



Cisco IOS ISSU プロセスの設定

冗長システムで稼働している場合、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継続されます。ほとんどのネットワークでは、計画されたソフトウェア アップグレードが大幅なダウンタイムの原因になります。ISSU により、Cisco IOS ソフトウェアが変更される間、パケットの転送が継続されます。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。



(注)

この章のスイッチ コマンドの構文および使用方法の詳細については、『*Catalyst 4500 Series Switch Cisco IOS Command Reference*』および次の URL の関連マニュアルを参照してください。

<http://www.cisco.com/univercd/cc/td/doc/product/software/ios122sr/cr/index.htm>

関連資料

関連トピック	資料のタイトル
ISSU の実行	『 <i>Cisco IOS Software: Guide to Performing In Service Software Upgrades</i> 』
Cisco Nonstop Forwarding (NSF) に関する情報	『 <i>Cisco Nonstop Forwarding</i> 』 http://www.cisco.com/univercd/cc/td/doc/product/software/ios122s/122snwft/release/122s20/fsnsf20s.htm
Stateful Switchover (SSO) に関する情報	『 <i>Stateful Switchover</i> 』 http://www.cisco.com/univercd/cc/td/doc/product/software/ios122s/122snwft/release/122s20/fssso20s.htm
ISSU クライアントおよび MPLS クライアント	『 <i>ISSU MPLS Clients</i> 』

内容

- [ISSU を実行するための前提条件 \(p.5-2\)](#)
- [ISSU の実行に関する制約事項 \(p.5-3\)](#)
- [ISSU の実行に関する情報 \(p.5-3\)](#)
- [ISSU プロセスの実行方法 \(p.5-14\)](#)

ISSU を実行するための前提条件

適用される前提条件は、次のとおりです。

- ISSU を適用できるのは冗長シャーシだけです。
- 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 エンジンの両方がシステムで使用でき、両方のエンジンのタイプが同じであることを確認します (たとえば、WS-X4516-10GE)。
- 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 フラッシュのどちら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。
- 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することができます。

次に、**show redundancy state** コマンドを使用して、冗長ファシリティ ステートに関する情報を表示する例を示します。

```
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#

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 の実行に関する制約事項

適用される制約事項は、次のとおりです。

- 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。システムで実行されている現在の IOS バージョンも ISSU をサポートしている必要があります。
Catalyst 4500 シリーズ スイッチで各種のコマンドを実行することにより、スーパーバイザ エンジンのバージョンと IOS の互換性を判別できます。Cisco Feature Navigator の ISSU アプリケーションを使用して判別することもできます。
- ISSU プロセスの実行中は、ハードウェアに変更を加えないでください。
- ISSU は、Cisco IOS 12.2(31)SGA およびそれ以降のリリースで使用できます。



(注)

すべてのラインカードがサポートされています。

ISSU の実行に関する情報

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。

- [SSO の概要 \(p.5-3\)](#)
- [NSF の概要 \(p.5-5\)](#)
- [ISSU プロセスの概要 \(p.5-6\)](#)
- [ISSU をサポートする Cisco IOS ソフトウェアのバージョン機能 \(p.5-11\)](#)
- [ISSU に対する SNMP サポート \(p.5-13\)](#)
- [Cisco Feature Navigator を使用した互換性の検証 \(p.5-13\)](#)

SSO 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。

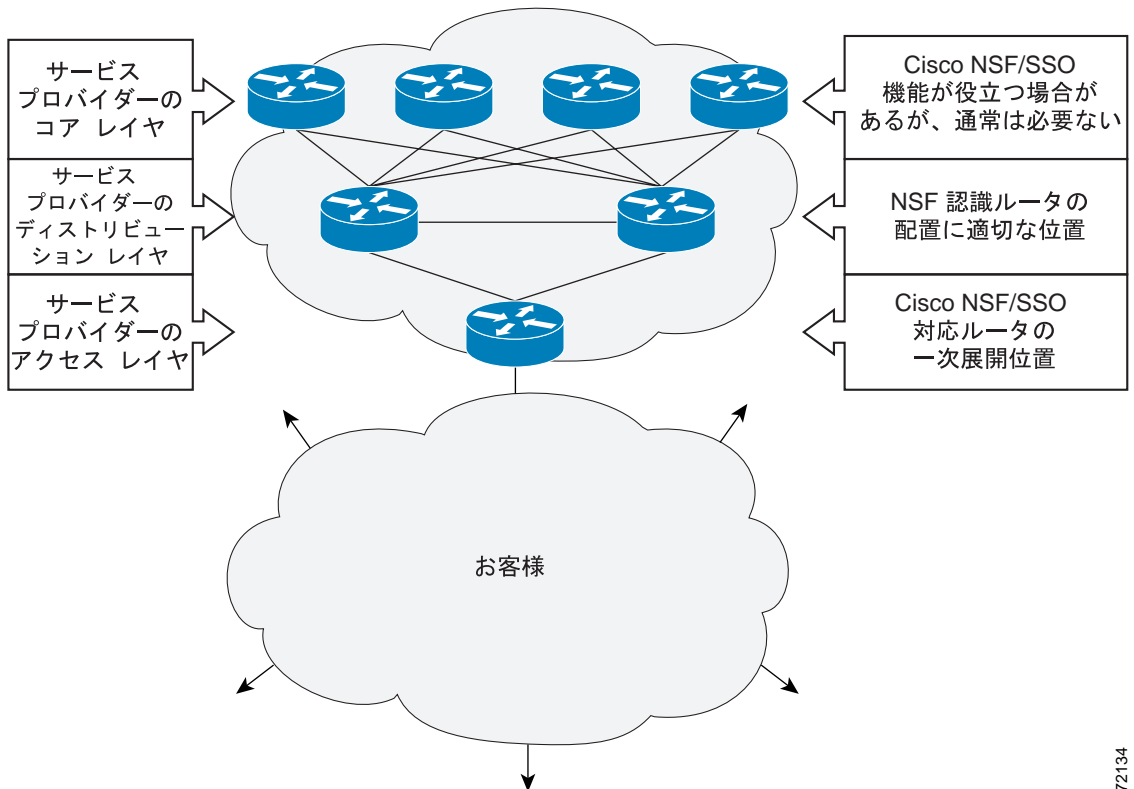
アクティブ スーパーバイザ エンジンが故障した場合、またはネットワーク デバイスから取り外された場合に、アクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンへのスイッチオーバーが行われます。

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 5-1 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。

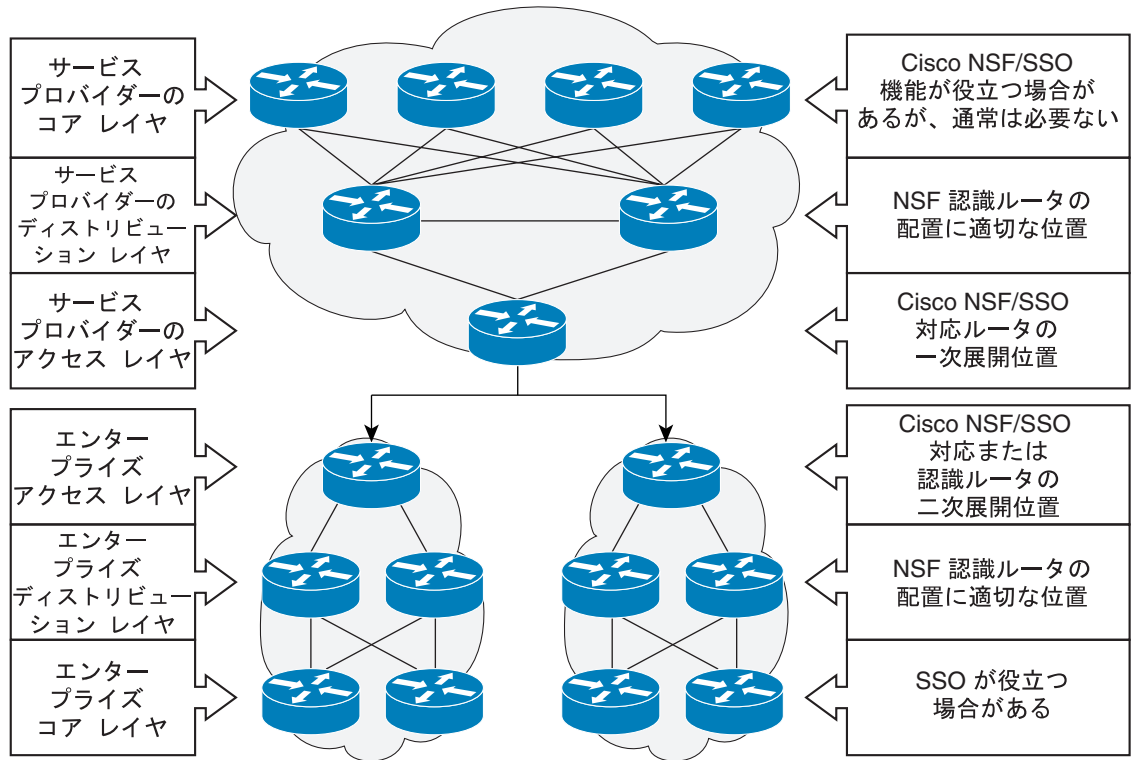
Cisco NSF プロトコルでは、隣接デバイスが Cisco NSF に関与している必要があるため、これらの隣接ディストリビューション レイヤデバイスには、Cisco NSF 認識ソフトウェア イメージをインストールする必要があります。目的に応じて、ネットワークのコア レイヤで Cisco NSF および SSO 機能を展開することもできます。これを行うと、特定の障害が発生した場合のネットワーク機能およびサービスの復元にかかる時間を削減できるため、さらにアベイラビリティが向上します。

図 5-1 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



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 5-2 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 5-2 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



SSO の詳細については、『*Stateful Switchover*』を参照してください。

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 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するルーティング フラップは、ルーティング動作を不安定にし、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。

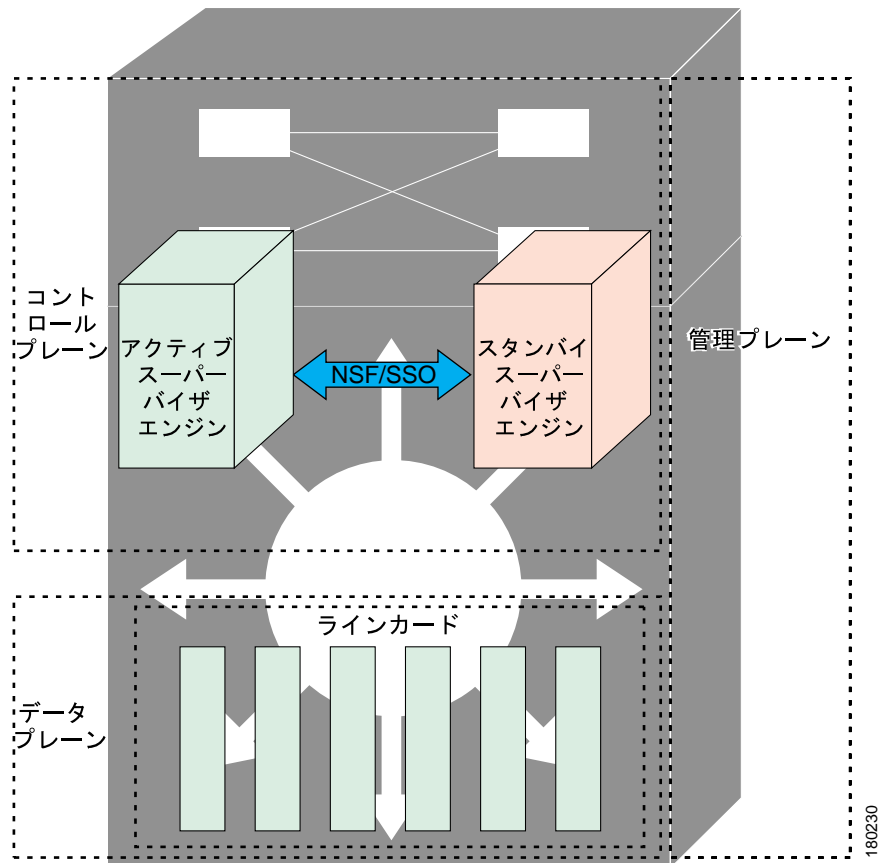
Cisco NSF によって、スイッチオーバー後にルーティング プロトコル情報が復元される間、データの packets の転送が既知のルートで続行されます。Cisco NSF を使用すると、ピア ネットワーキング デバイスでルーティング フラップが発生することがありません。スイッチオーバー時に、故障したアクティブ スーパーバイザ エンジンからスタンバイ スーパーバイザ エンジンが制御を引き継ぐ間も、データ トラフィックが転送されます。Cisco NSF 動作で重要なのは、スイッチオーバー時に物理リンクがアップの状態を維持できる点と、アクティブ スーパーバイザ エンジン上の Forwarding 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 プロセスにより、Cisco IOS ソフトウェア アップグレードまたはダウングレードを実行している間、パケットの転送が継続されます（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 5-8 \[p.5-11\]](#) 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 5-3](#) 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。これには、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、IOS のこれらの 2 つバージョンが異なるフィーチャセットとコマンドをサポートしていても、ISSU は 2 つのスーパーバイザ エンジンを同期させたまま維持します。

図 5-3 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

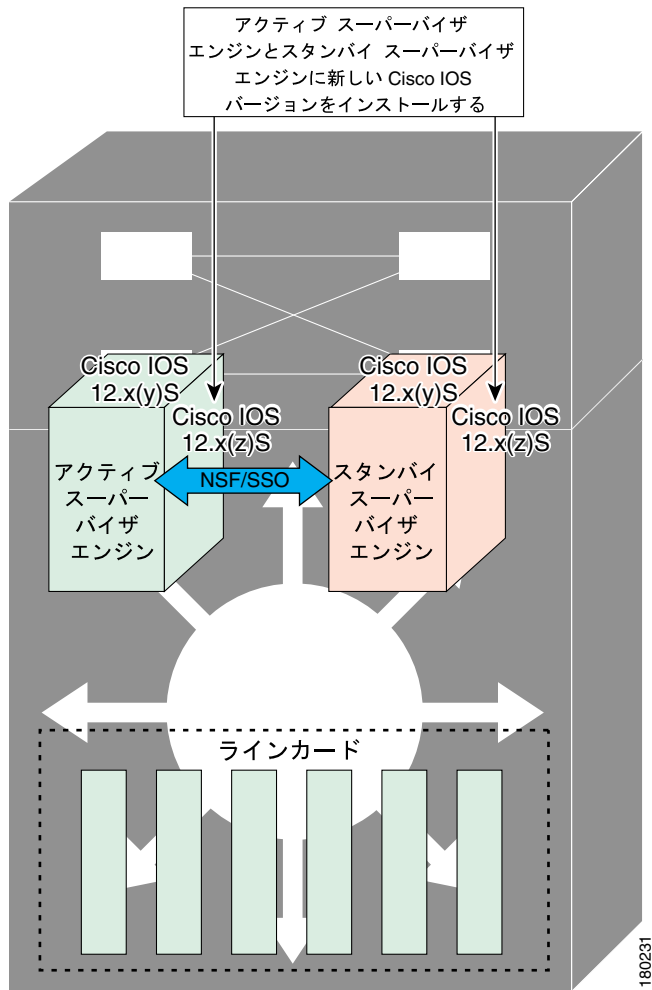


ISSU 対応スイッチは、2 つのスーパーバイザ エンジン（アクティブとスタンバイ）および 1 つまたは複数のラインカードで構成されています。ISSU プロセスを開始する前に、両方のスーパーバイザ エンジンのファイルシステムに Cisco IOS ソフトウェアをコピーします（[図 5-4](#) を参照）。



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

図 5-4 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



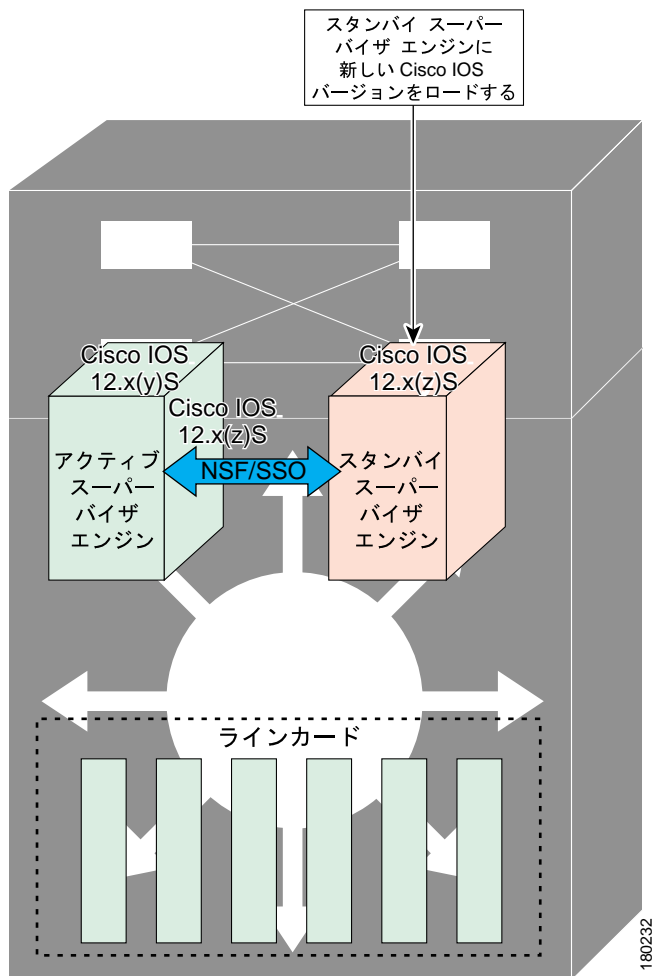
両ファイル システムに Cisco IOS ソフトウェアをコピーしたあと、新しい Cisco IOS ソフトウェアバージョンをスタンバイ スーパーバイザエンジンにロードします (図 5-5 を参照)。



(注)

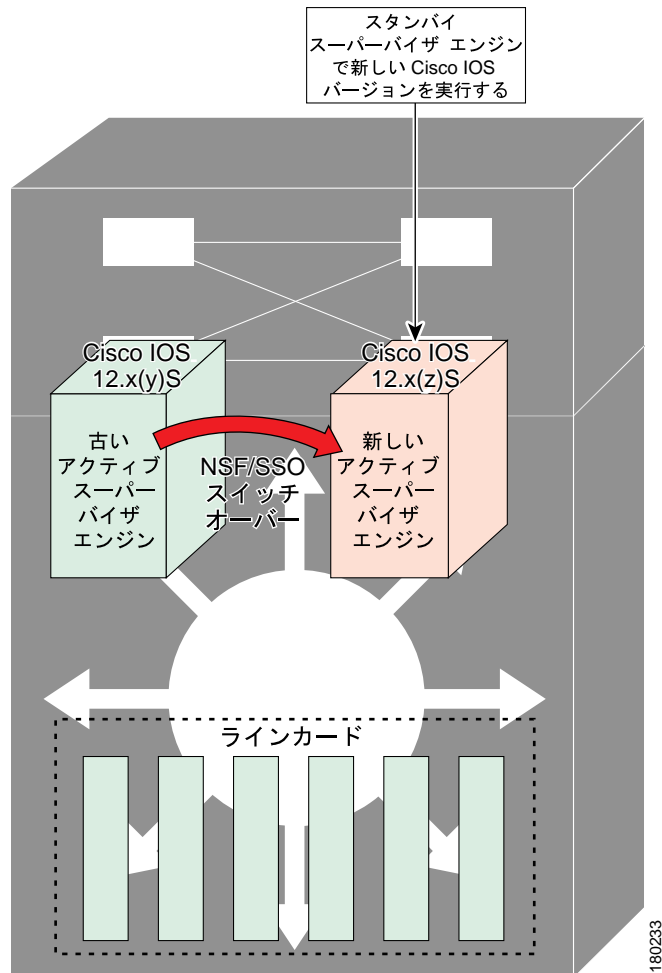
ISSU 機能がないと、アクティブ スーパーバイザエンジンとスタンバイ スーパーバイザエンジンが 2 つの異なる IOS イメージバージョンを実行している場合に、両者間で SSO/NSF 機能は動作しません。

図 5-5 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/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 5-6 を参照）。

図 5-6 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 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 5-7 を参照)。

図 5-7 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

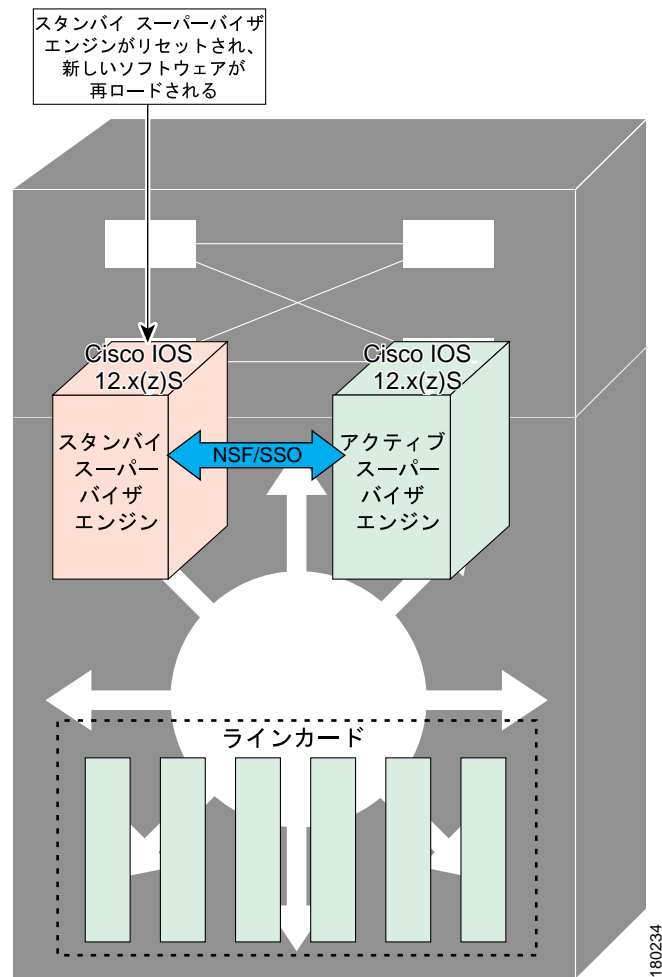
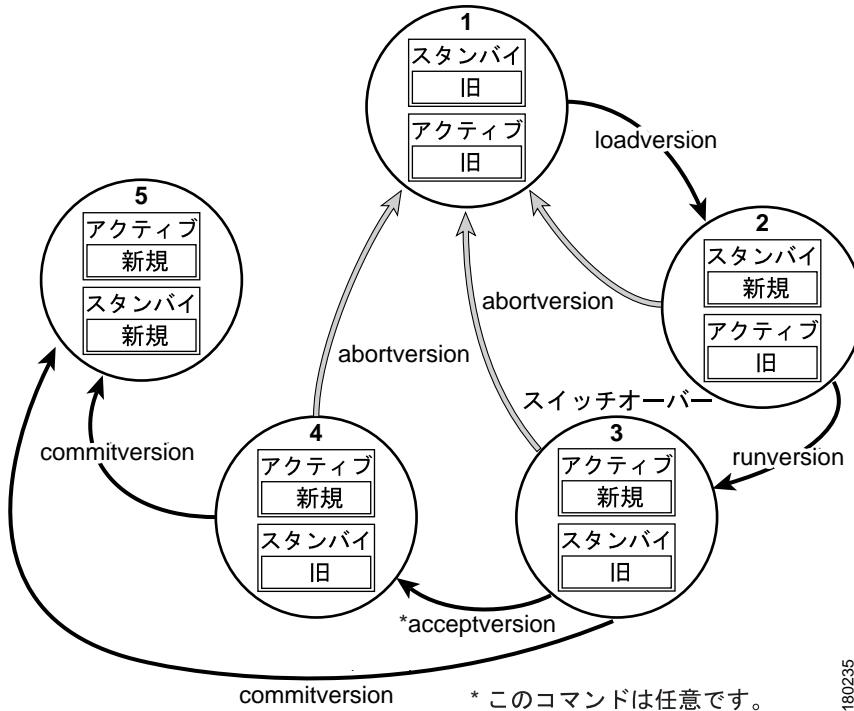


図 5-8 に、ISSU プロセス中の各ステップを示します。

図 5-8 ISSU プロセス中の各ステップ



ISSU をサポートする Cisco IOS ソフトウェアのバージョン機能

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にバージョン 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。バージョンが同じでないと、冗長モードが RPR に変更されます。ISSU 機能を使用した場合、Cisco IOS イメージが同じでなくても、互換性のあるリリース レベルであれば両方のイメージを SSO モードで相互運用し、パケットの転送を継続したまま、ソフトウェアのアップグレードを行うことが可能になります。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。そのため、互換性マトリクスで、問題のバージョンにかかわる他のイメージに関する情報が定義されます。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。

Cisco IOS インフラストラクチャが内部的に変更されて、ISSU とともにサブシステムバージョンが行われるように再設計されました。Cisco IOS サブシステムは、フィーチャセットおよびソフトウェア コンポーネントのグループ化に対応しています。スーパーバイザ エンジン間でステート情報を維持する機能またはサブシステムは、HA 認識または、SSO クライアントです。ISSU フレームワークと呼ばれるメカニズムまたは ISSU プロトコルによって、Cisco IOS ソフトウェア内のサブ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 エンジン間で通信を行って、スーパーバイザ エンジン間の通信のメッセージ バージョンをネゴシエーションすることができます。内部では、HA を認識するすべての NSF/SSO 対応アプリケーションまたはサブシステムが、このプロトコルに従って、異なるソフトウェア バージョンのピアとの通信を確立する必要があります（動作モードの詳細については、『*Stateful Switchover*』を参照してください）。

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せん。

- **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ビス アップグレードまたはダウングレードが行われても、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ビス 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、IOS バージョンが旧式から新規に移行される際に、一部のサブシステムがステートを常に維持することができません。マトリクス エントリでは、このようなイメージに対して **Base-level compatible (B)** が指定されます。
- **Incompatible** (互換性がない) — SSO が正常に機能するためには、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ビス 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 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、指定されたサポート ウィンドウ内の他のすべての Cisco IOS ソフトウェア バージョンに対してある Cisco IOS ソフトウェア イメージが持つ互換性の関係（たとえば、イメージが「認識」できるすべてのソフトウェア バージョン）が示され、各イメージにデータが格納された状態でリリースされます。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でのみ使用できるので、確認する場合にのみ有効です。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 (Management Information Base; 管理情報ベース) 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 エンジン間送受信機能を行います。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のみ、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
- そのイメージと互換性があるイメージを確認する
- 2 つのイメージを比較して、イメージの互換性レベル (Compatible、Base-level compatible、および Incompatible) を理解する
- 2 つのイメージを比較して、各 ISSU クライアントのクライアント互換性を参照する
- 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 プロセスの実行方法

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動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 5-8 \(p.5-11\)](#) を参照してください。

ISSU プロセスの実行中は、次の制約事項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。
- ダウングレードを行う場合、Cisco IOS ソフトウェア イメージのダウングレード 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。

ここでは、次の内容について説明します。

- [ISSU ソフトウェア インストールの確認 \(p.5-14\)](#)
- [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 \(p.5-17\)](#) (必須)
- [スタンバイ スーパーバイザ エンジンへの切り替え \(p.5-20\)](#) (必須)
- [ISSU ロールバック タイマーの停止 \(任意\) \(p.5-23\)](#) (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 \(p.5-24\)](#)
- [ISSU プロセス中のソフトウェア アップグレードの中断 \(p.5-26\)](#)
- [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 \(p.5-27\)](#)
- [ISSU 互換性マトリクス情報の表示 \(p.5-28\)](#)

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 — スタンバイ スーパーバイザ エンジンがリセットされている間のこのエンジンの状態
- 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、もう一方はスタンバイ) の初期ステートです。ISSU プロセスが完了したあとの最終ステートでもあります。
- 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ています。
- 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ます。新しくアクティブになったスーパーバイザ エンジンが現在新しい Cisco IOS ソフトウェア イメージを実行しています。
- System Reset (SR) ステート — このステートは Init ステートに達する前に **issu abortversion** コマンドを実行した場合、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に、発生します。

show コマンドを入力して、ISSU プロセス中のステートに関する情報を取得して、ISSU ソフトウェア インストールを確認できます。

手順の要約

1. **enable**
2. **show issu state [detail]**
3. **show redundancy**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none">• 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# show issu state [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。
ステップ 3	Switch# show redundancy	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 ステータスの確認

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 ステータスで、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 ステータス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。
- (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タスを判別します。

ISSU プロセスの実行方法

- 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 冗長モードを開始し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 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 エンジンで次の手順を実行します。

手順の要約

1. `enable`
2. `issu loadversion active-slot active-image-new standby-slot standby-image-new [forced]`
3. `show issu state [detail]`
4. `show redundancy[states]`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> <code>enable</code>	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# <code>issu loadversion active-slot active-image-new standby-slot standby-image-new [forced]</code>	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)。 issu loadversion 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リロードされます。 forced 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# <code>show issu state [detail]</code>	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 モードを開始していることを確認します。 issu loadversion 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。 show issu state 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# <code>show redundancy [states]</code>	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 ステートで起動し、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 エンジンで次の手順を実行します。

手順の要約

1. **enable**
2. **issu runversion standby-slot [standby-image-new]**
3. **show issu state [detail]**
4. **show redundancy[states]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	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# issu runversion <i>standby-slot [standby-image-new]</i>	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。 issu runversion コマンドを入力すると SSO のスイッチオーバーが実行され、設定されている場合は NSF プロシージャが起動します。
ステップ 3	Switch# show issu state [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# show redundancy [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 エンジンの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]
```

A switchover happens at this point. At the new active supervisor engine, do the following after old active supervisor engine comes up as standby.

```
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 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 **commitversion** コマンドを実行します）。
- 45 分間のロールバック タイマー ウィンドウ内に次のステップ ([**acceptversion**] 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。

手順の要約

1. **enable**
2. **issu acceptversion active-slot-number [active-slot-number]**
3. **show issu state [detail]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	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# issu acceptversion active-slot [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 issu acceptversion 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# show issu rollback-timer	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] 情報に、自動ロールバックが行われるまでの時間が示されています。

```
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 エンジンで次の手順を実行します。

手順の要約

1. `enable`
2. `issu commitversion standby-slot [standby-image-new]`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> <code>enable</code>	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> • 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# <code>issu commitversion standby-slot-number [standby-image-new]</code>	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。
ステップ 3	Switch# <code>show redundancy [states]</code>	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# <code>show issu state [detail]</code>	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` コマンドを発行したあと、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。

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 コマンドを発行して、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** コマンドを発行する前)。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 コマンドを発行したあとにプロセスを手動で中断した場合、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 回め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)

アクティブなスーパーバイザのコマンドで **abortversion** コマンドを発行する前に、スタンバイ スーパーバイザ エンジンが完全に起動され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実行します。

手順の要約

1. **enable**
2. **issu abortversion active-slot [active-image-new]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	特権 EXEC モードをイネーブルにします。
ステップ 2	Switch# issu abortversion active slot [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、ロールバック タイマーを 45 分以上に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** コマンドを実行することにより、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。そのため、**issu acceptversion** コマンドを入力することは、ISSU プロセスを進めるのに極めて重要です。

この段階で **issu commitversion** コマンドを実行することは、**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。現在の状態で一定期間実行しない予定で、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 ステートでのみ設定できます。

ここでは、ロールバック タイマーを設定する方法について説明します。

手順の要約

1. **enable**
2. **configure terminal**
3. **issu set rollback-timer hh::mm::ss**
4. **show issu rollback-timer**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	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# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	Switch(config)# issu set rollback-timer hh::mm::ss	ロールバック タイマー値を設定します。
ステップ 4	Switch(config)# exit	特権 EXEC モードに戻ります。
ステップ 5	Switch# show issu rollback-timer	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。

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。

手順の要約

1. `enable`
2. `show issu comp-matrix {negotiated | stored | xml}`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Switch> enable	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> 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# show issu comp-matrix { negotiated stored / xml }	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"> negotiated — 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。 stored — 保存された互換性マトリクス情報を表示します。 xml — 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。

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
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

ISSU プロセスの実行方法

```

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 GaliosRedundancy	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 cat4500-ENTSERVICES-M(112) - cat4500-ENTSERVICES-M(112)
```

```
=====
```

```
Start Flag (0xDEADBABE)
```

```
My Image ver: 12.2(905.7)HAEFT
```

```
Peer Version    Compatability
```

```
-----
```

```
12.2(31)SGA                Base(2)
```

```
12.2(31)SGA1              Base(2)
```

```
12.2(905.7)HAEFT          Comp(3)
```

