



## CHAPTER 6

# Cisco IOS インサーブिस ソフトウェア アップグレード プロセスの設定



(注)

Cisco IOS 12.2(31) SGA 以降、ISSU は Catalyst 4500 でサポートされます。すべてのラインカードがサポートされています。

冗長システムで稼働している場合、インサーブिस ソフトウェア アップグレード (ISSU) プロセスにより、Cisco IOS ソフトウェアが更新または変更される間もパケットの転送が続行されます。ほとんどのネットワークでは、計画されたソフトウェア アップグレードがダウンタイムの大きな原因になっています。ISSU により、Cisco IOS ソフトウェアが変更される間、パケットの転送が続行されます。これにより、ネットワークの可用性が向上し、計画されたソフトウェア アップグレードによって発生するダウンタイムが抑えられます。ここでは、ISSU の概念について説明し、システムで ISSU を実行するための手順について説明します。

ここでは、次の内容について説明します。

- 「ISSU を実行するための前提条件」 (P.6-2)
- 「ISSU について」 (P.6-3)
- 「ISSU プロセスの実行」 (P.6-15)
- 「関連資料」 (P.6-42)



(注)

この章で使用するスイッチ コマンドの構文および使用方法の詳細については、次の URL で『Cisco Catalyst 4500 Series Switch Command Reference』と関連資料を参照してください。

<http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html>

『Catalyst 4500 Series Switch Command Reference』に掲載されていないコマンドについては、より詳細な Cisco IOS ライブラリを参照してください。次の URL で『Cisco IOS Command Reference』と関連資料を参照してください。

<http://www.cisco.com/en/US/products/ps6350/index.html>

## ISSU を実行するための前提条件

ISSU を実行する前に以下の前提条件を満たす必要があります。

- 既存のイメージとターゲット イメージのイメージ タイプは一致する必要があります。たとえば、数分のトラフィック損失の発生なしで、IP Base イメージから Enterprise Services イメージに（またはその逆に）アップグレードすることはできません。



(注) 同様の制限が、暗号イメージと非暗号イメージの間に適用されます。

- アクティブ スーパーバイザ エンジンおよびスタンバイ スーパーバイザ エンジンは、同じスーパーバイザ エンジン ハードウェア（同じモデル、メモリ、NFL ドータ カードなど）を備える必要があります。
- ISSU プロセスを開始する前に、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方のファイル システム（ブートフラッシュまたはコンパクト フラッシュ）に新規および古い Cisco IOS ソフトウェア イメージがロードされている必要があります。

古いイメージは、ブートフラッシュまたはコンパクト フラッシュのいずれかに格納されている必要があります。ISSU プロセスが展開される前にブート変数を変更するべきではないので、これらのいずれかのロケーションからシステムが起動されている必要があります。



(注) ISSU が正常に実行されるためには **auto-boot** をイネーブルにする必要があります。

- ステートフル スイッチオーバー (SSO) が設定されており、スタンバイ スーパーバイザ エンジンが STANDBY HOT ステートである必要があります。

**show module**、**show running-config**、**show redundancy state** コマンドを使用すると、SSO がイネーブルかどうかを確認できます。

SSO をイネーブルにしていない場合は、SSO をイネーブルにし、設定する方法の詳細について、『*Stateful Switchover*』を参照してください。

- NSF が設定されており、正常に稼働している必要があります。NSF をイネーブルにしていない場合は、NSF をイネーブルにし、設定する方法の詳細について、『*Cisco Nonstop Forwarding*』を参照してください。
- ISSU を実行する前に、システムが冗長モード SSO に設定されており、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方のファイル システムに新しい ISSU 互換イメージが含まれていることを確認します。現在システム上で動作している Cisco IOS バージョンも ISSU をサポートする必要があります。

Catalyst 4500 シリーズ スイッチにさまざまなコマンドを入力するか、Cisco Feature Navigator の ISSU アプリケーションによって、スーパーバイザ エンジンのバージョンングおよび Cisco IOS の互換性を判別できます。

- no ip routing** コマンドを入力すると ISSU は SSO モードから RPR モードにフォールバックするため、トラフィックの損失が生じます。
- 自動ブートがオンになり、現在のブート済みイメージは、BOOT 環境変数で指定されたものと一致します。これらの設定および確認方法の詳細については、「[ブート フィールドの変更および boot コマンドの使用](#)」(P.3-29) を参照してください。
- no ip routing** コマンドを入力すると ISSU は SSO モードから RPR モードにフォールバックするため、トラフィックの損失が生じます。

# ISSU について



(注)

ISSU の実行中は、ハードウェアに変更を加えないでください。

ISSU を実行する前に、次の概念を理解しておく必要があります。

- 「Stateful Switchover の概要」 (P.6-3)
- 「NSF の概要」 (P.6-5)
- 「ISSU プロセスの概要」 (P.6-6)
- 「ISSU アップグレードの実行 : 2 つの方法」 (P.6-11)
- 「changeversion プロセス」 (P.6-12)
- 「ISSU の実行に関するガイドライン」 (P.6-13)
- 「ISSU をサポートする Cisco IOS ソフトウェアのバージョンニング機能」 (P.6-13)
- 「ISSU に対する SNMP サポート」 (P.6-15)
- 「Cisco Feature Navigator を使用した互換性の検証」 (P.6-15)

## Stateful Switchover の概要

SSO 機能の展開は、Cisco IOS スイッチで構築されたネットワークの可用性を向上させる全体的なプログラムの 1 ステップです。

デュアル スーパーバイザ エンジンをサポートする特定のシスコ ネットワーキング デバイス上で、SSO はスーパーバイザ エンジンの冗長構成を活用してネットワークの可用性を向上させます。SSO は、スーパーバイザ エンジンの 1 つをアクティブ プロセッサ、もう一方をスタンバイ プロセッサとして設定することにより、これを実現します。2 つのスーパーバイザ エンジン間の初期同期後に、SSO は両方のスーパーバイザ エンジンのステート情報を動的にリアルタイムで同期化します。

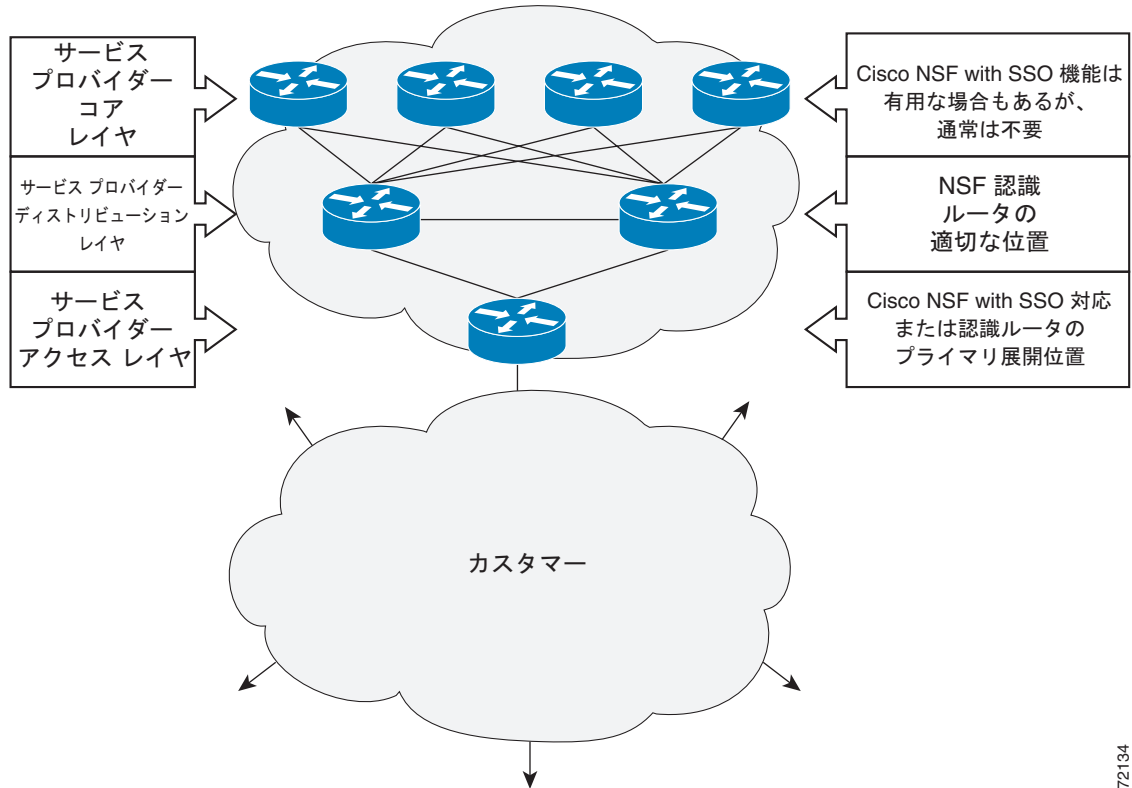
アクティブ スーパーバイザ エンジンが故障した場合、またはネットワーク デバイスから取り外された場合に、アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへのスイッチオーバーが行われます。

Cisco NSF は、SSO と併用します。Cisco NSF によって、スイッチオーバー後にルーティング プロトコル情報が復元される間、データ パケットの転送が既知のルートで続行されます。Cisco NSF を使用すると、ピア ネットワーキング デバイスでルーティング フラップが発生することがなくなるため、カスタマーに対するサービス停止を回避できます。

図 6-1 は、サービス プロバイダー ネットワークに SSO が展開される一般的な方法を示します。この例では、Cisco NSF/SSO がサービス プロバイダー ネットワークのアクセス レイヤ (エッジ) でイネーブルにされています。このポイントで障害が発生すると、サービス プロバイダー ネットワークへのアクセスが必要なエンタープライズ カスタマーのサービスを損なう可能性があります。

Cisco NSF プロトコルは、ネイバー デバイスが Cisco NSF に参加している必要があるため、それらのネイバー ディストリビューション レイヤ デバイスに Cisco NSF 対応のソフトウェア イメージをインストールする必要があります。目的に応じて、ネットワークのコア レイヤで Cisco NSF および SSO 機能を展開することもできます。これを行うと、特定の障害が発生した場合のネットワーク機能およびサービスの復元に必要な時間を短縮できるため、可用性が向上します。

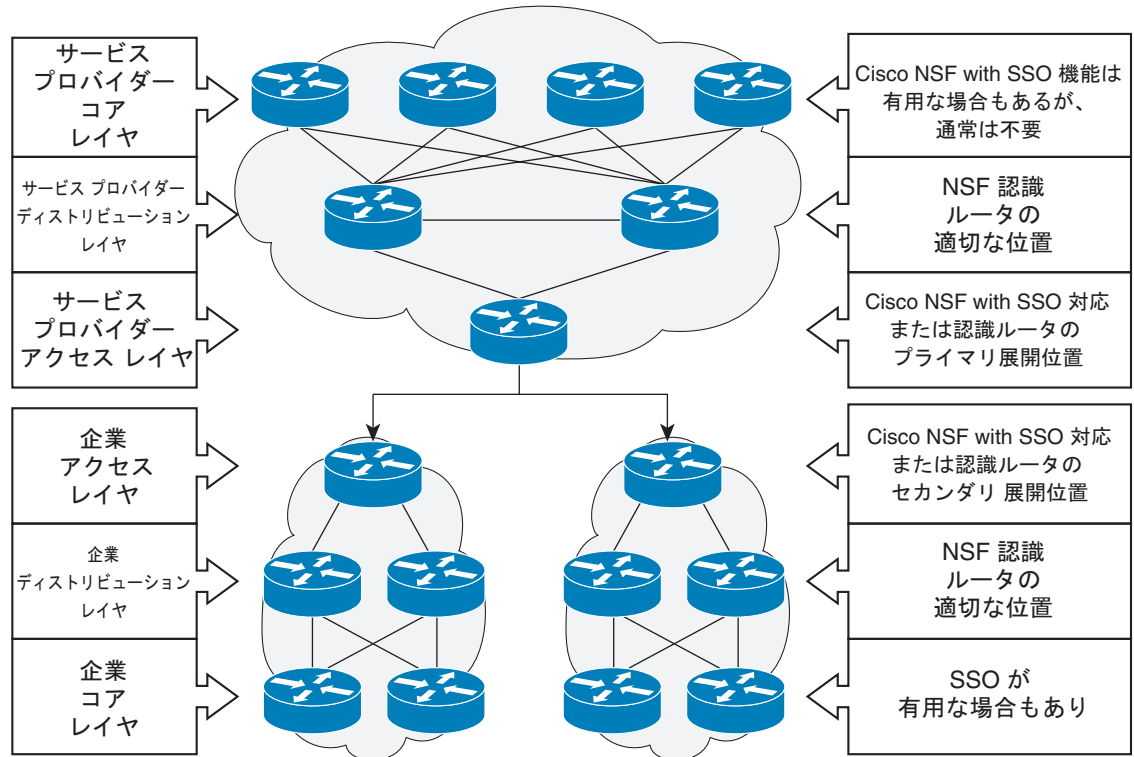
図 6-1 Cisco NSF/SSO ネットワーク構成 : サービス プロバイダー ネットワーク



72134

アベイラビリティの向上は、シングルポイント障害が存在するネットワーク内の他のポイントに Cisco NSF/SSO を展開することによって得られます。図 6-2 は、エンタープライズ ネットワーク アクセス レイヤに Cisco NSF/SSO を適用するもう 1 つの展開方法を示します。この例では、エンタープライズ ネットワーク内の各アクセス ポイントが、ネットワーク設計内の他のシングルポイント障害を表します。この例では、スイッチオーバーまたは計画されたソフトウェア アップグレードが行われても、エンタープライズ カスタマー セッションは中断することなくネットワーク内で稼働し続けます。

図 6-2 Cisco NSF/SSO ネットワーク構成 : エンタープライズ ネットワーク



## NSF の概要

Cisco NSF は、Cisco IOS ソフトウェアの SSO 機能と連動します。SSO は、Cisco NSF の前提条件です。NSF は、SSO と連動して、スイッチオーバー後にユーザがネットワークを使用できない時間を最小限に抑えます。Cisco NSF の主要目的は、スーパーバイザ エンジンのスイッチオーバー後も IP パケットの転送を継続させることです。

通常、ネットワークング デバイスが再起動すると、そのデバイスのすべてのルーティング ピアは、デバイスがダウンし、そのあと再びアップになったことを検知します。このような移行によって、いわゆるルーティング フラップが発生します。ルーティング フラップは、複数のルーティング ドメインに広がる場合があります。ルーティングの再起動によって発生したルーティング フラップによって、ルーティングが不安定になります。これはネットワーク全体のパフォーマンスに悪影響を及ぼします。Cisco NSF は、SSO 対応のデバイスにおけるルーティング フラップを抑止することによって、ネットワークの安定性を保ちます。

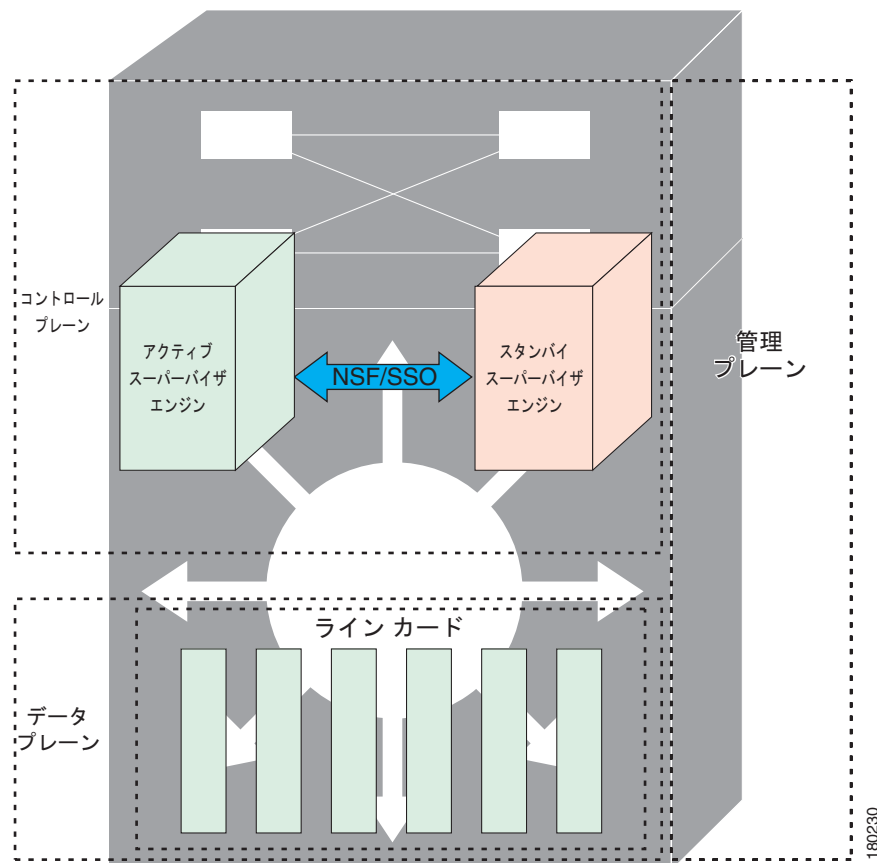
Cisco NSF によって、スイッチオーバー後にルーティング プロトコル情報が復元される間、データの packets の転送が既知のルートで続行されます。Cisco NSF を使用すると、ピア ネットワーク デバイスでルーティング フラップが発生することがありません。スイッチオーバー時に、故障したアクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンが制御を引き継ぐ間も、データトラフィックが転送されます。Cisco NSF 動作で重要なのは、スイッチオーバー時に物理リンクがアップの状態を維持できる点と、アクティブ スーパーバイザ エンジン上の Forwarding Information Base (FIB; 転送情報ベース) との同期性が保たれる点です。

## ISSU プロセスの概要

ISSU プロセスを使用すると、システムによるパケット転送を続行しながら、Cisco IOS ソフトウェアのアップグレードまたはダウングレードを実行できます。(ISSU プロセスの間に使用されるコマンドのインストラクションについては、[図 6-8 \(P.6-11\)](#) を参照してください)。Cisco IOS ISSU は Cisco IOS ハイ アベイラビリティ インフラストラクチャ (Cisco NSF/SSO およびハードウェアの冗長構成) を利用し、システムの稼働中に変更を行えるようにすることによって、ソフトウェア アップグレードまたはバージョン変更に伴うダウンタイムをなくします ([図 6-3](#) を参照)。

SSO/NSF モードは、アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへの設定とランタイム ステートの同期をサポートしています。このプロセスを発生させるには、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方のイメージが同じである必要があります。アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンのイメージが異なる場合は、その 2 つのバージョンの Cisco IOS が別々の機能とコマンドのセットをサポートしていても、ISSU によって 2 つのスーパーバイザ エンジンの同期を維持することができます。

図 6-3 ISSU プロセスでのハイ アベイラビリティ機能およびハードウェアの冗長構成

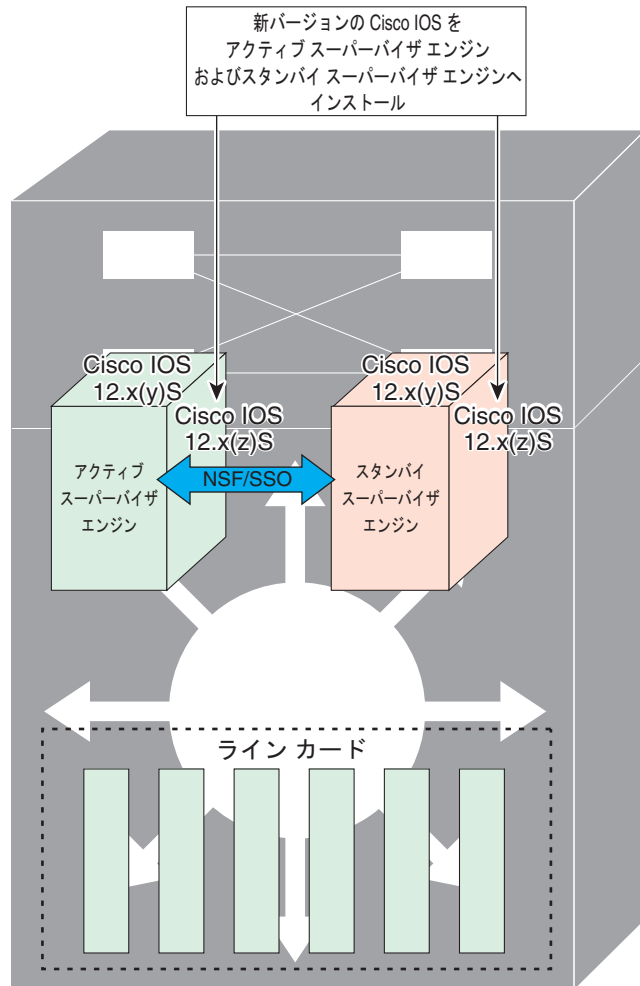


ISSU 対応スイッチは、2 つのスーパーバイザ エンジン（アクティブとスタンバイ）および 1 つまたは複数のラインカードで構成されています。ISSU プロセスを開始する前に、両方のスーパーバイザ エンジンのファイル システムに Cisco IOS ソフトウェアをコピーします（図 6-4 を参照）。



(注) 次の図では、Cisco IOS 12.x(y)S は、Cisco IOS の現行バージョンを表しています。

図 6-4 両方のスーパーバイザ エンジンへの新しい Cisco IOS ソフトウェア バージョンのインストールまたはコピー



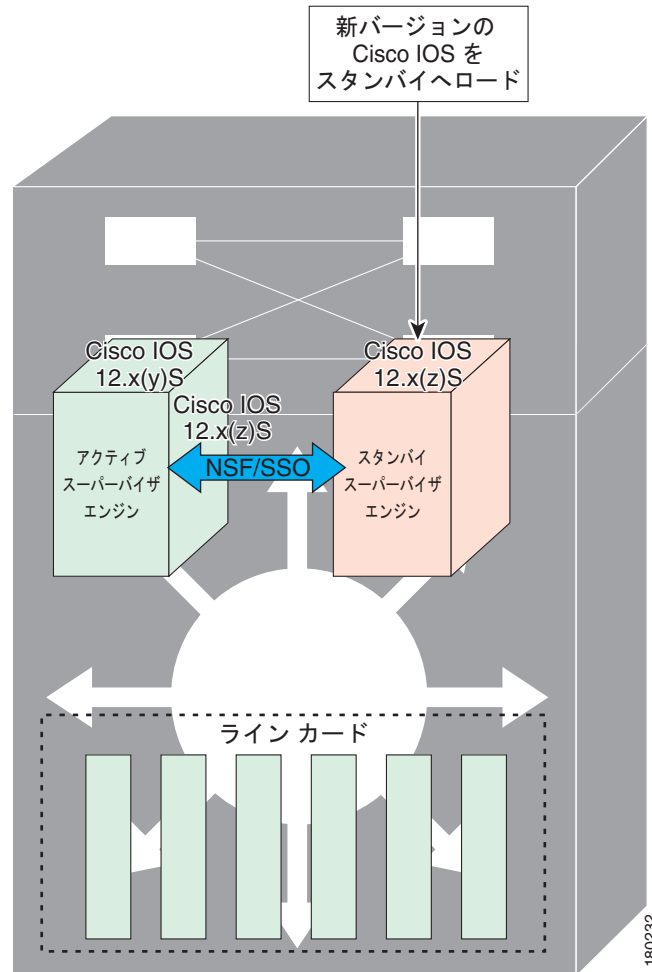
両ファイル システムに Cisco IOS ソフトウェアをコピーしたあと、新しい Cisco IOS ソフトウェア バージョンをスタンバイ スーパーバイザ エンジンにロードします (図 6-5 を参照)。



(注)

ISSU 機能がないと、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンが 2 つの異なる Cisco IOS イメージ バージョンを実行している場合に、両者間で SSO または NSF 機能は動作しません。

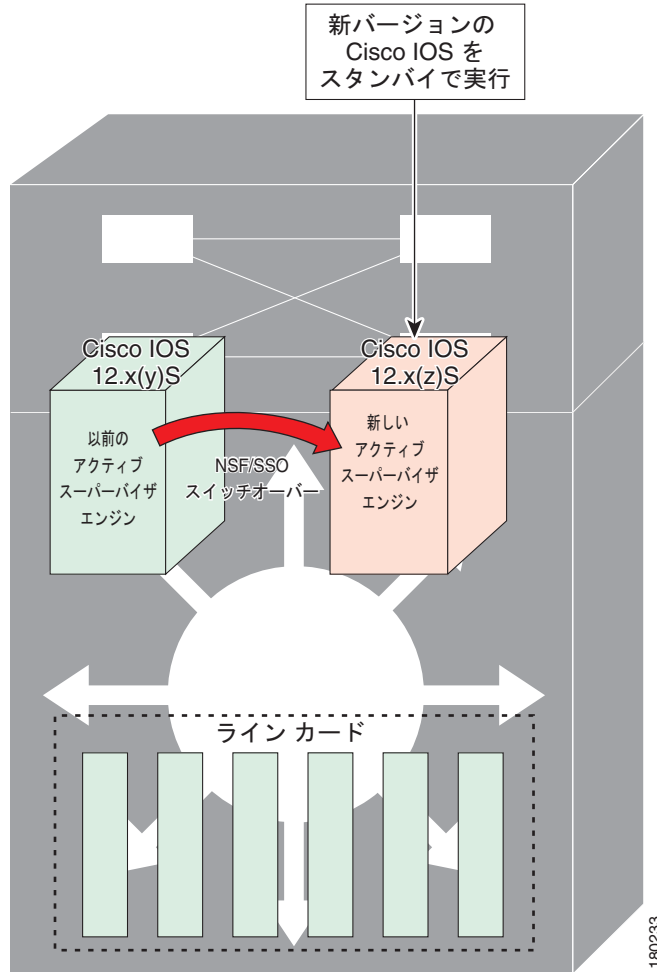
図 6-5 スタンバイ スーパーバイザ エンジンへの新しい Cisco IOS ソフトウェア バージョンのロード





スイッチオーバー（RPR ではなく、NSF または SSO）のあと、スタンバイ スーパーバイザ エンジンが新しくアクティブになったスーパーバイザ エンジンとして機能を引き継ぎます（図 6-6 を参照）。

図 6-6 スタンバイ スーパーバイザ エンジンへのスイッチオーバー



以前にアクティブだったスーパーバイザ エンジンには古い Cisco IOS イメージがロードされているので、新しくアクティブになったスーパーバイザ エンジンに問題が発生した場合には、中断して、すでに古いイメージを実行しているアクティブだったスーパーバイザ エンジンにスイッチオーバーできます。その後、アクティブだったスーパーバイザ エンジンに新しい Cisco IOS ソフトウェア バージョンがロードされ、新しいスタンバイ スーパーバイザ エンジンになります (図 6-7 を参照)。

図 6-7 新しくスタンバイになったスーパーバイザ エンジンへの新しい Cisco IOS ソフトウェアのロード

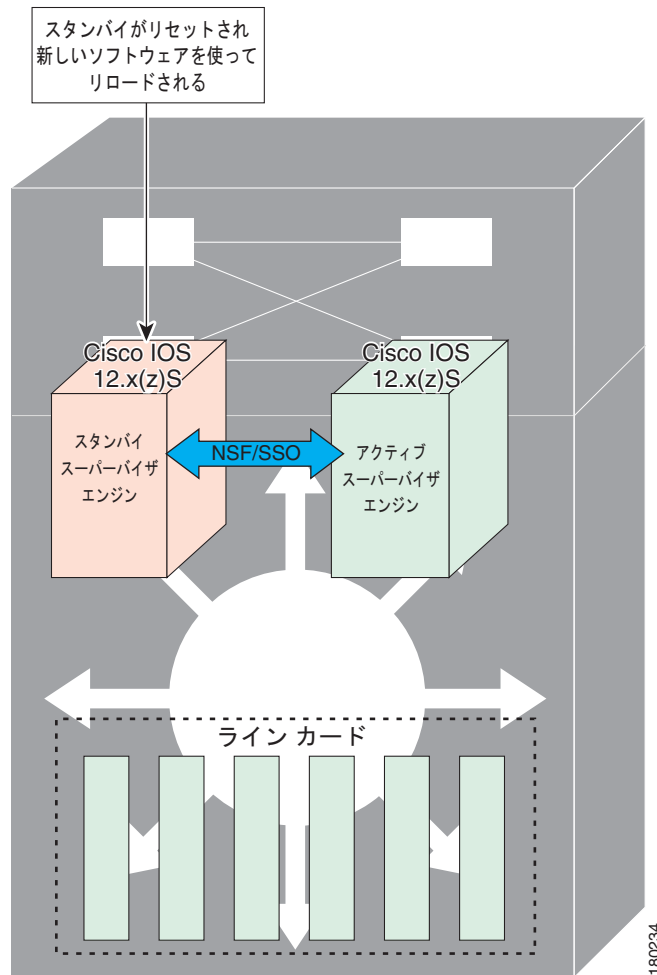
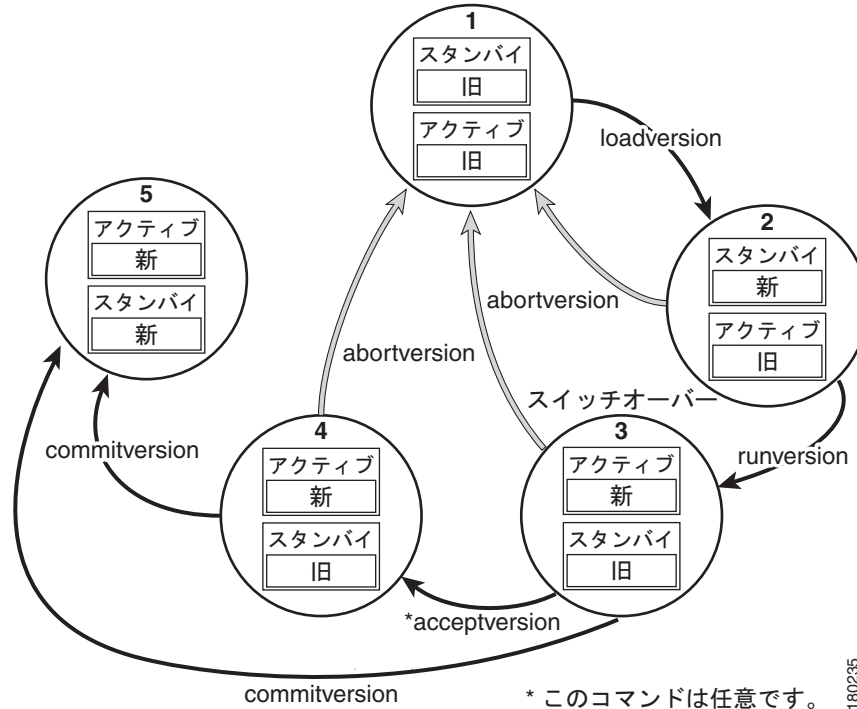


図 6-8 に、ISSU プロセス中の各ステップを示します。

図 6-8 ISSU プロセス中の各ステップ



## ISSU アップグレードの実行 : 2 つの方法

ISSU のアップグレード方法には、4 つのコマンドを使用した手動方式と 1 つのコマンドを使用した自動方式の 2 種類があります。

通常の ISSU アップグレードプロセスには、4 種類の ISSU exec コマンド (**issu loadversion**、**issu runversion**、**issu acceptversion**、**issu commitversion**) の発行とコマンドが成功したかどうかを評価するための追加の **show** コマンドの実行が含まれます。ISSU プロセスは複雑ですが、サービスの中断を想定しないようにする必要があります。複数の ISSU コマンドを使用する場合は、サービスを中断させないように配慮する必要があります。ただし、シナリオによっては、このアップグレード手順がやこしく、最小限のものになる場合があります。代表例は、多数の Catalyst 4500 スイッチ上での ISSU アップグレードを伴うネットワーク アップグレードです。このような場合は、最初に、1 台のスイッチ上で通常 (4 つのコマンド) の ISSU アップグレード手順を実行して正常なアップグレードを確認することを推奨します。その後で、1 つの **issu changeversion** コマンドを使用して、ネットワーク内の残りの Catalyst 4500 スイッチ上で自動 ISSU を実行します。



(注) **issu changeversion** コマンドを使用するには、古い IOS バージョンと新しい IOS バージョンの両方が **issu changeversion** 機能をサポートする必要があります。

## changeversion プロセス

**issu changeversion** コマンドは、1 ステップで完了する ISSU アップグレード サイクルを起動します。このコマンドは、4 つの標準コマンド (**issu loadversion**、**issu runversion**、**issu acceptversion**、および **issu commitversion**) すべてのロジックをユーザの介入なしで実行するため、単一の CLI ステップを通じたアップグレードの簡略化が図られます。

加えて、**issu changeversion** では、将来のアップグレード プロセスをスケジュールすることができます。これにより、いくつものシステムのステージングによって、潜在的な中断が最小の損害となる場合にアップグレードを順次実行することができます。

スタンバイ スーパーバイザ エンジンが初期化して、システムが **terminal** ステート (RPR/SSO) に到達したら、アップグレード プロセスが完了し、**BOOT** 変数が新しい IOS ソフトウェア イメージに恒久的に書き換えられます。そのため、どの RP をリセットしてもシステムは新しいソフトウェア イメージをブートします。状態遷移が発生したことをアップグレードの監視者に通知するために、コンソール メッセージと **syslog** メッセージが生成されます。

通常の ISSU アップグレード手順と同様に、**issu changeversion** コマンドによって開始された進行中のアップグレード手順は、**issu abortversion** コマンドで中断できます。システムが何らかの問題を検出するか、またはアップグレード中に健全でないシステムを検出すると、アップグレードは自動的に中断される場合があります。

**issu runversion** コマンドを 4 ステップの手動アップグレード プロセス時に入力したときに、互換性のない ISSU クライアントが存在すると、アップグレード プロセスは、それらとその副作用を報告し、ユーザがアップグレードを中止できるようにします。1 ステップのアップグレード プロセスの実行中に、プロセスが **runversion** ステートに達した場合、ベース クライアントに互換性があれば自動的にアップグレードを続行します。または、クライアントに互換性がないため自動的に中止します。ユーザが RPR モードでアップグレード手順を継続したい場合は、通常の ISSU コマンドセットを使用して、**issu loadversion** コマンドの入力時に **force** オプションを指定する必要があります。

## Changeversion : quick オプション

**issu changeversion** コマンドには、自動 ISSU アップグレードの実行に必要な時間を短縮可能な **quick** コマンド オプションがあります。**quick** コマンド オプションが指定された場合は、ISSU アップグレードの状態遷移が前述の状態遷移と異なります。このオプションでは、**loadversion** 段階上のソフトウェア ロジックは、以前の説明と同じままで、**runversion** および **commitversion** を実行するロジックが結合されます。このロジックでは、新しいスタンバイ (以前のアクティブ) スーパーバイザ上に古いソフトウェア バージョンをロードするアップグレード手順内のステップが省略されるため、自動 ISSU アップグレードに必要な時間が約 3 分の 1 短縮されます。

## Changeversion のスケジューリング : in オプションと at オプション

**issu changeversion** には、将来の自動 ISSU アップグレードをスケジュールできる **in** および **at** コマンド オプションがあります。

**at** コマンド オプションは、特定の時刻に開始するように自動 ISSU アップグレードをスケジュールします。このオプションを使用して、次の 24 時間以内にアップグレードを開始する正確な時刻 (**hh:mm**、24 時間形式) を指定します。

**in** コマンド オプションは、一定時間が経過後に自動 ISSU アップグレードを開始するようにスケジュールします。このオプションは、アップグレードを開始する前に経過する必要がある時間および分の長さ (**hh:mm** 形式) を指定します。最大値は 99:59 です。

## Changeversion の配置シナリオ

一般的な **issu changeversion** コマンドの使用シナリオは、大規模なインストール基盤を持つ経験豊富なユーザを対象としています。このようなユーザの多くは、実稼働ネットワークと同様のトポロジと設定を使用して新しいイメージを検証します。この検証プロセスは、従来の複数コマンドプロセスと新しい **issu changeversion** コマンドプロセスの両方を使用して実行する必要があります。ユーザが IOS ソフトウェア イメージを認定し、広くそれを展開する場合は、ネットワークの効率的なアップグレードを実行するために単一コマンドのプロセスを使用できます。

## 進行中の Changeversion 手順の中断

**issu changeversion** コマンドの機能は、ユーザの介入なしで ISSU ソフトウェア アップグレードを実行するように設計されています。ただし、さまざまな状態をアップグレードが遷移する際に、コンソールにステータス メッセージが表示されます。何らかの異常が、ネットワークのピアまたは他の部分などでの自動アップグレード中に認められた場合は、**issu abortversion** コマンドを使用して、**commitversion** 操作の前にプロセスの任意の時点で手動によりアップグレードを中止できます。

## ISSU の実行に関するガイドライン

ISSU プロセスの実行中は、次のガイドラインに注意してください。

- ISSU を使用している場合でも、メンテナンス ウィンドウの間にアップグレードを実行することを推奨します。
- ISSU プロセス中は、設定の変更が必要になるような新しい機能をイネーブルにしないでください。



(注) これらをイネーブルにすると、コマンドは新しいバージョンだけでサポートされるため、システムは RPR モードを開始します。

- ダウングレードを行う場合、Cisco IOS ソフトウェア ハンドルのダウングレードリビジョンにない機能が合ったときは、ISSU プロセスを開始する前にその機能をディセーブルにしてください。

## ISSU をサポートする Cisco IOS ソフトウェアのバージョンニング機能

ISSU が導入される以前は、SSO モードを実行するには、各スーパーバイザ エンジンで同じ Cisco IOS ソフトウェア バージョンを実行する必要がありました。



(注) 冗長 HA 構成のシステムの動作モードは、スタンバイ スーパーバイザ エンジンがアクティブ スーパーバイザ エンジンに登録するときバージョン スtring を交換することによって決まります。

システムが SSO モードを開始するのは、両方のスーパーバイザ エンジンで実行されているバージョンが同じである場合だけです。同じでないと、冗長モードが RPR に変更されます。ISSU 機能を使用した場合、この実装では Cisco IOS イメージの 2 つの異なる、しかし互換性のあるリリース レベルを SSO モードで相互動作することで、ソフトウェア アップグレードを実行しながら、パケット転送を継続することができます。ISSU 機能が導入される前に行われていたバージョンチェックでは、システムが動作モードを決定できなくなりました。

ISSU では、ソフトウェア バージョン間の互換性を判別するための追加情報が必要になります。当該イメージと比べた他のイメージに関する情報を含む互換性マトリクスが定義されます。この互換性マトリクスは、2 つのソフトウェア バージョン (1 つは、アクティブ スーパーバイザ エンジンで実行される

ソフトウェア バージョンで、もう一方はスタンバイ スーパーバイザ エンジンで実行されるソフトウェア バージョン) の互換性を表し、これによって、システムは実現可能な最も高度な動作モードを判別できます。バージョンに互換性がないと、SSO 動作モードに進むことができません。

## 互換性マトリクス

アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方の Cisco IOS ソフトウェアが ISSU に対応しており、古いイメージと新しいイメージに互換性がある場合に、ISSU プロセスを実行できます。互換性マトリクス情報では、次のようにリリース間の互換性が示されます。

- **Compatible** (互換性がある) : ベースレベルのシステム インフラストラクチャとすべてのオプションの HA 認識サブシステムに互換性があります。これらのバージョン間のインサーブिस アップグレードまたはダウングレードが正常に行われ、サービスに対する影響は最小限ですみます。マトリクス エントリでは、このようなイメージに対して **Compatible (C)** が指定されます。
- **Base-level compatible** (ベースレベルで互換性がある) : 1 つまたは複数のオプションの HA 認識サブシステムに互換性がありません。これらのバージョン間のインサーブिस アップグレードまたはダウングレードは正常に行われますが、一部のサブシステムでは、旧式から Cisco IOS の新バージョンへの移行中に、ステートを必ずしも維持できない場合があります。マトリクス エントリでは、このようなイメージに対して **Base-level compatible (B)** が指定されます。

ただし、マトリクス エントリが B の場合でも、機能を失うことなく、ISSU アップグレードを実行できる必要があります。ダウングレードでは新しいイメージに追加機能があると、一部の機能が失われることがあります。

- **Incompatible** (互換性がない) : SSO が正常に機能するためには、Cisco IOS 内に存在するシステム インフラストラクチャのコア セットがステートフル方式で相互動作できる必要があります。必要なこれらのいずれかの機能またはサブシステムが相互動作できないと、Cisco IOS ソフトウェア イメージの 2 つのバージョンに互換性がないと判定されます。これらのバージョン間でインサーブिस アップグレードまたはダウングレードを行うことはできません。マトリクス エントリでは、このようなイメージに対して **Incompatible (I)** が指定されます。システムは、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの Cisco IOS バージョンに互換性がない間は RPR モードで稼働します。

ISSU をサポートしないピアで ISSU を実行しようとする、システムは代わりに RPR を自動的に使用します。

互換性マトリクスには、ある Cisco IOS ソフトウェア イメージと、指定されたサポート ウィンドウの含まれる他のすべての Cisco IOS ソフトウェア バージョン (たとえば、イメージが「認識」しているすべてのソフトウェア バージョン) との互換性関係を示し、イメージごとに作成され、リリースされます。マトリクスには、自身のリリースと以前のリリース間の互換性の情報が含まれています。常に最新のリリースに、その分野の既存のリリースとの互換性に関する最新情報が含まれます。互換性マトリクスは Cisco IOS ソフトウェア イメージ内および Cisco.com で入手できるため、ISSU プロセスを使用してアップグレードを行えるかどうかを前もって判別できます。

任意のシステムの 2 つのソフトウェア バージョン間の互換性マトリクス データを表示するには、**show issu comp-matrix stored** コマンドを入力します。



(注)

このコマンドは、ISSU プロセスが開始したあとにだけ使用できるので、確認する場合にだけ有効です。ISSU を開始する前に互換性マトリクスをチェックする場合に便利です。Feature Navigator を使用すると、必要な情報を取得できます。

<http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp>

## ISSU に対する SNMP サポート

SSO に対する SNMP は、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンが同じ Cisco IOS ソフトウェア バージョンを実行していることを前提として、SNMP 設定と MIB を同期化するメカニズムを提供し、アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへの SSO をサポートしています。この前提は、ISSU には当てはまりません。

ISSU を使用した場合、SNMP クライアントは必要に応じて、2 つの異なる Cisco IOS バージョン間で MIB の変換を行うことができます。SNMP クライアントはすべての MIB の変換を行い、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジン間の送受信機能を処理します。SNMP の実行時に、両方の Cisco IOS リリースの MIB バージョンが同じである場合にだけ、MIB がアクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンに完全に同期化されます。

## Cisco Feature Navigator を使用した互換性の検証

Cisco Feature Navigator の ISSU アプリケーションでは、次の内容を実行することができます。

- ISSU 対応イメージを選択する
- 対象のイメージと互換性を持つイメージを特定する
- 2 つのイメージを比較し、それらのイメージの互換性レベル（互換性がある、ベースレベルで互換性がある、互換性がない）を把握する
- 2 つのイメージを比較し、各 ISSU クライアントのクライアント互換性を確認する
- イメージのリリース ノートに対するリンクを提供する

## ISSU プロセスの実行

デバイスの動作モードであり、ISSU を実行するための前提条件である SSO とは異なり、ISSU プロセスはスイッチの稼働中に実行される一連のステップです。このステップによって、Cisco IOS ソフトウェアが新しいソフトウェアにアップグレードまたは変更されますが、トラフィックへの影響は最小限に抑えられます。



(注) ISSU プロセスで使用されるコマンドの説明については、[図 6-8 \(P.6-11\)](#) を参照してください。

ここでは、次の内容について説明します。

- 「[以前のリリースから Cisco IOS XE 3.4.0SG/15.1\(2\)SG への ISSU のアップグレード](#)」 (P.6-16)
- 「[Cisco IOS XE 3.4.0SG/15.1\(2\)SG から以前のリリースへの ISSU のダウングレード](#)」 (P.6-17)
- 「[ISSU ソフトウェア インストールの確認](#)」 (P.6-18)
- 「[スタンバイ スーパーバイザ エンジンへの新しい Cisco IOS ソフトウェアのロード](#)」 (P.6-21) (必須)
- 「[スタンバイ スーパーバイザ エンジンへの切り替え](#)」 (P.6-24) (必須)
- 「[ISSU ロールバック タイマーの停止 \(任意\)](#)」 (P.6-26) (任意)
- 「[新しくスタンバイになったスーパーバイザ エンジンへの新しい Cisco IOS ソフトウェアのロード](#)」 (P.6-27)
- 「[ISSU プロセス中のソフトウェア アップグレードの中断](#)」 (P.6-35)

- 「アップグレード問題を回避するためのロールバック タイマーの設定」 (P.6-35)
- 「ISSU 互換性マトリクス情報の表示」 (P.6-37)

## 以前のリリースから Cisco IOS XE 3.4.0SG/15.1(2)SG への ISSU のアップグレード

Cisco IOS XE 3.4.0SG/15.1(2)SG よりも前のイメージは以前の CLI 形式を使用し、Cisco IOS XE 3.4.0SG および 15.1(2)SG のイメージは新しい CLI 形式を使用するため、アップグレードは次で構成されています。

- Cisco IOS XE 3.4.0SG/15.1(2)SG にスイッチのイメージをアップグレードする。
- mgmtVrf を以前の CLI 形式からより後の形式にアップグレードして、インターフェイスの IPv6 アドレスを削除する。
- mgmtVrf の下で IPv6 アドレス ファミリーをイネーブルにし、Fa1 の IPv6 アドレスを再設定する。

次のような設定が IOS XE 3.4.0SG/15.1(2)SG 以前のイメージに存在する必要があります。

```
ip vrf mgmtVrf
!
interface FastEthernet1
 ip vrf forwarding mgmtVrf
 ip address 10.1.1.1 255.255.255.0
 speed auto
 duplex auto
 ipv6 address 2000::1/64
!
```

**ステップ 1** Cisco IOS XE 3.4.0SG/15.1(2)SG イメージへの ISSU のアップグレードを実行します。

**ステップ 2** VRF upgrade コマンドを実行します。

```
Switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# vrf upgrade-cli multi-af-mode common-policies vrf mgmtVrf
You are about to upgrade to the multi-AF VRF syntax commands.
You will lose any IPv6 address configured on interfaces
belonging to upgraded VRFs.

Are you sure ? [yes]:
Number of VRFs upgraded: 1
Switch(config)# exit
```

設定は次のように表示されます。

```
vrf definition mgmtVrf
!
 address-family ipv4
 exit-address-family
!
interface FastEthernet1
 vrf forwarding mgmtVrf
 ip address 10.1.1.1 255.255.255.0
 speed auto
 duplex auto
!
```

**ステップ 3** IPv6 アドレス ファミリーをイネーブルにし、IPv6 アドレスを追加するようにスイッチを設定します。



```
Switch# config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# vrf definition mgmtVrf
Switch(config-vrf)# address-family ipv6
Switch(config-vrf-af)# exit
Switch(config-vrf)# exit
Switch(config)# interface fa1
Switch(config-if)# ipv6 address 2000::1/64
Switch(config-if)# end
```

設定は次のように表示されます。

```
vrf definition mgmtVrf
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
interface FastEthernet1
vrf forwarding mgmtVrf
ip address 10.1.1.1 255.255.255.0
speed auto
duplex auto
ipv6 address 2000::1/64
```

## Cisco IOS XE 3.4.0SG/15.1(2)SG から以前のリリースへの ISSU のダウングレード

Cisco IOS XE 3.4.0SG/15.1(2)SG のイメージは新しい CLI 形式を使用し、以前のイメージは以前の CLI 形式を使用するため、ダウングレード手順は次のとおりです。

- mgmtVrf を新しい CLI 形式からより古い CLI 形式にダウングレードして、インターフェイスの IPv6 アドレスを削除する。
- 以前のリリースへスイッチのイメージをダウングレードする。
- Fa1 の IPv6 アドレスを再設定する。

次のような設定が IOS XE 3.4.0SG/15.1(2)SG イメージを実行しているスイッチに表示されます。

```
vrf definition mgmtVrf
!
address-family ipv4
exit-address-family
!
address-family ipv6
exit-address-family
!
interface FastEthernet1
vrf forwarding mgmtVrf
ip address 10.1.1.1 255.255.255.0
speed auto
duplex auto
ipv6 address 2000::1/64
!
```

**ステップ 1** Cisco IOS XE 3.4.0SG/15.1(2)SG より前のリリースへのダウングレードを実行します。

```
Switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# no vrf upgrade-cli multi-af-mode common-policies vrf mgmtVrf
You are about to downgrade to the single-AF VRF syntax commands.
You will lose any IPv6 address configured on interfaces
belonging to downgraded VRFs.

Are you sure ? [yes]:
% ipv6 addresses from all interfaces in VRF mgmtVrf have been removed
Number of VRFs downgraded: 1
Switch(config)#
```

設定は次のように表示されます。

```
ip vrf mgmtVrf
!
interface FastEthernet1
 ip vrf forwarding mgmtVrf
 ip address 10.1.1.1 255.255.255.0
 speed auto
 duplex auto
!
```

**ステップ 2** Cisco IOS XE 3.4.0SG/15.1(2)SGn 以前のイメージへの ISSU のダウングレードを実行します。

**ステップ 3** IPv6 アドレスを再設定します。

```
Switch# config t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface fa1
Switch(config-if)# ipv6 address 2000::1/64
Switch(config-if)# end
Switch#
```

設定は次のように表示されます。

```
ip vrf mgmtVrf
!
interface FastEthernet1
 ip vrf forwarding mgmtVrf
 ip address 10.1.1.1 255.255.255.0
 speed auto
 duplex auto
 ipv6 address 2000::1/64
```

## ISSU ソフトウェア インストールの確認

ISSU プロセスには、5 つのステート (Disabled、Init、Load Version、Run Version、および System Reset) があります。 **show issu state** コマンドを使用すると、現在の ISSU ステートを取得できます。

- **Disabled** ステート : エンジンがリセットされている間のスタンバイ スーパーバイザ エンジンの状態。
- **Init** ステート : ISSU プロセスが開始する前の、2 つのスーパーバイザ エンジン (1 つはアクティブで、他方はスタンバイ) の初期ステート。ISSU プロセスが完了したあとの最終ステートでもある。
- **Load Version (LV)** ステート : スタンバイ スーパーバイザ エンジンに新しい Cisco IOS ソフトウェア バージョンがロードされた場合のステート。

- Run Version (RV) ステート : **issu runversion** コマンドによって、スーパーバイザ エンジンのスイッチオーバーが強制されるステート。新たにアクティブになったスーパーバイザ エンジンが新しい Cisco IOS ソフトウェア イメージを実行。
- System Reset (SR) ステート : Init ステートに達する前に **issu abortversion** コマンドを入力した場合、または **issu acceptversion** コマンドを実行する前にロールバック タイマーの期限が切れた場合のステート。

ISSU ソフトウェア インストレーションを確認するには、次のように **show** コマンドを入力します。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>• プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ2	Switch# <b>show issu state [detail]</b>	ISSU プロセス中のスーパーバイザ エンジンのステートを表示します。
ステップ3	Switch# <b>show redundancy</b>	デバイスの現在または過去のステータス、モード、および関連する冗長情報を表示します。

次に、ISSU プロセス中のスーパーバイザ エンジンのステートと現在のステータスを表示する例を示します。

```
Switch> enable
Switch# show issu state
Switch# show redundancy
```

## ISSU プロセスを開始する前の冗長モードの確認

ISSU プロセスを開始する前に、システムの冗長モードを確認して、NSF/SSO を必ず設定するようにしてください。

次に、システムが SSO モードを開始しており、スロット 1 がアクティブ スーパーバイザ エンジンで、スロット 2 がスタンバイ スーパーバイザ エンジンであることを確認する例を示します。両方のスーパーバイザ エンジンで同じ Cisco IOS ソフトウェア イメージを実行しています。

```
Switch# show redundancy states
my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 1

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

client count = 39
client_notification_TMR = 240000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 0
keep_alive threshold = 18
RF debug mask = 0x0
```

```

Switch# show redundancy
Redundant System Information :
-----
    Available system uptime = 1 minute
Switchovers system experienced = 0
    Standby failures = 0
    Last switchover reason = none

    Hardware Mode = Duplex
Configured Redundancy Mode = Stateful Switchover
Operating Redundancy Mode = Stateful Switchover
Maintenance Mode = Disabled
Communications = Up

Current Processor Information :
-----
    Active Location = slot 1
    Current Software state = ACTIVE
    Uptime in current state = 0 minutes
    Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
    BOOT = bootflash:old_image,1;
    Configuration register = 0x822

Peer Processor Information :
-----
    Standby Location = slot 2
    Current Software state = STANDBY HOT
    Uptime in current state = 1 minute
    Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
    BOOT = bootflash:old_image,1;
    Configuration register = 0x822

```

## ISSU プロセスを開始する前の ISSU ステートの確認

アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンがアップおよび ISSU Init ステートで、ブート変数が設定されており、有効なファイルが指定されていることを確認します。

次に、プロセスを開始する前に ISSU ステートを表示する例を示します。

```

Switch# show issu state detail
    Slot = 1
    RP State = Active
    ISSU State = Init
    Boot Variable = bootflash:old_image,1;
    Operating Mode = Stateful Switchover
    Primary Version = N/A
    Secondary Version = N/A
    Current Version = bootflash:old_image

    Slot = 2
    RP State = Standby
    ISSU State = Init

```

```

Boot Variable = bootflash:old_image,1;
Operating Mode = Stateful Switchover
Primary Version = N/A
Secondary Version = N/A
Current Version = bootflash:old_image

```

新しい Cisco IOS ソフトウェア バージョンが両方のスーパーバイザ エンジンに存在する必要があります。次に、新しいバージョンが存在することを確認するために、それぞれのスーパーバイザ エンジンのディレクトリ情報を表示する例を示します。

```

Switch# dir bootflash:
Directory of bootflash:/

   5  -rwx   13636500   Sep 6 2006 09:32:33 +00:00  old_image
   6  -rwx   13636500   Sep 6 2006 09:34:07 +00:00  new_image

61341696 bytes total (1111388 bytes free)

Switch# dir slavebootflash:
Directory of slavebootflash:/

   4  -rwx   13636500   Sep 6 2006 09:40:10 +00:00  old_image
   5  -rwx   13636500   Sep 6 2006 09:42:13 +00:00  new_image

61341696 bytes total (1116224 bytes free)

```

## スタンバイ スーパーバイザ エンジンへの新しい Cisco IOS ソフトウェアのロード

ここでは、ISSU を使用して、スタンバイ スーパーバイザ エンジンに新しい Cisco IOS ソフトウェア バージョンをロードする方法について説明します。

### 前提条件

- 新しい Cisco IOS ソフトウェア イメージのバージョンがアクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方にすでに存在していることを確認します。また、適切なブート パラメータ (BOOT ストリングおよびコンフィギュレーション レジスタ) がスタンバイ スーパーバイザ エンジンに設定されていることを確認します。



(注) スイッチは ISSU 手順が試行される前にブート ストリング設定でブートする必要があります。



(注) ISSU が正常に実行されるためには **auto-boot** をイネーブルにする必要があります。

- (任意) 追加のテストおよびコマンドを実行して、あとで比較するために必要なピアおよびインターフェイスの現在のステートを判別します。
- システム (アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方) が SSO 冗長モードになっていることを確認します。システムが SSO モードではなく、RPR モードである場合、ISSU CLI コマンドを使用してシステムをアップグレードすることはできませんが、アップグレード中にシステムが大量のパケットを損失します。

スーパーバイザ エンジンに SSO モードを設定する方法の詳細については、『*Stateful Switchover*』を参照してください。

## ISSU プロセスの実行

- ISSU が機能するためには、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンのイメージ名が一致する必要があります。

アクティブ スーパーバイザ エンジンで次の作業を行います。

	コマンドまたはアクション	目的
ステップ 1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。  <ul style="list-style-type: none"> <li>プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ 2	Switch# <b>issu loadversion active-slot active-image-new standby-slot standby-image-new [forced]</b>	ISSU プロセスを開始します。また、新しい Cisco IOS ソフトウェア バージョンに互換性がないことが検知された場合には、自動ロールバックを無効にします (任意)。  <b>issu loadversion</b> コマンドを入力してからスタンバイ スーパーバイザ エンジンに Cisco IOS ソフトウェアがロードされて、スタンバイ スーパーバイザ エンジンが SSO モードに移行するまでには数秒かかります。これによって、スタンバイ スーパーバイザ エンジンに新しいイメージがロードされます。  <b>forced</b> オプションを使用すると、スタンバイ スーパーバイザ エンジンが新しいイメージで起動します。スタンバイ スーパーバイザ エンジンにイメージがロードされたあと、イメージに互換性がないと、システムは強制的に RPR モードになります。それ以外の場合、システムは SSO モードを続行します。
ステップ 3	Switch# <b>show issu state [detail]</b>	ISSU プロセス中のスーパーバイザ エンジンのステータスを表示します。ISSU プロセスのこの時点で、このコマンドを使用して、スタンバイ スーパーバイザ エンジンがロードされ、SSO モードになっていることを確認します。  <b>issu loadversion</b> コマンドを入力してから、スタンバイ スーパーバイザ エンジンに Cisco IOS ソフトウェアがロードされ、スタンバイ スーパーバイザ エンジンが SSO モードに移行するまでに数秒かかる場合があります。 <b>show issu state</b> コマンドを入力するタイミングが早すぎると、必要な情報が表示されない場合があります。
ステップ 4	Switch# <b>show redundancy [states]</b>	冗長ファシリティ ステータス情報を表示します。

次に、ISSU プロセスを開始し、スタンバイ スーパーバイザ エンジンを Standby Hot ステータスで起動し、スタンバイ スーパーバイザ エンジン (スロット 2) に新しいイメージをロードする例を示します。

```
Switch> enable
Switch# issu loadversion 1 bootflash:new_image 2 slavebootflash:new_image
Switch# show issu state detail
      Slot = 1
      RP State = Active
      ISSU State = Load Version
      Boot Variable = bootflash:old_image,12
      Operating Mode = Stateful Switchover
      Primary Version = bootflash:old_image
      Secondary Version = bootflash:new_image
      Current Version = bootflash:old_image

      Slot = 2
```

```

RP State = Standby
ISSU State = Load Version
Boot Variable = bootflash:new_image,12;bootflash:old_image,12
Operating Mode = Stateful Switchover
Primary Version = bootflash:old_image
Secondary Version = bootflash:new_image
Current Version = bootflash:new_image

```

Switch# **show redundancy states**

```

my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 1

```

```

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

```

```

client count = 39
client_notification_TMR = 240000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 1
keep_alive threshold = 18
RF debug mask = 0x0

```

次に、forced オプションによってシステムが RPR モードに移行する例を示します。

Switch> **enable**

Switch# **issu loadversion 1 bootflash:new\_image 2 slavebootflash:new\_image forced**

Switch# **show issu state detail**

```

Slot = 1
RP State = Active
ISSU State = Load Version
Boot Variable = bootflash:old_image,12
Operating Mode = RPR
Primary Version = bootflash:old_image
Secondary Version = bootflash:new_image
Current Version = bootflash:old_image

Slot = 2
RP State = Standby
ISSU State = Load Version
Boot Variable = bootflash:new_image,12;bootflash:old_image,12
Operating Mode = RPR
Primary Version = bootflash:old_image
Secondary Version = bootflash:new_image
Current Version = bootflash:new_image

```

次に、冗長モードが RPR として表示される例を示します。

Switch# **show redundancy states**

```

my state = 13 -ACTIVE
peer state = 4 -STANDBY COLD
Mode = Duplex
Unit = Primary
Unit ID = 1

```

```

Redundancy Mode (Operational) = RPR
Redundancy Mode (Configured) = Stateful Switchover
Redundancy State = RPR
Maintenance Mode = Disabled
  Manual Swact = enabled
  Communications = Up

  client count = 39
  client_notification_TMR = 240000 milliseconds
  keep_alive TMR = 9000 milliseconds
  keep_alive count = 1
  keep_alive threshold = 18
  RF debug mask = 0x0

```

## スタンバイ スーパーバイザ エンジンへの切り替え

この作業では、新しい Cisco IOS ソフトウェア イメージを実行しているスタンバイ スーパーバイザ エンジンへのスイッチオーバー方法について説明します。

アクティブ スーパーバイザ エンジンで次の作業を行います。

	コマンドまたはアクション	目的
ステップ 1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。  • プロンプトが表示されたら、パスワードを入力します。
ステップ 2	Switch# <b>issu runversion standby-slot</b> [standby-image-new]	アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへのスイッチオーバーを強制し、アクティブだった（現在はスタンバイ）スーパーバイザ エンジンに古いイメージをリロードします。  <b>issu runversion</b> コマンドを入力すると、SSO スイッチオーバーが実行され、設定されている場合は NSF プロシージャが呼び出されます。
ステップ 3	Switch# <b>show issu state</b> [detail]	ISSU プロセス中のスーパーバイザ エンジンのステータスを表示します。ISSU プロセスのこの時点で、このコマンドを使用して、スロット 2 でスイッチオーバーが行われていることを確認します。
ステップ 4	Switch# <b>show redundancy</b> [states]	冗長ファシリティ ステータス情報を表示します。

次に、スタンバイだったスーパーバイザ エンジン（スロット 2）へのスイッチオーバーを発生させ、アクティブだったスーパーバイザ エンジンをリセットしたうえで古いイメージをリロードしてスタンバイ スーパーバイザ エンジンにする例を示します。

```

Switch> enable
Switch# issu runversion 2 slavebootflash:new_image
This command will reload the Active unit. Proceed ? [confirm]

```

スイッチオーバーはこの時点で行われます。古いアクティブ スーパーバイザ エンジンがスタンバイ エンジンになってから、新しいアクティブ スーパーバイザ エンジン上で次を行います。



```
Switch# show issu state detail
      Slot = 2
      RP State = Active
      ISSU State = Run Version
      Boot Variable = bootflash:new_image,12;bootflash:old_image,12
      Operating Mode = Stateful Switchover
      Primary Version = bootflash:new_image
      Secondary Version = bootflash:old_image
      Current Version = bootflash:new_image

      Slot = 1
      RP State = Standby
      ISSU State = Run Version
      Boot Variable = bootflash:old_image,12
      Operating Mode = Stateful Switchover
      Primary Version = bootflash:new_image
      Secondary Version = bootflash:old_image
      Current Version = bootflash:old_image
```



(注)

新しくアクティブになったスーパーバイザ エンジンには現在新しいソフトウェア バージョンを実行し、スタンバイ スーパーバイザ エンジンには古いソフトウェア バージョンを実行し、STANDBY HOT ステータスの状態です。

```
Switch# show redundancy states
my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Secondary
Unit ID = 2

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

client count = 39
client_notification_TMR = 240000 milliseconds
keep_alive TMR = 9000 milliseconds
keep_alive count = 1
keep_alive threshold = 18
RF debug mask = 0x0
```

**runversion** コマンドが完了すると、新しくアクティブになったスーパーバイザ エンジンが新しいソフトウェア バージョンを実行し、アクティブだったスーパーバイザ エンジンがスタンバイ スーパーバイザ エンジンになります。スタンバイがリセットされたうえでリロードされますが、以前のソフトウェア バージョンのまま、STANDBY HOT ステータスでオンラインに戻ります。次に、これらの状態を確認する例を示します。

```
Switch# show redundancy
Redundant System Information :
-----
      Available system uptime = 23 minutes
Switchovers system experienced = 1
      Standby failures = 0
      Last switchover reason = user forced
```

```

        Hardware Mode = Duplex
    Configured Redundancy Mode = Stateful Switchover
        Operating Redundancy Mode = Stateful Switchover
        Maintenance Mode = Disabled
        Communications = Up

Current Processor Information :
-----
        Active Location = slot 2
        Current Software state = ACTIVE
        Uptime in current state = 11 minutes
        Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
        (cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
        Technical Support: http://www.cisco.com/techsupport
        Copyright (c) 1986-2006 by Cisco Systems, Inc.
        Compiled Tue 05-Sep-06 16:16 by sanjdas
        BOOT = bootflash:new_image,12;bootflash:old_image,12
        Configuration register = 0x822

Peer Processor Information :
-----
        Standby Location = slot 1
        Current Software state = STANDBY HOT
        Uptime in current state = 4 minutes
        Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
        (cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
        Technical Support: http://www.cisco.com/techsupport
        Copyright (c) 1986-2006 by Cisco Systems, Inc.
        Compiled Tue 05-Sep-06 16:16 by sanjdas
        BOOT = bootflash:old_image,12
        Configuration register = 0x822

```

## ISSU ロールバック タイマーの停止（任意）

ここでは、ロールバック タイマーを停止する方法について説明します。これは、任意で行う操作です。ロールバック タイマーが「タイムアウト」する前に次の手順を実行しなかった場合、システムが自動的に ISSU プロセスを中断し、元の Cisco IOS ソフトウェア バージョンに戻ります。デフォルトのロールバック タイマーは 45 分です。

行う必要がある操作は、次のように判断します。

- スイッチを長時間この状態で維持する場合は、ロールバック タイマーを停止する必要があります（その後、確認して、直接 **acceptversion** コマンドを実行します）。
- 45 分間のロールバック タイマー時間内に次のステップ（「**commitversion**」を実行）に進む場合は、ロールバック タイマーを停止する必要はありません。



(注) **issu runversion** コマンドのあと、任意で **issu acceptversion** コマンドを実行することができます。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ2	Switch# <b>issu acceptversion active-slot</b> [active-image-new]	ロールバック タイマーを中止し、新しい Cisco IOS ISSU プロセスが ISSU プロセス中に自動的に中断されていないようにします。  ロールバック タイマーによって指定された時間内に <b>issu acceptversion</b> コマンドを入力して、スーパーバイザ エンジンが外部への接続を確立したことを承認します。そうしないと、ISSU プロセスが終了し、システムはスタンバイ スーパーバイザ エンジンに切り替えて、以前の Cisco IOS ソフトウェア バージョンに戻ります。
ステップ3	Switch# <b>show issu rollback-timer</b>	自動ロールバックが行われるまでの時間を表示します。

次に、停止する前のタイマーを表示する例を示します。次の例では、Automatic Rollback Time 情報は自動ロールバックが行われるまでの時間を示します。

```
Switch> enable
Switch# show issu rollback-timer
Rollback Process State = In progress
Configured Rollback Time = 45:00
Automatic Rollback Time = 38:30

Switch# issu acceptversion 2 bootflash:new_image
% Rollback timer stopped. Please issue the commitversion command.
Switch# show issu rollback-timer
Rollback Process State = Not in progress
Configured Rollback Time = 45:00
```

## 新しくスタンバイになったスーパーバイザ エンジンへの新しい Cisco IOS ソフトウェアのロード

ここでは、新しくスタンバイになったスーパーバイザ エンジンに新しい Cisco IOS ソフトウェア バージョンをロードする方法について説明します。

アクティブ スーパーバイザ エンジンで次の作業を行います。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ2	Switch# <b>issu commitversion standby-slot-number</b> [standby-image-new]	新しい Cisco IOS ソフトウェア イメージがスタンバイ スーパーバイザ エンジンにロードされるようにします。

	コマンドまたはアクション	目的
ステップ 3	Switch# <b>show redundancy</b> [states]	冗長ファシリティ ステート情報を表示します。
ステップ 4	Switch# <b>show issu state</b> [detail]	ISSU プロセス中のスーパーバイザ エンジンのステートを表示します。ISSU プロセスのこの時点で、このコマンドを使用して、スロット 2 でスイッチオーバーが行われていることを確認します。

次に、現在のスタンバイ スーパーバイザ エンジン (スロット 1) をリセットして、新しい Cisco IOS ソフトウェア バージョンをリロードする例を示します。**commitversion** コマンドを入力したあと、スタンバイ スーパーバイザ エンジンが Standby Hot ステートで起動します。

```
Switch> enable
Switch# issu commitversion 1 slavebootflash:new_image
```

Wait till standby supervisor is reloaded with the new image. Then apply the following:

```
Switch# show redundancy states
00:17:12: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
  my state = 13 -ACTIVE
  peer state = 8 -STANDBY HOT
  Mode = Duplex
  Unit = Secondary
  Unit ID = 2
```

```
Redundancy Mode (Operational) = Stateful Switchover
Redundancy Mode (Configured) = Stateful Switchover
Redundancy State = Stateful Switchover
Maintenance Mode = Disabled
  Manual Swact = enabled
  Communications = Up
```

```
  client count = 39
  client_notification_TMR = 240000 milliseconds
  keep_alive TMR = 9000 milliseconds
  keep_alive count = 0
  keep_alive threshold = 18
  RF debug mask = 0x0
```

```
Switch# show redundancy
Redundant System Information :
-----
  Available system uptime = 41 minutes
  Switchovers system experienced = 1
  Standby failures = 1
  Last switchover reason = user forced
```

```
  Hardware Mode = Duplex
  Configured Redundancy Mode = Stateful Switchover
  Operating Redundancy Mode = Stateful Switchover
  Maintenance Mode = Disabled
  Communications = Up
```

```
Current Processor Information :
-----
  Active Location = slot 2
  Current Software state = ACTIVE
  Uptime in current state = 29 minutes
  Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
```

```

Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
          BOOT = bootflash:new_image,12;bootflash:old_image,1;
Configuration register = 0x822

Peer Processor Information :
-----
          Standby Location = slot 1
          Current Software state = STANDBY HOT
          Uptime in current state = 12 minutes
          Image Version = Cisco IOS Software, Catalyst 4500 L3 Switch Software
(cat4500-ENTSERVICES-M), Version 12.2(31)SGA, RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2006 by Cisco Systems, Inc.
Compiled Tue 05-Sep-06 16:16 by sanjdas
          BOOT = bootflash:new_image,12;bootflash:old_image,1;
Configuration register = 0x822

Switch# show issu state detail
          Slot = 2
          RP State = Active
          ISSU State = Init
          Boot Variable = bootflash:new_image,12;bootflash:old_image,1;
          Operating Mode = Stateful Switchover
          Primary Version = N/A
          Secondary Version = N/A
          Current Version = bootflash:new_image

          Slot = 1
          RP State = Standby
          ISSU State = Init
          Boot Variable = bootflash:new_image,12;bootflash:old_image,1;
          Operating Mode = Stateful Switchover
          Primary Version = N/A
          Secondary Version = N/A
          Current Version = bootflash:new_image

```

ISSU プロセスが完了しました。これ以降、Cisco IOS ソフトウェア バージョンのアップグレードまたはダウングレードを行うには、新しい ISSU プロセスの起動が必要になります。

## changeversion を使用した ISSU アップグレードの自動化

このタスクでは、**issu changeversion** コマンドを使用して 1 ステップの ISSU アップグレードを実行する方法について説明します。

### 前提条件

- 新しい Cisco IOS ソフトウェア イメージのバージョンがアクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方にすでに存在していることを確認します。また、適切なブート パラメータ (BOOT 文字列と設定レジスタ) が、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンに設定されていることを確認します。
- (任意) 追加のテストおよびコマンドを実行して、あとで比較するために必要なピアおよびインターフェイスの現在のステートを判別します。
- システム (アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンの両方) が SSO 冗長モードになっていることを確認します。システムが RPR モードである場合、ISSU CLI コマンドを使用してシステムをアップグレードすることはできませんが、アップグレード中にシステムが大量のパケットを損失します。

## ISSU プロセスの実行

スーパーバイザ エンジンに SSO モードを設定する方法の詳細については、『Stateful Switchover』を参照してください（第 10 章「Supervisor Engine 6-E および Supervisor Engine 6L-E で RPR および SSO を使用したスーパーバイザ エンジンの冗長構成の設定」を参照）。

- ISSU が機能するためには、アクティブ スーパーバイザ エンジンとスタンバイ スーパーバイザ エンジン上の IOS XE ソフトウェア イメージ ファイル名が一致する必要があります。

アクティブ スーパーバイザ エンジンで次の手順を実行します。

	コマンドまたはアクション	目的
ステップ 1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 プロンプトが表示されたら、パスワードを入力します。
ステップ 2	Switch# <b>issu changeversion</b> [active-slot active-image-new] [standby-slot standby-image-new] [at hh:mm   in hh:mm] [quick]	単一ステップの完全アップグレードプロセス サイクルを開始します。ユーザの介入なしで 4 つの標準コマンド (issu loadversion、issu runversion、issu acceptversion、および issu commitversion) のロジックを実行します。 <i>active-slot</i> : アクティブなスロット番号を定義します。 <i>new-image</i> : アップグレード先の IOS XE イメージの URL を指定します。 <i>standby-slot</i> : スタンバイ スロット番号を定義します。 <i>standby-image</i> : スタンバイ IOS XE イメージの URL を指定します。 <b>at hh:mm</b> : 将来開始する ISSU アップグレードをスケジュールします。次の 24 時間以内にアップグレードを開始する正確な時刻 (hh:mm、24 時間形式) を入力します。 <b>in hh:mm</b> : 将来開始する ISSU アップグレードをスケジュールします。アップグレードを開始する前に経過する時間および分の長さ (hh:mm 形式) を指定します (最大 99:59)。 <b>quick</b> : スイッチオーバー時に、スタンバイ スーパーバイザ エンジンを古いイメージではなく、新しいイメージでブートすることによって、アップグレードを高速化します。
ステップ 3	Switch# <b>show issu state</b> [detail]	ISSU プロセス中のスーパーバイザ エンジンのステータスを表示します。ISSU プロセスのこの時点で、このコマンドを使用して、スタンバイ スーパーバイザ エンジンがロードされ、SSO モードになっていることを確認します。
ステップ 4	Switch# <b>show redundancy</b> [states]	冗長ファシリティ ステータス情報を表示します。

次に、スロット番号 5 (現在のアクティブ スーパーバイザ エンジンのスロット) で **issu changeversion** コマンドを使用して ISSU アップグレードプロセスを開始する例を示します。show issu state detail コマンドと show redundancy コマンドの出力が含まれており、アップグレード手順前後のスーパーバイザ ステータスが確認できます。



(注) 下の出力に含まれる成功メッセージは、ISSU アップグレード手順が ISSU ステータスを進むため、いくらか遅延した後で表示されます。

```
Switch> enable
```

```
Switch# show issu state detail
      Slot = 5
      RP State = Active
      ISSU State = Init
      Operating Mode = Stateful Switchover
      Current Image = bootflash:x.bin
      Pre-ISSU (Original) Image = N/A
      Post-ISSU (Targeted) Image = N/A

      Slot = 6
      RP State = Standby
      ISSU State = Init
      Operating Mode = Stateful Switchover
      Current Image = bootflash:x.bin
      Pre-ISSU (Original) Image = N/A
      Post-ISSU (Targeted) Image = N/A

Switch# show redundancy
Redundant System Information :

-----
      Available system uptime = 12 minutes
      Switchovers system experienced = 0
      Standby failures = 0
      Last switchover reason = none

      Hardware Mode = Duplex
      Configured Redundancy Mode = Stateful Switchover
      Operating Redundancy Mode = Stateful Switchover
      Maintenance Mode = Disabled
      Communications = Up

Current Processor Information :
-----
      Active Location = slot 5
      Current Software state = ACTIVE
      Uptime in current state = 9 minutes
      Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

Peer Processor Information :
-----
      Standby Location = slot 6
      Current Software state = STANDBY HOT
      Uptime in current state = 2 minutes
      Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

Switch# issu changeversion bootflash:y.bin
% 'issu changeversion' is now executing 'issu loadversion'
% issu loadversion executed successfully, Standby is being reloaded

% changeversion finished executing loadversion, waiting for standby to reload and reach
SSO ...
```



(注) スタンバイにターゲット イメージがリロードされます。

.....  
.....

```
*Feb 25 20:41:00.479: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion is now executing
'issu runversion'
*Feb 25 20:41:03.639: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion successfully executed
'issu runversion'
```



(注) スイッチオーバーが発生します。

.....  
.....

新しいアクティブ スーパーバイザ エンジンのコンソールを確認します。

```
*Feb 25 20:47:39.859: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
*Feb 25 20:47:39.971: %INSTALLER-7-ISSU_OP_SUCC:  issu changeversion is now executing
'issu commitversion'
```

....  
....



(注) 新しいスタンバイ スーパーバイザにターゲット イメージがリロードされます。SSO terminal ステートに達すると **changeversion** が成功します。

```
*Feb 25 20:54:16.092: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Feb 25 20:54:16.094: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)
Switch#
```

Switch# **show issu state detail**

```
Slot = 6
RP State = Active
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:y.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A
```

```
Slot = 5
RP State = Standby
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:y.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A
```

Switch# **show redundancy**

Redundant System Information :

```
-----
Available system uptime = 12 minutes
Switchovers system experienced = 0
Standby failures = 0
Last switchover reason = none
```



```

Hardware Mode = Duplex
Configured Redundancy Mode = Stateful Switchover
Operating Redundancy Mode = Stateful Switchover
Maintenance Mode = Disabled
Communications = Up

```

## Current Processor Information :

```

-----
Active Location = slot 6
Current Software state = ACTIVE
Uptime in current state = 9 minutes
Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

```

## Peer Processor Information :

```

-----
Standby Location = slot 5
Current Software state = STANDBY HOT
Uptime in current state = 2 minutes
Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

```

次に、**at** コマンド オプション付きの **issu changeversion** を使用して、指定した時刻に ISSU アップグレード手順が自動的に開始されるようにスケジュールする例を示します。この例では、ISSU アップグレードが 16:30 (24 時間形式) に開始するように指定します。**show issu state detail** コマンドと **show redundancy** コマンドの出力は、**issu changeversion** コマンドが入力された前後のスーパーバイザの状態を示すために含まれます。

```

Switch> enable
Switch# show issu state detail

Slot = 5
RP State = Active
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:x.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

Slot = 6
RP State = Standby
ISSU State = Init
Operating Mode = Stateful Switchover
Current Image = bootflash:x.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

```

## Switch# show redundancy

Redundant System Information :

```

-----
Available system uptime = 12 minutes
Switchovers system experienced = 0
Standby failures = 0
Last switchover reason = none

```

```

Hardware Mode = Duplex
Configured Redundancy Mode = Stateful Switchover
Operating Redundancy Mode = Stateful Switchover
Maintenance Mode = Disabled
Communications = Up

```

## Current Processor Information :

```

-----
Active Location = slot 5
Current Software state = ACTIVE
Uptime in current state = 9 minutes
Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

```

## Peer Processor Information :

```

-----
Standby Location = slot 6
Current Software state = STANDBY HOT
Uptime in current state = 2 minutes
Image Version = Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3
Switch Software (cat4500e-UNIVERSALK9-M), Version 03.00.00.1.68 CISCO UNIVERSAL
DEVELOPMENT K10 IOSD TEST VERSION
Copyright (c) 1986-2010 by Cisco Systems, Inc.
Compiled Sun 29-Aug-10 03:57 by gsbuprod
Configuration register = 0x2920

```

```

Switch# issu changeversion 5 bootflash:y.bin 6 slavebootflash:y at 16:30
% 'issu changeversion' was executed at [ Apr 12 16:27:43 ].
% The planned ISSU changeversion is to occur in (hh:mm:ss) [ 00:03:00 ] at [ Apr 12
16:30:43 ].
% Current system time: [ Apr 12 16:27:43 ]
% Planned upgrade image: bootflash:y.bin
% To cancel the planned upgrade, please execute 'issu abortversion'

```

Switch# **show issu state detail**

```

Slot = 5
RP State = Active
ISSU State = Init
Changeversion = TRUE
Operating Mode = Stateful Switchover
Current Image = bootflash:x.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

Slot = 6
RP State = Standby
ISSU State = Init
Changeversion = TRUE
Operating Mode = Stateful Switchover
Current Image = bootflash:x.bin
Pre-ISSU (Original) Image = N/A
Post-ISSU (Targeted) Image = N/A

```

## ISSU プロセス中のソフトウェア アップグレードの中断

**issu abortversion** コマンドを入力して、どの段階においても手動で ISSU プロセスを中断できます (**issu commitversion** コマンドを入力する前)。また、ソフトウェアによる障害の検知によっても、ISSU プロセスは自動的に中断します。



(注) スタンバイ スーパーバイザ エンジンが Standby Hot ステートに移行する前に、**issu abortversion** コマンドを発行すると、トラフィックが中断する可能性があります。

**issu loadversion** コマンドを入力した後にプロセスを打ち切ると、スタンバイ スーパーバイザ エンジンがリセットされ、元のソフトウェアがリロードされます。

**issu runversion** または **issu acceptversion** コマンドのいずれかを入力したあとにプロセスが中断された場合は、元のソフトウェア バージョンを引き続き実行している新しいスタンバイ スーパーバイザ エンジンで 2 回目のスイッチオーバーが実行されます。新しいソフトウェアを実行していたスーパーバイザ エンジンがリセットされ、元のソフトウェア バージョンがリロードされます。



(注) アクティブ スーパーバイザ エンジン上で **abortversion** コマンドを入力する前に、スタンバイ スーパーバイザ エンジンが完全にブートしていることを確認します。

ここでは、**issu commitversion** コマンドを使用して ISSU プロセスを完了する前に、ISSU プロセスを中断する方法について説明します。

アクティブ スーパーバイザ エンジンで次の作業を行います。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。  • プロンプトが表示されたら、パスワードを入力します。
ステップ2	Switch# <b>issu abortversion active slot</b> [active-image-new]	進行中の ISSU アップグレードまたはダウングレードプロセスをキャンセルし、ルータのステートを、プロセスが開始する前のステートに戻します。

次に、スロット番号 2 (現在アクティブなスーパーバイザ エンジンのスロット) の ISSU プロセスを中断する例を示します。

```
Switch> enable
Switch# issu abortversion 2
```

## アップグレード問題を回避するためのロールバック タイマーの設定

Cisco IOS ソフトウェアは、新しくアクティブになったスーパーバイザ エンジンとスタンバイ スーパーバイザ エンジンとの通信がアップグレードによって、切断された状態になるのを回避するために、ISSU ロールバック タイマーを維持します。

新しいソフトウェアがコミットされていない場合、または Run Version モード中にスイッチへの接続が失われた場合にユーザが待つ必要がないように、ロールバック タイマーを 45 分 (デフォルト) 以内に設定することもできます。新しいイメージをコミットする前に新しい Cisco IOS ソフトウェアの動作を確認するための十分な時間が必要な場合は、45 分を超える値にロールバック タイマーを設定することもできます。



(注) 有効なタイマー値の範囲は、0 ～ 7200 秒 (2 時間) です。0 秒の値を設定すると、ロールバック タイマーはディセーブルになります。

ISSU プロセスが正常に行われていることに満足し、現在の状態を保つ場合は、**issu acceptversion** コマンドを入力することにより、承諾したことを示す必要があります。これにより、ロールバック タイマーが停止します。**issu acceptversion** コマンドを入力することは、ISSU プロセスを進めるうえで非常に重要です。

この段階で **issu commitversion** コマンドを入力することは、**issu acceptversion** コマンドと **issu commitversion** コマンドの両方を入力することと同じです。今のところ現在の状態で実行しない予定で、新しいソフトウェア バージョンに満足している場合は、**issu commitversion** コマンドを使用します。



(注) ロールバック タイマーは、ISSU の Init ステートでだけ設定できます。

ロールバック タイマーを設定するには、次の作業を行います。

	コマンドまたはアクション	目的
ステップ 1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ 2	Switch# <b>configure terminal</b>	グローバル コンフィギュレーション モードを開始します。
ステップ 3	Switch(config)# <b>issu set rollback-timer</b> <i>hh:mm:ss</i>	ロールバック タイマー値を設定します。
ステップ 4	Switch(config)# <b>exit</b>	特権 EXEC モードに戻ります。
ステップ 5	Switch# <b>show issu rollback-timer</b>	ISSU ロールバック タイマーの現在の設定を表示します。

次に、ロールバック タイマーを 3600 秒に設定する例を示します。

```
Switch> enable
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# issu set rollback-timer 3600
% Rollback timer value set to [ 3600 ] seconds

Switch(config)# exit

Switch# show issu rollback-timer
Rollback Process State = Not in progress
Configured Rollback Time = 60:00
```

次の例で示すように、ロールバック タイマーを LV ステートで設定することはできません。

```
Switch# show issu state detail
Slot = 1
RP State = Active
ISSU State = Load Version
Boot Variable = bootflash:old_image,12
Operating Mode = RPR
Primary Version = bootflash:old_image
Secondary Version = bootflash:new_image
Current Version = bootflash:old_image
```

```

Slot = 2
RP State = Standby
ISSU State = Load Version
Boot Variable = bootflash:new_image,12;bootflash:old_image,12
Operating Mode = RPR
Primary Version = bootflash:old_image
Secondary Version = bootflash:new_image
Current Version = bootflash:new_image

Switch# show issu rollback-timer
Rollback Process State = Not in progress
Configured Rollback Time = 60:00

Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# issu set rollback-timer 20
% ISSU state should be [ init ] to set the rollback timer

```

## ISSU 互換性マトリクス情報の表示

ISSU 互換性マトリクスには、該当するバージョンにかかわる他のソフトウェア イメージに関する情報が含まれます。この互換性マトリクスには、2つのソフトウェア バージョン（1つは、アクティブ スーパーバイザ エンジンで実行されるソフトウェア バージョンで、もう一方はスタンバイ スーパーバイザ エンジンで実行されるソフトウェア バージョン）の互換性が示され、これによって、システムは実現可能な最も高度な動作モードを判別できます。この情報は、ユーザが ISSU を使用するかどうかを判断する場合にも役立ちます。

ISSU 互換性マトリクスに関する情報を表示するには、次の作業を行います。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>プロンプトが表示されたら、パスワードを入力します。</li> </ul>
ステップ2	Switch# <b>show issu comp-matrix</b> { <b>negotiated</b>   <b>stored</b>   <b>xml</b> }	ISSU 互換性マトリクスに関する情報を表示します。 <ul style="list-style-type: none"> <li><b>negotiated</b> : ネゴシエートされた互換性マトリクス情報を表示します。</li> <li><b>stored</b> : 保存された互換性マトリクス情報を表示します。</li> <li><b>xml</b> : ネゴシエートされた互換性マトリクス情報を XML 形式で表示します。</li> </ul>

次に、ネゴシエートされた互換性マトリクスに関する情報を表示する例を示します。

```

Switch> enable
Switch# show issu comp-matrix negotiated

CardType: WS-C4507R(112), Uid: 2, Image Ver: 12.2(31)SGA
Image Name: cat4500-ENTSERVICES-M

Cid      Eid      Sid      pSid     pUid     Compatibility
=====
2        1        262151   3        1        COMPATIBLE
3        1        262160   5        1        COMPATIBLE
4        1        262163   9        1        COMPATIBLE

```

## ISSU プロセスの実行

5	1	262186	25	1	COMPATIBLE
7	1	262156	10	1	COMPATIBLE
8	1	262148	7	1	COMPATIBLE
9	1	262155	1	1	COMPATIBLE
10	1	262158	2	1	COMPATIBLE
11	1	262172	6	1	COMPATIBLE
100	1	262166	13	1	COMPATIBLE
110	113	262159	14	1	COMPATIBLE
200	1	262167	24	1	COMPATIBLE
2002	1	-	-	-	UNAVAILABLE
2003	1	262185	23	1	COMPATIBLE
2004	1	262175	16	1	COMPATIBLE
2008	1	262147	26	1	COMPATIBLE
2008	1	262168	27	1	COMPATIBLE
2010	1	262171	32	1	COMPATIBLE
2012	1	262180	31	1	COMPATIBLE
2021	1	262170	41	1	COMPATIBLE
2022	1	262152	42	1	COMPATIBLE
2023	1	-	-	-	UNAVAILABLE
2024	1	-	-	-	UNAVAILABLE
2025	1	-	-	-	UNAVAILABLE
2026	1	-	-	-	UNAVAILABLE
2027	1	-	-	-	UNAVAILABLE
2028	1	-	-	-	UNAVAILABLE
2054	1	262169	8	1	COMPATIBLE
2058	1	262154	29	1	COMPATIBLE
2059	1	262179	30	1	COMPATIBLE
2067	1	262153	12	1	COMPATIBLE
2068	1	196638	40	1	COMPATIBLE
2070	1	262145	21	1	COMPATIBLE
2071	1	262178	11	1	COMPATIBLE
2072	1	262162	28	1	COMPATIBLE
2073	1	262177	33	1	COMPATIBLE
2077	1	262165	35	1	COMPATIBLE
2078	1	196637	34	1	COMPATIBLE
2079	1	262176	36	1	COMPATIBLE
2081	1	262150	37	1	COMPATIBLE
2082	1	262161	39	1	COMPATIBLE
2083	1	262184	20	1	COMPATIBLE
2084	1	262183	38	1	COMPATIBLE
4001	101	262181	17	1	COMPATIBLE
4002	201	262164	18	1	COMPATIBLE
4003	301	262182	19	1	COMPATIBLE
4004	401	262146	22	1	COMPATIBLE
4005	1	262149	4	1	COMPATIBLE

Message group summary:

Cid	Eid	GrpId	Sid	pSid	pUId	Nego Result
2	1	1	262151	3	1	Y
3	1	1	262160	5	1	Y
4	1	1	262163	9	1	Y
5	1	1	262186	25	1	Y
7	1	1	262156	10	1	Y
8	1	1	262148	7	1	Y
9	1	1	262155	1	1	Y
10	1	1	262158	2	1	Y
11	1	1	262172	6	1	Y
100	1	1	262166	13	1	Y
110	113	115	262159	14	1	Y
200	1	1	262167	24	1	Y
2002	1	2	-	-	-	N - did not negotiate
2003	1	1	262185	23	1	Y
2004	1	1	262175	16	1	Y

2008	1	1	262147	26	1	Y
2008	1	2	262168	27	1	Y
2010	1	1	262171	32	1	Y
2012	1	1	262180	31	1	Y
2021	1	1	262170	41	1	Y
2022	1	1	262152	42	1	Y
2023	1	1	-	-	-	N - did not negotiate
2024	1	1	-	-	-	N - did not negotiate
2025	1	1	-	-	-	N - did not negotiate
2026	1	1	-	-	-	N - did not negotiate
2027	1	1	-	-	-	N - did not negotiate
2028	1	1	-	-	-	N - did not negotiate
2054	1	1	262169	8	1	Y
2058	1	1	262154	29	1	Y
2059	1	1	262179	30	1	Y
2067	1	1	262153	12	1	Y
2068	1	1	196638	40	1	Y
2070	1	1	262145	21	1	Y
2071	1	1	262178	11	1	Y
2072	1	1	262162	28	1	Y
2073	1	1	262177	33	1	Y
2077	1	1	262165	35	1	Y
2078	1	1	196637	34	1	Y
2079	1	1	262176	36	1	Y
2081	1	1	262150	37	1	Y
2082	1	1	262161	39	1	Y
2083	1	1	262184	20	1	Y
2084	1	1	262183	38	1	Y
4001	101	1	262181	17	1	Y
4002	201	1	262164	18	1	Y
4003	301	1	262182	19	1	Y
4004	401	1	262146	22	1	Y
4005	1	1	262149	4	1	Y

## List of Clients:

Cid	Client Name	Base/Non-Base
2	ISSU Proto client	Base
3	ISSU RF	Base
4	ISSU CF client	Base
5	ISSU Network RF client	Base
7	ISSU CONFIG SYNC	Base
8	ISSU ifIndex sync	Base
9	ISSU IPC client	Base
10	ISSU IPC Server client	Base
11	ISSU Red Mode Client	Base
100	ISSU rfs client	Base
110	ISSU ifs client	Base
200	ISSU Event Manager client	Base
2002	CEF Push ISSU client	Base
2003	ISSU XDR client	Base
2004	ISSU SNMP client	Non-Base
2008	ISSU Tableid Client	Base
2010	ARP HA	Base
2012	ISSU HSRP Client	Non-Base
2021	XDR Int Priority ISSU cli	Base
2022	XDR Proc Priority ISSU cl	Base
2023	FIB HWIDB ISSU client	Base
2024	FIB IDB ISSU client	Base
2025	FIB HW subblock ISSU clie	Base
2026	FIB SW subblock ISSU clie	Base
2027	Adjacency ISSU client	Base
2028	FIB IPV4 ISSU client	Base
2054	ISSU process client	Base

```

2058      ISIS ISSU RTR client      Non-Base
2059      ISIS ISSU UPD client      Non-Base
2067      ISSU PM Client            Base
2068      ISSU PAGP_SWITCH Client  Non-Base
2070      ISSU Port Security client Non-Base
2071      ISSU Switch VLAN client  Non-Base
2072      ISSU dot1x client         Non-Base
2073      ISSU STP                  Non-Base
2077      ISSU STP MSTP            Non-Base
2078      ISSU STP IEEE             Non-Base
2079      ISSU STP RSTP            Non-Base
2081      ISSU DHCP Snooping client Non-Base
2082      ISSU IP Host client       Non-Base
2083      ISSU Inline Power client  Non-Base
2084      ISSU IGMP Snooping client Non-Base
4001      ISSU C4K Chassis client  Base
4002      ISSU C4K Port client     Base
4003      ISSU C4K Rkios client    Base
4004      ISSU C4K HostMan client  Base
4005      ISSU C4k GaliosRedundancyBase

```

次に、保存された互換性マトリクスに関する情報を表示する例を示します。

```
Switch# show issu comp-matrix stored
```

```
Number of Matrices in Table = 1
```

```
(1) Matrix for cat4500-ENTSERVICES-M(112) - cat4500-ENTSERVICES-M(112)
```

```

=====
Start Flag (0xDEADBABE)

      My Image ver: 12.2(53)SG
      Peer Version  Compatibility
      -----
      12.2(31)SGA5      Base(2)
      12.2(44)SG        Base(2)
      12.2(31)SGA6      Base(2)
      12.2(31)SGA7      Base(2)
      12.2(46)SG        Base(2)
      12.2(44)SG1       Base(2)
      12.2(31)SGA8      Base(2)
      12.2(50)SG        Dynamic(0)
      12.2(31)SGA9      Base(2)
      12.2(50)SG1       Dynamic(0)
      12.2(50)SG2       Dynamic(0)
      12.2(52)SG        Dynamic(0)
      12.2(31)SGA10     Base(2)
      12.2(50)SG3       Dynamic(0)
      12.2(53)SG        Comp(3)

```

Dynamic(0) が Dynamic Image Version Compatibility (DIVC) 機能付きの Cisco IOS Release 12.2(50) SG で導入されました。DIVC では、Incomp(1)、Base(2)、または Comp(3) ではなく Dynamic(0) が保存されます。互換性は、2 の異なる DIVC 対応イメージが ISSU 時にアクティブ スーパーバイザ エンジンおよびスタンバイ スーパーバイザ エンジン上で実行されているときのランタイム時に決定されず。

Catalyst 4500 スイッチでは、保存された互換性マトリクスの値 Dynamic(0) は通常、2 つのイメージ間のロールバック ネゴシエーション時に、Base(2) または Comp(3) になります。保存された互換性マトリクスに他のイメージ名があるかぎり、Incomp(1) は確認できません。



## ISSU 互換性マトリクス情報の表示

ISSU 互換性マトリクスには、他の IOS XE ソフトウェア リリースと該当するバージョンに関する情報が含まれます。この互換性マトリクスには、2 つのソフトウェア バージョン (1 つは、アクティブ スーパーバイザ エンジンで実行されるソフトウェア バージョンで、もう一方はスタンバイ スーパーバイザ エンジンで実行されるソフトウェア バージョン) の互換性が示され、これによって、システムは実現可能な最も高度な動作モードを判別できます。この情報は、ユーザが ISSU を使用するかどうかを判断する場合にも役立ちます。

ここでは、ISSU 互換性マトリクスに関する情報を表示する方法を示します。

	コマンドまたはアクション	目的
ステップ1	Switch> <b>enable</b>	特権 EXEC モードをイネーブルにします。プロンプトが表示されたら、パスワードを入力します。
ステップ2	Switch# <b>show issu comp-matrix</b> { <b>negotiated</b>   <b>stored</b>   <b>xml</b> }	ISSU 互換性マトリクスに関する情報を表示します。 <ul style="list-style-type: none"> <li>• <b>negotiated</b> : ネゴシエートされた互換性マトリクス情報を表示します。</li> <li>• <b>stored</b> : 保存された互換性マトリクス情報を表示します。</li> <li>• <b>xml</b> : ネゴシエートされた互換性マトリクス情報を XML 形式で表示します。</li> </ul> <b>(注)</b> これらのコマンドは、IOSd プロセス内部のデータしか表示しません。システム全体に関する情報を表示するには、 <b>show package compatibility</b> を使用します。
ステップ3	Switch# <b>show package compatibility</b>	システムのすべてのクライアント互換性に関する情報を表示します。

次に、ネゴシエートされた互換性マトリクスに関する情報を表示する例を示します。

```
Switch> enable
Switch# show issu comp-matrix negotiated

CardType: WS-C4507R-E(182), Uid: 4, Image Ver: 03.00.00.1.68
Image Name: cat4500e-UNIVERSALK9-M

Cid      Eid      Sid      pSid    pUid     Compatibility
=====
2        1        131078   3        3        COMPATIBLE
3        1        131100   5        3        COMPATIBLE
4        1        131123   9        3        COMPATIBLE
.....
.....

Message group summary:
Cid      Eid      GrpId    Sid      pSid    pUid     Nego Result
=====
2        1        1        131078   3        3        Y
3        1        1        131100   5        3        Y
4        1        1        131123   9        3        Y
.....
.....
```

```
List of Clients:
Cid      Client Name          Base/Non-Base
=====
2        ISSU Proto client   Base
3        ISSU RF             Base
4        ISSU CF client      Base
.....
.....
```

次に、保存された互換性マトリクスに関する情報を表示する例を示します。

```
Switch# show issu comp-matrix stored

Number of Matrices in Table = 1

(1) Matrix for cat4500e-ENTSERVICESK9-M(182) - cat4500ex-ENTSERVICESK9-M(182)
=====
Start Flag (0xDEADBABE)

My Image ver: 03.01.00.SG
Peer Version  Compatibility
-----
03.01.00.SG   Comp(3)

Switch#
```

Dynamic Image Version Compatibility (DIVC) では、Incomp(1)、Base(2)、または Comp(3) ではなく Dynamic(0) が保存されます。互換性は、2 の異なる DIVC 対応イメージが ISSU 時にアクティブスーパーバイザエンジンおよびスタンバイスーパーバイザエンジン上で実行されているときのランタイム時に決定されます。

Catalyst 4500 スイッチでは、保存された互換性マトリクスの値 Dynamic(0) は通常、2 つのソフトウェアイメージ間のランタイムネゴシエーション時に、Base(2) または Comp(3) になります。保存された互換性マトリクスに他のイメージ名があるかぎり、Incomp(1) は確認できません。

次に、ネゴシエートされた非 IOSd クライアントに関する情報を表示する例を示します。

```
Switch# show package compatibility

PackageName      PeerPackageName      ModuleName      Compatibility
-----
rp_base          rp_base              aaa              COMPATIBLE
rp_base          rp_base              aaacommon       COMPATIBLE
rp_base          rp_base              access_policy   COMPATIBLE
rp_base          rp_base              app_sess        COMPATIBLE
rp_base          rp_base              app_sess_ios    COMPATIBLE
rp_base          rp_base              auth_mgr        COMPATIBLE
.....
.....
```

## 関連資料

関連項目	参照先
ISSU の実行	『Cisco IOS Software: Guide to Performing In Service Software Upgrades』
Cisco Nonstop Forwarding (NSF) に関する情報	『Cisco Nonstop Forwarding』  <a href="http://www.cisco.com/en/US/docs/ios/12_2s/feature/guide/fsnsf20s.html">http://www.cisco.com/en/US/docs/ios/12_2s/feature/guide/fsnsf20s.html</a>

関連項目	参照先
ステートフル スイッチオーバー (SSO) に関する情報	『 <i>Stateful Switchover</i> 』 <a href="http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.html">http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/sso120s.html</a>
ISSU クライアントおよび MPLS クライアント	『ISSU MPLS Clients』

