには、フィルタを追加する必要があります。1 つのサイトのゲート

ウェルルータが到達不能になった場合、トラフィックが他のサイトのゲートウェイに送信されるようにフィルタを削除する必要があります。



クラスタリングの参考資料

このセクションには、クラスタリングの動作に関する詳細情報が含まれます。

ASA の各機能とクラスタリング

ASA の一部の機能は ASA クラスタリングではサポートされず、一部の機能は制御ノードだけでサポートされます。その他の機能については適切な使用に関する警告があります。

クラスタリングでサポートされない機能

次の各機能は、クラスタリングが有効なときは設定できず、コマンドは拒否されます。

- TLS プロキシを使用するユニファイド コミュニケーション機能
- リモート アクセス VPN (SSL VPN および IPSec VPN)
- 仮想トンネルインターフェイス (VTI)
- 次のアプリケーション インспекション：
 - CTIQBE

- H323、H225、および RAS
 - IPsec パススルー
 - MGCP
 - MMP
 - RTSP
 - SCCP (Skinny)
 - WAAS
 - WCCP
-
- ボットネット トラフィック フィルタ
 - Auto Update Server
 - DHCP クライアント、サーバー、およびプロキシ。DHCP リレーはサポートされていません。
 - VPN ロード バランシング
 - フェールオーバー
 - 統合ルーティングおよびブリッジング
 - FIPS モード

クラスタリングの中央集中型機能

次の機能は、制御ノード上だけでサポートされます。クラスタの場合もスケーリングされません。



- (注) 中央集中型機能のトラフィックは、クラスタ制御リンク経由でメンバーノードから制御ノードに転送されます。

再分散機能を使用する場合は、中央集中型機能のトラフィックが中央集中型機能として分類される前に再分散が行われて、制御ノード以外のノードに転送されることがあります。この場合は、トラフィックが制御ノードに送り返されます。

中央集中型機能については、制御ノードで障害が発生するとすべての接続がドロップされるので、新しい制御ノード上で接続を再確立する必要があります。

- 次のアプリケーション インспекション：
 - DCERPC
 - ESMTP
 - IM

- NetBIOS
 - PPTP
 - RADIUS
 - RSH
 - SNMP
 - SQLNET
 - SUNRPC
 - TFTP
 - XDMCP
- スタティック ルート モニタリング
 - ネットワーク アクセスの認証および許可。アカウントリングは非集中型です。
 - フィルタリング サービス
 - サイト間 VPN
 - IGMP マルチキャスト コントロール プレーン プロトコル処理（データ プレーン転送はクラスタ全体に分散されます）
 - PIM マルチキャスト コントロール プレーン プロトコル処理（データ プレーン転送はクラスタ全体に分散されます）
 - ダイナミックルーティング

個々のノードに適用される機能

これらの機能は、クラスタ全体または制御ノードではなく、各 ASA ノードに適用されます。

- QoS : QoS ポリシーは、コンフィギュレーション複製の一部としてクラスタ全体で同期されます。ただし、ポリシーは各ノードに個別に適用されます。たとえば、出力に対してポリシーを設定する場合は、適合レートおよび適合バースト値は、特定の ASA から出て行くトラフィックに適用されます。3 ノードから成るクラスタがあり、トラフィックが均等に分散している場合、適合レートは実際にクラスタのレートの3倍になります。
- 脅威検出 : 脅威検出はノードごとに個別に機能します。たとえば、上位統計情報はノード固有です。たとえば、ポートスキャン検出が機能しないのは、スキャントラフィックが全ノード間でロードバランシングされ、1つのノードですべてのトラフィックを確認できないためです。
- リソース管理 : マルチコンテキストモードでのリソース管理は、ローカル使用状況に基づいて各ノードに個別に適用されます。
- LISP トラフィック : UDP ポート 4342 上の LISP トラフィックは、各受信ノードによって検査されますが、ディレクタは割り当てられません。各ノードは、クラスタ間で共有され

る EID テーブルに追加されますが、LISP トラフィック自体はクラスタ状態の共有に参加しません。

ネットワーク アクセス用の AAA とクラスタリング

ネットワーク アクセス用の AAA は、認証、許可、アカウントिंगの 3 つのコンポーネントで構成されます。認証と許可は、クラスタリング制御ノード上で中央集中型機能として実装されており、データ構造がクラスタデータノードに複製されます。制御ノードが選択された場合、確立済みの認証済みユーザーおよびユーザーに関連付けられた許可を引き続き中断なく運用するために必要なすべての情報を新しい制御ノードが保有します。ユーザー認証のアイドルおよび絶対タイムアウトは、制御ノードが変更されたときも維持されます。

アカウントINGは、クラスタ内の分散型機能として実装されています。アカウントINGはフロー単位で実行されるため、フローに対するアカウントINGが設定されている場合、そのフローを所有するクラスタノードがアカウントING開始と停止のメッセージを AAA サーバーに送信します。

接続設定とクラスタリング

接続制限は、クラスタ全体に適用されます (**set connection conn-max**、**set connection embryonic-conn-max**、**set connection per-client-embryonic-max** および **set connection per-client-max** コマンドページを参照)。各ノードには、ブロードキャストメッセージに基づくクラスタ全体のカウンタの推定値があります。クラスタ全体で接続制限を設定しても、効率性を考慮して、厳密に制限数で適用されない場合があります。各ノードでは、任意の時点でのクラスタ全体のカウンタ値が過大評価または過小評価される可能性があります。ただし、ロードバランシングされたクラスタでは、時間の経過とともに情報が更新されます。

FTP とクラスタリング

- FTPD チャンネルとコントロールチャンネルのフローがそれぞれ別のクラスタメンバーによって所有されている場合は、D チャンネルのオーナーは定期的にアイドルタイムアウトアップデートをコントロールチャンネルのオーナーに送信し、アイドルタイムアウト値を更新します。ただし、コントロールフローのオーナーがリロードされて、コントロールフローが再ホスティングされた場合は、親子フロー関係は維持されなくなります。したがって、コントロールフローのアイドルタイムアウトは更新されません。
- FTP アクセスに AAA を使用する場合、制御チャンネルのフローは制御ノードに集中されません。

ICMP インспекションとクラスタリング

クラスタを通過する ICMP および ICMP エラーパケットのフローは、ICMP/ICMP エラーインспекションが有効かどうかによって異なります。ICMP インспекションを使用しない場合、ICMP は一方向のフローであり、ディレクタフローはサポートされません。ICMP インспекションを使用する場合、ICMP フローは双方向になり、ディレクタ/バックアップフローによってバックアップされます。検査された ICMP フローの違いの 1 つは、転送されたパケットの

ディレクタ処理にあります。ディレクタは、パケットをフォワーダに返す代わりに、フローオーナーに ICMP エコー応答パケットを転送します。

マルチキャスト ルーティングとクラスタリング

マルチキャスト ルーティングは、インターフェイス モードによって動作が異なります。

スパンド EtherChannel モードでのマルチキャストルーティング

スパンド EtherChannel モードでは、ファストパス転送が確立されるまで、制御ユニットがすべてのマルチキャスト ルーティング パケットとデータパケットを処理します。接続が確立された後は、各データユニットがマルチキャスト データ パケットを転送できます。

個別インターフェイス モードでのマルチキャストルーティング

個別インターフェイスモードでは、マルチキャストに関してユニットが個別に動作することはありません。データおよびルーティングのパケットはすべて制御ユニットで処理されて転送されるので、パケットレプリケーションが回避されます。

NAT とクラスタリング

NAT は、クラスタの全体的なスループットに影響を与えることがあります。インバウンドおよびアウトバウンドの NAT パケットが、それぞれクラスタ内の別の ASA に送信されることがあります。ロード バランシング アルゴリズムは IP アドレスとポートに依存していますが、NAT が使用される場合は、インバウンドとアウトバウンドとで、パケットの IP アドレスやポートが異なるからです。NAT オーナーではない ASA に到着したパケットは、クラスタ制御リンクを介してオーナーに転送されるため、クラスタ制御リンクに大量のトラフィックが発生します。NAT オーナーは、セキュリティおよびポリシーチェックの結果に応じてパケットの接続を作成できない可能性があるため、受信側ノードは、オーナーへの転送フローを作成しないことに注意してください。

それでもクラスタリングで NAT を使用する場合は、次のガイドラインを考慮してください。

- ポート ブロック割り当てによる PAT : この機能については、次のガイドラインを参照してください。
 - ホストあたりの最大制限は、クラスタ全体の制限ではなく、ノードごとに個別に適用されます。したがって、ホストあたりの最大制限が 1 に設定されている 3 ノードクラスタでは、ホストからのトラフィックが 3 つのノードすべてにロードバランシングされている場合、3 つのブロックを各ノードに 1 つずつ割り当てることができます。
 - バックアッププールからバックアップノードで作成されたポートブロックは、ホストあたりの最大制限の適用時には考慮されません。
 - PAT プールが完全に新しい IP アドレスの範囲で変更される On-the-fly PAT ルールの変更では、新しいプールが有効になっていてもまだ送信中の xlate バックアップ要求に対する xlate バックアップの作成が失敗します。この動作はポートのブロック割り当て機能に固有なものではなく、プールが分散されトラフィックがクラスタノード間でロードバランシングされるクラスタ展開でのみ見られる一時的な PAT プールの問題です。

- クラスタで動作している場合、ブロック割り当てサイズを変更することはできません。新しいサイズは、クラスタ内の各デバイスをリロードした後のみ有効になります。各デバイスのリロードの必要性を回避するために、すべてのブロック割り当てルールを削除し、それらのルールに関連するすべての `xlate` をクリアすることをお勧めします。その後、ブロックサイズを変更し、ブロック割り当てルールを再作成できます。
- ダイナミック PAT の NAT プールアドレス配布：PAT プールを設定すると、クラスタはプール内の各 IP アドレスをポートブロックに分割します。デフォルトでは、各ブロックは 512 ポートですが、ポートブロック割り当てルールを設定すると、代わりにユーザのブロック設定が使用されます。これらのブロックはクラスタ内のノード間で均等に分散されるため、各ノードには PAT プール内の IP アドレスごとに 1 つ以上のブロックがあります。したがって、想定される PAT 接続数に対して十分である場合には、クラスタの PAT プールに含める IP アドレスを 1 つだけにすることができます。PAT プールの NAT ルールで予約済みポート 1 ~ 1023 を含めるようにオプションを設定しない限り、ポートブロックは 1024 ~ 65535 のポート範囲をカバーします。
- 複数のルールにおける PAT プールの再利用：複数のルールで同じ PAT プールを使用するには、ルールにおけるインターフェイスの選択に注意を払う必要があります。すべてのルールで特定のインターフェイスを使用するか、あるいはすべてのルールで「任意」のインターフェイスを使用するか、いずれかを選択する必要があります。ルール全般にわたって特定のインターフェイスと「任意」のインターフェイスを混在させることはできません。混在させると、システムがリターントラフィックとクラスタ内の適切なノードを一致させることができなくなる場合があります。ルールごとに固有の PAT プールを使用することは、最も信頼性の高いオプションです。
- ラウンドロビンなし：PAT プールのラウンドロビンは、クラスタリングではサポートされません。
- 拡張 PAT なし：拡張 PAT はクラスタリングでサポートされません。
- 制御ノードによって管理されるダイナミック NAT `xlate`：制御ノードが `xlate` テーブルを維持し、データノードに複製します。ダイナミック NAT を必要とする接続をデータノードが受信したときに、その `xlate` がテーブル内がない場合、データノードは制御ノードに `xlate` を要求します。データノードが接続を所有します。
- 旧式の `xlates`：接続所有者の `xlate` アイドル時間が更新されません。したがって、アイドル時間がアイドルタイムアウトを超える可能性があります。 `refcnt` が 0 で、アイドルタイマー値が設定されたタイムアウトより大きい場合は、旧式の `xlate` であることを示します。
- `per-session` PAT 機能：クラスタリングに限りませんが、`per-session` PAT 機能によって PAT の拡張性が向上します。クラスタリングの場合は、各データノードが独自の PAT 接続を持てます。対照的に、`multi-session` PAT 接続は制御ノードに転送する必要があり、制御ノードがオーナーとなります。デフォルトでは、すべての TCP トラフィックおよび UDP DNS トラフィックは `per-session` PAT `xlate` を使用します。これに対し、ICMP および他のすべての UDP トラフィックは `multi-session` を使用します。TCP および UDP に対しこれらのデフォルトを変更するように `per-session` NAT ルールを設定できますが、ICMP に `per-session` PAT を設定することはできません。H.323、SIP、または Skinny などの `multi-session` PAT

のメリットを活用できるトラフィックでは、関連付けられている TCP ポートに対し per-session PAT を無効にできます（それらの H.323 および SIP の UDP ポートはデフォルトですでに multi-session になっています）。per-session PAT の詳細については、ファイアウォールの設定ガイドを参照してください。

- 次のインスペクション用のスタティック PAT はありません。
 - FTP
 - PPTP
 - RSH
 - SQLNET
 - TFTP
 - XDMCP
 - SIP

- 1 万を超える非常に多くの NAT ルールがある場合は、デバイスの CLI で **asp rule-engine transactional-commit nat** コマンドを使用してトランザクション コミット モデルを有効にする必要があります。有効にしないと、ノードがクラスタに参加できない可能性があります。

ダイナミックルーティングおよびクラスタリング

ここでは、クラスタリングでダイナミックルーティングを使用する方法について説明します。

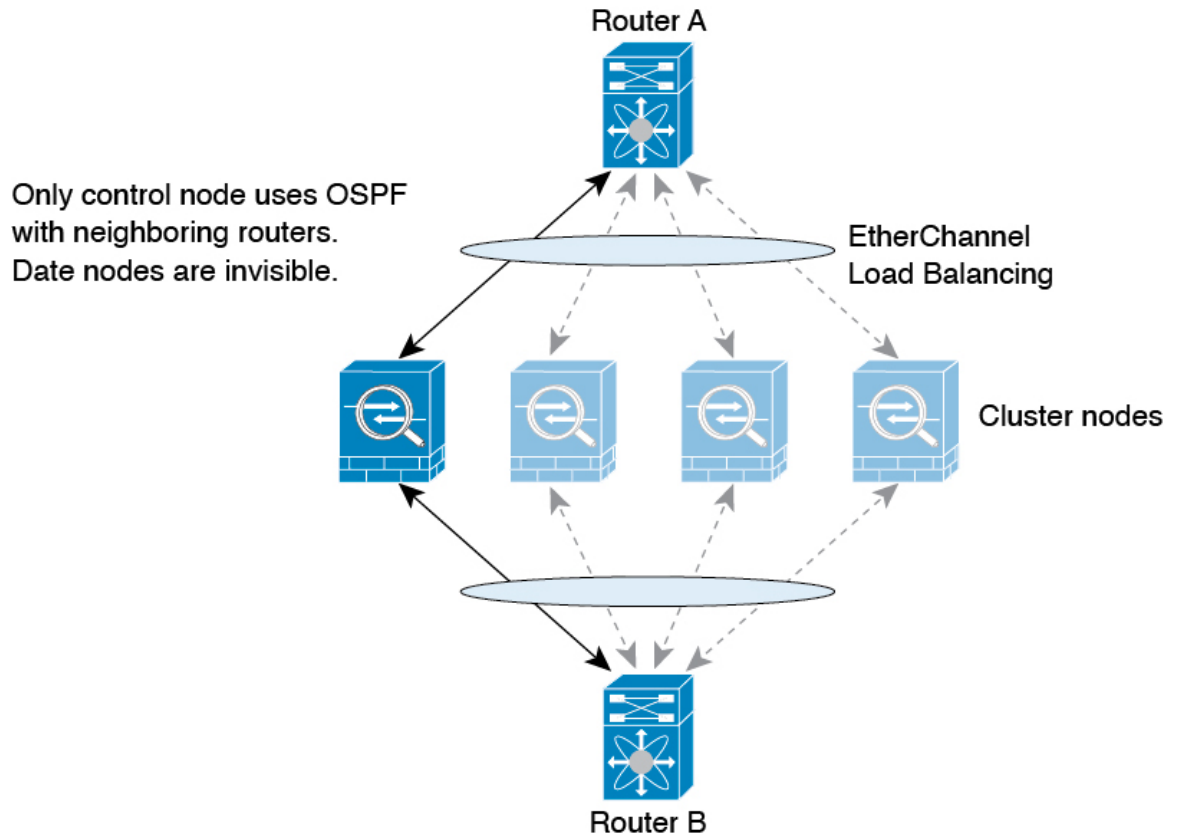
スパンド EtherChannel モードでのダイナミックルーティング



(注) IS-IS は、スパンド EtherChannel モードではサポートされていません。

スパンド EtherChannel モード：ルーティングプロセスは制御ノードでのみ実行されます。ルートは制御ノードを介して学習され、データノードに複製されます。ルーティングパケットは、データノードに到着すると制御ノードにリダイレクトされます。

図 1: スパンド EtherChannel モードでのダイナミック ルーティング



データノードが制御ノードからルートを学習すると、各ノードが個別に転送の判断を行います。

OSPF LSA データベースは、制御ノードからデータノードに同期されません。制御ノードのスイッチオーバーが発生した場合、ネイバールータが再起動を検出します。スイッチオーバーは透過的ではありません。OSPF プロセスが IP アドレスの 1 つをルータ ID として選択します。必須ではありませんが、スタティック ルータ ID を割り当てることができます。これで、同じルータ ID がクラスタ全体で使用されるようになります。割り込みを解決するには、OSPF ノンストップ フォワーディング 機能を参照してください。

SCTP とクラスタリング

SCTP アソシエーションは、（ロードバランシングにより）任意のノードに作成できますが、マルチホーミング接続は同じノードに存在する必要があります。

SIP インспекションとクラスタリング

制御フローは、（ロードバランシングにより）任意のノードに作成できますが、子データフローは同じノードに存在する必要があります。

TLS プロキシ設定はサポートされていません。

SNMP とクラスタリング

SNMP エージェントは、個々の ASA を、その 診断インターフェイスのローカル IP アドレスによってポーリングします。クラスタの統合データをポーリングすることはできません。

SNMP ポーリングには、メインクラスタ IP アドレスではなく、常にローカルアドレスを使用してください。SNMP エージェントがメインクラスタ IP アドレスをポーリングする場合、新しい制御ノードが選択されると、新しい制御ノードのポーリングは失敗します。

クラスタリングで SNMPv3 を使用している場合、最初のクラスタ形成後に新しいクラスタノードを追加すると、SNMPv3 ユーザーは新しいノードに複製されません。SNMPv3 ユーザーは、制御ノードに再追加して、新しいノードに強制的に複製するようにするか、データノードに直接追加する必要があります。

STUN とクラスタリング

ピンホールが複製される時、STUN インспекションはフェールオーバー モードとクラスタモードでサポートされます。ただし、トランザクション ID はノード間で複製されません。STUN 要求の受信後にノードに障害が発生し、別のノードが STUN 応答を受信した場合、STUN 応答はドロップされます。

syslog および NetFlow とクラスタリング

- **Syslog** : クラスタの各ノードは自身の syslog メッセージを生成します。ロギングを設定して、各ノードの syslog メッセージヘッダー フィールドで同じデバイス ID を使用するか、別の ID を使用するかを設定できます。たとえば、ホスト名設定はクラスタ内のすべてのノードに複製されて共有されます。ホスト名をデバイス ID として使用するようロギングを設定した場合、すべてのノードで生成される syslog メッセージが1つのノードから生成されているように見えます。クラスタブートストラップ設定で割り当てられたローカルノード名をデバイス ID として使用するようロギングを設定した場合、syslog メッセージはそれぞれ別のノードから生成されているように見えます。
- **NetFlow** : クラスタの各ノードは自身の NetFlow ストリームを生成します。NetFlow コレクタは、各 ASA を独立した NetFlow エクスポートとしてのみ扱うことができます。

Cisco TrustSec とクラスタリング

制御ノードだけがセキュリティグループタグ (SGT) 情報を学習します。その後、制御ノードからデータノードに SGT が渡されるため、データノードは、セキュリティポリシーに基づいて SGT の一致を判断できます。

VPN とクラスタリング

サイト間 VPN は、中央集中型機能です。制御ノードのみが VPN 接続をサポートします。



(注) リモート アクセス VPN は、クラスタリングではサポートされません。

VPN 機能を使用できるのは制御ノードだけであり、クラスタの高可用性機能は活用されません。制御ノードで障害が発生した場合は、すべての既存の VPN 接続が失われ、VPN ユーザにとってはサービスの中断となります。新しい制御ノードが選定されたときに、VPN 接続を再確立する必要があります。

VPN トンネルをスパンド EtherChannel アドレスに接続すると、接続が自動的に制御ノードに転送されます。

VPN 関連のキーと証明書は、すべてのノードに複製されます。

パフォーマンス スケーリング係数

複数のユニットをクラスタに結合すると、期待できる合計クラスタパフォーマンスは、最大合計スループットの約 80% になります。

たとえば、モデルが単独稼働で約 10 Gbps のトラフィックを処理できる場合、8 ユニットのクラスタでは、最大合計スループットは 80 Gbps (8 ユニット x 10 Gbps) の約 80% で 64 Gbps になります。

制御ノードの選定

クラスタのノードは、クラスタ制御リンクを介して通信して制御ノードを選定します。方法は次のとおりです。

1. ノードに対してクラスタリングをイネーブルにしたとき（または、クラスタリングがイネーブル済みの状態でそのユニットを初めて起動したとき）に、そのノードは選定要求を 3 秒間隔でブロードキャストします。
2. プライオリティの高い他のノードがこの選定要求に応答します。プライオリティは 1 ~ 100 の範囲内で設定され、1 が最高のプライオリティです。
3. 45 秒経過しても、プライオリティの高い他のノードからの応答を受信していない場合は、そのノードが制御ノードになります。



(注) 最高のプライオリティを持つノードが複数ある場合は、クラスタノード名、次にシリアル番号を使用して制御ノードが決定されます。

4. 後からクラスタに参加したノードのプライオリティの方が高い場合でも、そのノードが自動的に制御ノードになることはありません。既存の制御ノードは常に制御ノードのままです。ただし、制御ノードが応答を停止すると、その時点で新しい制御ノードが選定されます。
5. 「スプリットブレイン」シナリオで一時的に複数の制御ノードが存在する場合、優先順位が最も高いノードが制御ノードの役割を保持し、他のノードはデータノードの役割に戻ります。



- (注) ノードを手動で強制的に制御ノードにすることができます。中央集中型機能については、制御ノード変更を強制するとすべての接続がドロップされるので、新しい制御ノード上で接続を再確立する必要があります。

クラスタ内のハイアベイラビリティ

クラスタリングは、ノードとインターフェイスの正常性をモニターし、ノード間で接続状態を複製することにより、ハイアベイラビリティを実現します。

ノードヘルスマニタリング

各ノードは、クラスタ制御リンクを介してブロードキャストハートビートパケットを定期的に送信します。設定可能なタイムアウト期間内にデータノードからハートビートパケットまたはその他のパケットを受信しない場合、制御ノードはクラスタからデータノードを削除します。データノードが制御ノードからパケットを受信しない場合、残りのノードから新しい制御ノードが選択されます。

ノードで実際に障害が発生したためではなく、ネットワークの障害が原因で、ノードがクラスタ制御リンクを介して相互に通信できない場合、クラスタは「スプリットブレイン」シナリオに移行する可能性があります。このシナリオでは、分離されたデータノードが独自の制御ノードを選択します。たとえば、2つのクラスタロケーション間でルータに障害が発生した場合、ロケーション1の元の制御ノードは、ロケーション2のデータノードをクラスタから削除します。一方、ロケーション2のノードは、独自の制御ノードを選択し、独自のクラスタを形成します。このシナリオでは、非対称トラフィックが失敗する可能性があることに注意してください。クラスタ制御リンクが復元されると、より優先順位の高い制御ノードが制御ノードの役割を保持します。

詳細については、[制御ノードの選定 \(101 ページ\)](#) を参照してください。

インターフェイスモニタリング

各ノードは、使用中のすべての指名されたハードウェアインターフェイスのリンクステータスをモニターし、ステータス変更を制御ノードに報告します。

- スパンド EtherChannel : クラスタ Link Aggregation Control Protocol (cLACP) を使用します。各ノードは、リンクステータスおよび cLACP プロトコルメッセージをモニターして、ポートがまだ EtherChannel でアクティブであるかどうかを判断します。ステータスが制御ノードに報告されます。

ヘルスマニタリングをイネーブルにすると、すべての物理インターフェイス（主要な EtherChannel インターフェイスを含む）がデフォルトでモニターされるため、オプションでインターフェイスごとのモニタリングをディセーブルにすることができます。指名されたインターフェイスのみモニターできます。たとえば、指名された EtherChannel に障害が発生したと判断される必要がある場合、つまり、EtherChannel のすべてのメンバーポートはクラスタ削除をトリガーすることに失敗する必要があります（最小ポートバンドリング設定に応じて）。

ノードのモニタ対象のインターフェイスが失敗した場合、そのノードはクラスタから削除されます。ASAがメンバーをクラスタから削除するまでの時間は、そのノードが確立済みメンバーであるかクラスタに参加しようとしているかによって異なります。確立済みメンバーのインターフェイスがダウン状態の場合、ASAはそのメンバーを9秒後に削除します。ASAは、ノードがクラスタに参加する最初の90秒間はインターフェイスを監視しません。この間にインターフェイスのステータスが変化しても、ASAはクラスタから削除されません。

障害後のステータス

クラスタ内のノードで障害が発生したときに、そのノードでホストされている接続は他のノードにシームレスに移行されます。トラフィックフローのステート情報は、制御ノードのクラスタ制御リンクを介して共有されます。

制御ノードで障害が発生した場合、そのクラスタの他のメンバーのうち、優先順位が最高（番号が最小）のメンバーが制御ノードになります。

障害イベントに応じて、ASAは自動的にクラスタへの再参加を試みます。



- (注) ASAが非アクティブになり、クラスタへの自動再参加に失敗すると、すべてのデータインターフェイスがシャットダウンされ、管理専用インターフェイスのみがトラフィックを送受信できます。管理インターフェイスは、そのノードがクラスタIPプールから受け取ったIPアドレスを使用して引き続き稼働状態となります。ただし、リロードする場合、クラスタでノードがまだ非アクティブになっていると、管理インターフェイスは無効になります。さらに設定を行う場合は、コンソールポートを使用する必要があります。

クラスタへの再参加

クラスタノードがクラスタから削除された後、クラスタに再参加するための方法は、削除された理由によって異なります。

- クラスタ制御リンクの障害：（最初の参加時）クラスタ制御リンクの問題を解決した後、コンソールポートで **cluster group name** と入力してから **enable** と入力して、クラスタリングを再びイネーブルにすることによって、手動でクラスタに再参加する必要があります。
- クラスタに参加した後に障害が発生したクラスタ制御リンク：ASAは、無限に5分ごとに自動的に再参加を試みます。この動作は設定可能です。
- データインターフェイスの障害：ASAは自動的に最初は5分後、次に10分後、最終的に20分後に再参加を試みます。20分後に参加できない場合、ASAはクラスタリングをディセーブルにします。データインターフェイスの問題を解決した後、コンソールポートで **cluster group name** と入力してから **enable** と入力して、クラスタリングを手動でイネーブルにする必要があります。この動作は設定可能です。
- ノードの障害：ノードがヘルスチェック失敗のためクラスタから削除された場合、クラスタへの再参加は失敗の原因によって異なります。たとえば、一時的な電源障害の場合は、クラスタ制御リンクが稼働していて、クラスタリングが **enable** コマンドでまだイネーブ

ルになっているなら、ノードは再起動するとクラスタに再参加することを意味します。ASA は 5 秒ごとにクラスタへの再参加を試みます。

- 内部エラー：内部の障害には、アプリケーション同期のタイムアウト、矛盾したアプリケーション ステータスなどがあります。ノードは、5 分、10 分、20 分の間隔で自動的にクラスタに再参加しようとします。この動作は設定可能です。

「[制御ノードのブートストラップの設定 \(32 ページ\)](#)」を参照してください。

データ パス接続状態の複製

どの接続にも、1つのオーナーおよび少なくとも1つのバックアップオーナーがクラスタ内にあります。バックアップオーナーは、障害が発生しても接続を引き継ぎません。代わりに、TCP/UDP のステート情報を保存します。これは、障害発生時に接続が新しいオーナーにシームレスに移管されるようにするためです。バックアップオーナーは通常ディレクタでもありません。

トラフィックの中には、TCP または UDP レイヤよりも上のステート情報を必要とするものがあります。この種類のトラフィックに対するクラスタリングのサポートの可否については、次の表を参照してください。

表 1: クラスタ全体で複製される機能

トラフィック	状態のサポート	注
アップタイム	対応	システム アップタイムをトラッキングします。
ARP テーブル	対応	—
MAC アドレス テーブル	対応	—
ユーザ アイデンティティ	対応	AAA ルール (uauth) が含まれます。
IPv6 ネイバー データベース	対応	—
ダイナミック ルーティング	対応	—
SNMP エンジン ID	なし	—
Firepower 4100/9300 の分散型 VPN (サイト間)	対応	バックアップセッションがアクティブセッションになると、新しいバックアップセッションが作成されます。

クラスタが接続を管理する方法

接続をクラスタの複数のノードにロードバランシングできます。接続のロールにより、通常動作時とハイ アベイラビリティ状況時の接続の処理方法が決まります。

接続のロール

接続ごとに定義された次のロールを参照してください。

- **オーナー**：通常、最初に接続を受信するノード。オーナーは、TCP 状態を保持し、パケットを処理します。1 つの接続に対してオーナーは 1 つだけです。元のオーナーに障害が発生すると、新しいノードが接続からパケットを受信したときにディレクタがそれらのノードの新しいオーナーを選択します。
- **バックアップオーナー**：オーナーから受信した TCP/UDP ステート情報を格納するノード。障害が発生した場合、新しいオーナーにシームレスに接続を転送できます。バックアップオーナーは、障害発生時に接続を引き継ぎません。オーナーが使用不可能になった場合、（ロードバランシングに基づき）その接続からのパケットを受信する最初のノードがバックアップオーナーに問い合わせ、関連するステート情報を取得し、そのノードが新しいオーナーになります。

ディレクタ（下記参照）がオーナーと同じノードでない限り、ディレクタはバックアップオーナーでもあります。オーナーが自分をディレクタとして選択した場合は、別のバックアップオーナーが選択されます。

1 台のシャーシに最大 3 つのクラスタノードを搭載できる Firepower 9300 のクラスタリングでは、バックアップオーナーがオーナーと同じシャーシにある場合、シャーシ障害からフローを保護するために、別のシャーシから追加のバックアップオーナーが選択されます。

サイト間クラスタリングのディレクタローカリゼーションを有効にすると、ローカルバックアップとグローバルバックアップの 2 つのバックアップオーナー権限があります。オーナーは、常に同じサイトのローカルバックアップをオーナー自身として選択します（サイト ID に基づいて）。グローバルバックアップはどのサイトにも配置でき、ローカルバックアップと同一ノードとすることもできます。オーナーは、両方のバックアップへ接続ステート情報を送信します。

サイトの冗長性が有効になっており、バックアップオーナーがオーナーと同じサイトに配置されている場合は、サイトの障害からフローを保護するために、別のサイトから追加のバックアップオーナーが選択されます。シャーシバックアップとサイトバックアップは独立しているため、フローにはシャーシバックアップとサイトバックアップの両方が含まれている場合があります。

- **ディレクタ**：フォワーダからのオーナーバックアップ要求を処理するノード。オーナーは、新しい接続を受信すると、送信元/宛先 IP アドレスおよびポートのハッシュに基づいてディレクタを選択し、新しい接続を登録するためにそのディレクタにメッセージを送信します。パケットがオーナー以外のノードに到着した場合、そのノードはどのノードがオーナーかをディレクタに問い合わせることで、パケットを転送できます。1 つの接続に

対してディレクタは1つだけです。ディレクタが失敗すると、オーナーは新しいディレクタを選択します。

ディレクタがオーナーと同じノードでない限り、ディレクタはバックアップオーナーでもあります（上記参照）。オーナーがディレクタとして自分自身を選択すると、別のバックアップオーナーが選択されます。

サイト間クラスタリングのディレクタローカリゼーションを有効にすると、ローカルディレクタとグローバルディレクタの2つのディレクタ権限が区別されます。オーナーは、同一サイト（Site Idに基づき）のローカルディレクタとして、常にオーナー自身を選択します。グローバルディレクタはどのサイトにも配置でき、ローカルディレクタと同一ノードとすることもできます。最初のオーナーに障害が発生すると、ローカルディレクタは、同じサイトの新しい接続オーナーを選択します。

ICMP/ICMPv6 ハッシュの詳細：

- エコーパケットの場合、送信元ポートは ICMP 識別子で、宛先ポートは 0 です。
 - 応答パケットの場合、送信元ポートは 0 で、宛先ポートは ICMP 識別子です。
 - 他のパケットの場合、送信元ポートと宛先ポートの両方が 0 です。
- **フォワーダ**：パケットをオーナーに転送するノード。フォワーダが接続のパケットを受信したときに、その接続のオーナーが自分ではない場合は、フォワーダはディレクタにオーナーを問い合わせしてから、そのオーナーへのフローを確立します。これは、この接続に関してフォワーダが受信するその他のパケット用です。ディレクタは、フォワーダにもなることができます。ディレクタローカリゼーションを有効にすると、フォワーダは常にローカルディレクタに問い合わせを行います。フォワーダがグローバルディレクタに問い合わせを行うのは、ローカルディレクタがオーナーを認識していない場合だけです。たとえば、別のサイトで所有されている接続のパケットをクラスタメンバーが受信する場合などです。フォワーダが SYN-ACK パケットを受信した場合、フォワーダはパケットの SYN キーからオーナーを直接取得できるので、ディレクタに問い合わせる必要がないことに注意してください。（TCP シーケンスのランダム化を無効にした場合は、SYN Cookie は使用されないため、ディレクタへの問い合わせが必要です）。存続期間が短いフロー（たとえば DNS や ICMP）の場合は、フォワーダは問い合わせの代わりにパケットを即座にディレクタに送信し、ディレクタがそのパケットをオーナーに送信します。1つの接続に対して、複数のフォワーダが存在できます。最も効率的なスループットを実現できるのは、フォワーダが1つもなく、接続のすべてのパケットをオーナーが受信するという、優れたロードバランシング方法が使用されている場合です。



(注) クラスタリングを使用する場合は、TCP シーケンスのランダム化を無効にすることは推奨されません。SYN/ACK パケットがドロップされる可能性があるため、一部の TCP セッションが確立されない可能性があります。

- **フラグメントオーナー**：フラグメント化されたパケットの場合、フラグメントを受信するクラスタノードは、フラグメントの送信元と宛先の IP アドレス、およびパケット ID の

ハッシュを使用してフラグメントオーナーを特定します。その後、すべてのフラグメントがクラスタ制御リンクを介してフラグメント所有者に転送されます。スイッチのロードバランシングで使用される5タプルは、最初のフラグメントにのみ含まれているため、フラグメントが異なるクラスタノードにロードバランシングされる場合があります。他のフラグメントには、送信元ポートと宛先ポートは含まれず、他のクラスタノードにロードバランシングされる場合があります。フラグメント所有者は一時的にパケットを再アセンブルするため、送信元/宛先 IP アドレスとポートのハッシュに基づいてディレクタを決定できます。新しい接続の場合は、フラグメントの所有者が接続所有者として登録されます。これが既存の接続の場合、フラグメント所有者は、クラスタ制御リンクを介して、指定された接続所有者にすべてのフラグメントを転送します。その後、接続の所有者はすべてのフラグメントを再構築します。

接続でポートアドレス変換 (PAT) を使用すると、PAT のタイプ (per-session または multi-session) が、クラスタのどのメンバが新しい接続のオーナーになるかに影響します。

- per-session PAT : オーナーは、接続の最初のパケットを受信するノードです。

デフォルトでは、TCP および DNS UDP トラフィックは per-session PAT を使用します。

- multi-session PAT : オーナーは常に制御ノードです。multi-session PAT 接続がデータノードで最初に受信される場合、データノードがその接続を制御ノードに転送します。

デフォルトでは、UDP (DNS UDP を除く) および ICMP トラフィックは multi-session PAT を使用するため、それらの接続は常に制御ノードによって所有されています。

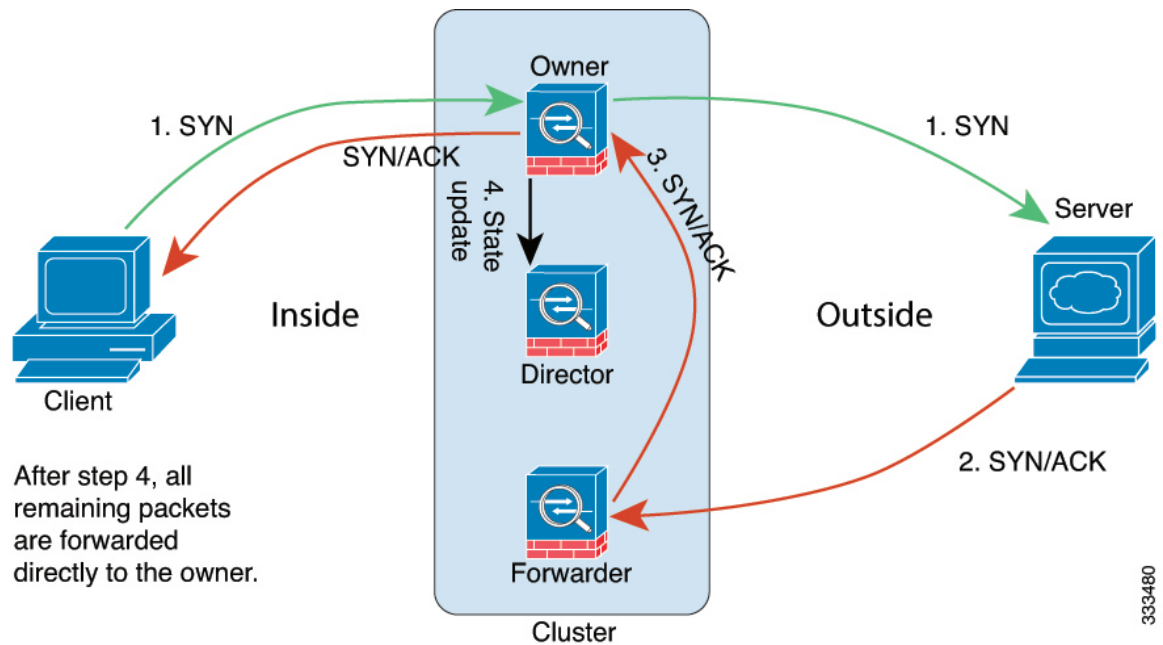
TCP および UDP の per-session PAT デフォルトを変更できるので、これらのプロトコルの接続は、その設定に応じて per-session または multi-session で処理されます。ICMP の場合は、デフォルトの multi-session PAT から変更することはできません。per-session PAT の詳細については、『ファイアウォールの構成ガイド』を参照してください。

新しい接続の所有権

新しい接続がロードバランシング経由でクラスタのノードに送信される場合は、そのノードがその接続の両方向のオーナーとなります。接続のパケットが別のノードに到着した場合は、そのパケットはクラスタ制御リンクを介してオーナーノードに転送されます。最適なパフォーマンスを得るには、適切な外部ロードバランシングが必要です。1つのフローの両方向が同じノードに到着するとともに、フローがノード間に均等に分散されるようにするためです。逆方向のフローが別のノードに到着した場合は、元のノードにリダイレクトされます。

TCP のサンプルデータフロー

次の例は、新しい接続の確立を示します。



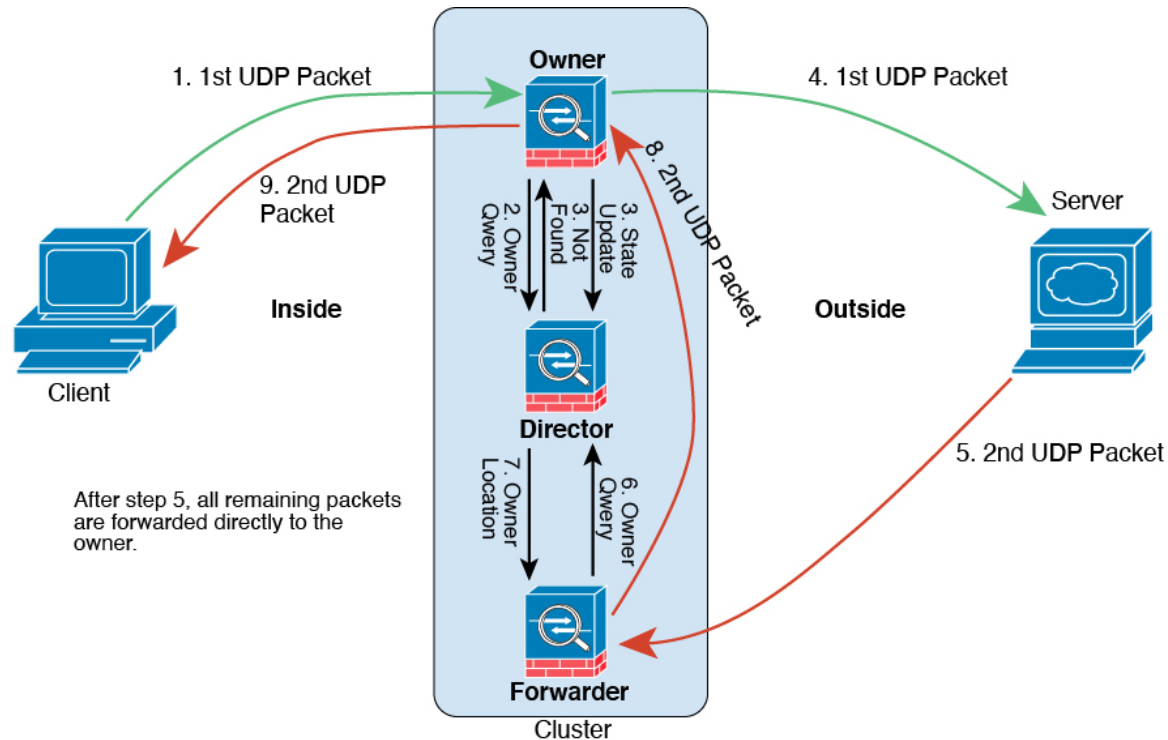
333480

1. SYN パケットがクライアントから発信され、ASA の 1 つ（ロードバランシング方法に基づく）に配信されます。これがオーナーとなります。オーナーはフローを作成し、オーナー情報をエンコードして SYN Cookie を生成し、パケットをサーバに転送します。
2. SYN-ACK パケットがサーバから発信され、別の ASA（ロードバランシング方法に基づく）に配信されます。この ASA はフォワーダです。
3. フォワーダはこの接続を所有してはいないので、オーナー情報を SYN Cookie からデコードし、オーナーへの転送フローを作成し、SYN-ACK をオーナーに転送します。
4. オーナーはディレクタに状態アップデートを送信し、SYN-ACK をクライアントに転送します。
5. ディレクタは状態アップデートをオーナーから受信し、オーナーへのフローを作成し、オーナーと同様に TCP 状態情報を記録します。ディレクタは、この接続のバックアップオーナーとしての役割を持ちます。
6. これ以降、フォワーダに配信されたパケットはすべて、オーナーに転送されます。
7. パケットがその他のノードに配信された場合、そのノードはディレクタに問い合わせ、オーナーを特定し、フローを確立します。
8. フローの状態が変化した場合、状態アップデートがオーナーからディレクタに送信されます。

ICMP および UDP のサンプルデータフロー

次の例は、新しい接続の確立を示します。

1. 図 2: ICMP および UDP データフロー



UDP パケットがクライアントから発信され、1つの ASA（ロードバランシング方法に基づく）に配信されます。

2. 最初のパケットを受信したノードは、送信元/宛先 IP アドレスとポートのハッシュに基づいて選択されたディレクタノードをクエリします。
3. ディレクタは既存のフローを検出せず、ディレクタフローを作成して、以前のノードにパケットを転送します。つまり、ディレクタがこのフローのオーナーを選択したことになります。
4. オーナーはフローを作成し、ディレクタに状態アップデートを送信して、サーバーにパケットを転送します。
5. 2 番目の UDP パケットはサーバーから発信され、フォワーダに配信されます。
6. フォワーダはディレクタに対して所有権情報をクエリします。存続期間が短いフロー（DNS など）の場合、フォワーダはクエリする代わりにパケットを即座にディレクタに送信し、ディレクタがそのパケットをオーナーに送信します。
7. ディレクタは所有権情報をフォワーダに返信します。
8. フォワーダは転送フローを作成してオーナー情報を記録し、パケットをオーナーに転送します。
9. オーナーはパケットをクライアントに転送します。

新しい TCP 接続のクラスタ全体での再分散

アップストリームまたはダウンストリームルータによるロードバランシングの結果として、フロー分散に偏りが生じた場合は、新しい TCP フローを過負荷のノードから他のノードにリダイレクトするように設定できます。既存のフローは他のノードには移動されません。

Secure Firewall 3100 の ASA クラスタリングの履歴

機能名	バージョン	機能情報
Secure Firewall 3100 でのクラスタリングのサポートが導入されました	9.17(1)	Spanned EtherChannel モードでは、最大 6 台の Secure Firewall 3100 ユニットのクラスタ化できます。

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。