



## コンフィギュレーションファイルの管理コンフィギュレーションガイド（Cisco IOS XE Gibraltar 16.10.x 向け）

初版：2012年11月30日

最終更新：2013年3月29日

### シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター

0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>





## 目次

---

### 第 1 章

#### 最初にお読みください 1

---

### 第 2 章

#### コンフィギュレーション ファイルの管理 3

##### 機能情報の確認 3

##### コンフィギュレーション ファイルの管理の前提条件 3

##### コンフィギュレーション ファイルの管理の制約事項 4

##### コンフィギュレーション ファイルの管理について 4

##### コンフィギュレーション ファイルのタイプ 4

##### コンフィギュレーション モードおよびコンフィギュレーション ソースの選択 4

##### CLI を使用したコンフィギュレーション ファイルの変更 5

##### コンフィギュレーション ファイルの場所 5

##### ネットワーク サーバからルータへのコンフィギュレーション ファイルのコピー 6

##### ルータから TFTP サーバへのコンフィギュレーション ファイルのコピー 6

##### ルータから FTP サーバへのコンフィギュレーション ファイルのコピー 7

##### VRF によるファイルのコピー 7

##### NVRAM より大きいコンフィギュレーション ファイル 8

##### コンフィギュレーション ファイルの圧縮 8

##### ネットワークからのコンフィギュレーション コマンドのロード 8

##### パーサー キャッシュの制御 9

##### コンフィギュレーション ファイルをダウンロードするルータの設定 10

##### ネットワークとホストのコンフィギュレーション ファイル 10

##### コンフィギュレーション ファイル情報の管理方法 10

##### コンフィギュレーション ファイル情報の表示 10

##### CLI でのコンフィギュレーション ファイルの変更 11

ルータから TFTP サーバへのコンフィギュレーション ファイルのコピー	13
次の作業	14
ルータから FTP サーバへのコンフィギュレーション ファイルのコピー	14
例	16
次の作業	16
TFTP サーバからルータへのコンフィギュレーション ファイルのコピー	16
次の作業	17
FTP サーバからルータへのコンフィギュレーション ファイルのコピー	18
例	19
次の作業	20
NVRAM より大きいコンフィギュレーション ファイルの保守	20
コンフィギュレーション ファイルの圧縮	20
パーサー キャッシュの管理	21
パーサー キャッシュのクリア	22
パーサー キャッシュのディセーブル化	22
パーサー キャッシュの再イネーブル化	23
次の作業	23
フラッシュ メモリからスタートアップまたは実行コンフィギュレーションへのコンフィ ギュレーション ファイルのコピー	24
FTP サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコ ピー	25
次の作業	26
scp サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー	26
TFTP サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコ ピー	27
スタートアップ コンフィギュレーション ファイルでのコンフィギュレーション コマンド の再実行	28
スタートアップ コンフィギュレーションのクリア	29
指定されたコンフィギュレーション ファイルの削除	30
その他の参考資料	31

第 3 章	コンフィギュレーション生成のパフォーマンス拡張	33
	機能情報の確認	33
	コンフィギュレーション生成のパフォーマンス拡張に関する制限事項	34
	コンフィギュレーション生成のパフォーマンス拡張について	34
	Cisco IOS XE ソフトウェアのコンフィギュレーションストレージ	34
	コンフィギュレーション生成のパフォーマンス強化の利点	34
	コンフィギュレーション生成のパフォーマンス強化の設定方法	35
	コンフィギュレーション生成のパフォーマンス強化の設定	35
	コンフィギュレーション生成のパフォーマンス強化の設定例	36
	コンフィギュレーション生成のパフォーマンス強化の設定例	36
	コンフィギュレーション生成のパフォーマンス強化の確認例	36
	その他の参考資料	36
	コンフィギュレーション生成のパフォーマンス強化の機能情報	38
第 4 章	排他的設定変更アクセスとアクセス セッション ロック	39
	機能情報の確認	39
	設定のロックについて	40
	排他的設定変更アクセスとアクセス セッション ロック	40
	アクセス セッション ロック	40
	排他的設定変更アクセスとアクセス セッション ロックの設定方法	41
	排他的設定変更アクセスとアクセス セッション ロックの有効化	41
	排他的設定変更アクセスの取得	42
	設定ロックのモニタリングとトラブルシューティング	43
	コンフィギュレーションのロックの設定例	44
	自動モードでの排他的ロックの設定例	44
	手動モードでの排他的ロックの設定例	44
	その他の参考資料	44
	排他的設定変更アクセスとアクセス セッション ロックの機能情報	46
第 5 章	コンフィギュレーションの置換とロールバック	49

機能情報の確認	49
コンフィギュレーションの置換とロールバックの前提条件	50
コンフィギュレーションの置換とロールバックの制約事項	50
コンフィギュレーションの置換とロールバックについて	51
コンフィギュレーション アーカイブ	51
コンフィギュレーションの置換	51
コンフィギュレーション ロールバック	53
コンフィギュレーション ロールバック変更確認の操作	53
コンフィギュレーションの置換とロールバックの利点	54
コンフィギュレーションの置換とロールバックの使用方法	54
コンフィギュレーション アーカイブの作成	54
コンフィギュレーションの置換やロールバック操作の実行	56
機能のモニタリングおよびトラブルシューティング	59
コンフィギュレーションの置換とロールバックの設定例	61
コンフィギュレーション アーカイブの作成例	61
保存された Cisco IOS コンフィギュレーション ファイルでの現在の実行コンフィギュレーションの置換：例	62
スタートアップ コンフィギュレーション ファイルの復元：例	62
例：configure confirm コマンドを使用したコンフィギュレーション置換操作の実行	63
コンフィギュレーション ロールバック操作の実行：例	63
その他の参考資料	64
コンフィギュレーションの置換とロールバックの機能情報	66

## 第 6 章

## コンフィギュレーションのコンテキスト差分ユーティリティ 69

機能情報の確認	69
コンフィギュレーションのコンテキスト差分ユーティリティの前提条件	70
コンフィギュレーションのコンテキスト差分ユーティリティの制限事項	70
コンフィギュレーションのコンテキスト差分ユーティリティについて	70
コンフィギュレーションのコンテキスト差分ユーティリティの利点	70
コンフィギュレーションのコンテキスト差分ユーティリティの出力形式	71
コンフィギュレーションのコンテキスト差分ユーティリティの使い方	72

コンフィギュレーションのコンテキスト差分ユーティリティを使用した行ごとのファイル比較の実行	72
コンフィギュレーションのコンテキスト差分ユーティリティの設定例	73
差分操作の例	73
差分比較操作の例	75
その他の参考資料	77
コンフィギュレーションのコンテキスト差分ユーティリティの機能情報	79

---

**第 7 章**
**コンフィギュレーション変更通知およびロギング 81**

機能情報の確認	81
コンフィギュレーション変更通知およびロギングの制約事項	82
コンフィギュレーション変更通知およびロギングについて	82
設定ログ	82
コンフィギュレーション変更通知およびコンフィギュレーション変更ロギング	83
EAL4+ 認証用のコンフィギュレーション ロガーの機能強化	83
コンフィギュレーション変更通知およびロギングの設定方法	84
コンフィギュレーション変更通知およびロギングの設定	84
設定ログ エントリおよび統計の表示	85
設定ログ エントリのクリア	87
ログサイズのリセットによる設定ログの消去	87
設定ログをディセーブルすることによる設定ログのクリア	88
コンフィギュレーション変更通知およびロギングの設定例	90
例：コンフィギュレーション変更通知およびロギングの設定	90
その他の参考資料	90
コンフィギュレーション変更通知およびロギングの機能情報	91

---

**第 8 章**
**コンフィギュレーションパーティショニング 93**

機能情報の確認	94
コンフィギュレーションパーティショニングについて	94
システム実行コンフィギュレーション	94
実行コンフィギュレーションを取得して表示またはコピーする	94

実行コンフィギュレーションをパーティショニングする利点	95
コンフィギュレーションパーティショニング機能を使用するには	95
コンフィギュレーションパーティションの表示	95
コンフィギュレーションパーティショニング機能をディセーブルにする	97
次の作業	98
コンフィギュレーションパーティショニングするためのコンフィギュレーション例	99
コンフィギュレーションパーティションの表示例	99
その他の参考資料	108
コンフィギュレーションパーティショニングの機能情報	110

## 第 9 章

<b>コンフィギュレーションのバージョン管理</b>	<b>111</b>
機能情報の確認	111
コンフィギュレーションのバージョン管理について	111
コンフィギュレーションアーカイブ	111
コンフィギュレーションのバージョン管理の設定方法	112
設定アーカイブの特性の設定	112
コンフィギュレーションのモニタリングとトラブルシューティング	114
コンフィギュレーションのバージョン管理の設定例	117
例：コンフィギュレーションアーカイブの作成	117
その他の参考資料	117
コンフィギュレーションのバージョン管理の機能情報	118

## 第 10 章

<b>コンフィギュレーション ロールバック変更確認</b>	<b>121</b>
機能情報の確認	121
コンフィギュレーション ロールバック変更確認について	122
コンフィギュレーション ロールバック変更確認の操作	122
コンフィギュレーション ロールバック変更確認の設定方法	122
コンフィギュレーションの置換またはコンフィギュレーションのロールバック操作の確認を伴う実行	122
コンフィギュレーション ロールバック変更確認の設定例	125
例：configure confirm コマンドを使用したコンフィギュレーション置換操作の実行	125



その他の参考資料	125
コンフィギュレーション ロールバック変更確認の機能情報	126

---

**第 11 章**
**コンフィギュレーション ロガー永続性 129**

機能情報の確認	129
コンフィギュレーション ロガー永続性の前提条件	129
コンフィギュレーション ロガー永続性について	130
コンフィギュレーション ロガー永続性を使用したコンフィギュレーション ファイルの保存	130
保持されたコマンド	131
コンフィギュレーション ロガー永続性機能を設定する方法	131
コンフィギュレーション ロガー永続性機能のイネーブル化	131
コンフィギュレーション ロガー永続性機能の検証とトラブルシューティング	133
コンフィギュレーション ロガー永続性機能の設定例	135
Cisco 7200 シリーズ ルータでのコンフィギュレーション ロガー永続性機能の設定例	135
その他の参考資料	135
コンフィギュレーション ロガー永続性の機能情報	136
用語集	137

---

**第 12 章**
**ソフトウェア メンテナンス アップグレード 139**

ソフトウェア メンテナンス アップグレードについて	139
ソフトウェア メンテナンス アップグレード	139
サポートされるプラットフォーム	140
ソフトウェア メンテナンス アップグレード パッケージ	140
ソフトウェア メンテナンス アップグレードのワークフロー	140
SMU のリロード	140
ソフトウェア メンテナンス アップグレードの制約事項	141
ソフトウェア メンテナンス アップグレードの設定方法	141
SMU の追加、アクティブ化、コミット	141
SMU のロールバック、非アクティブ化、または削除	142
ソフトウェア メンテナンス アップグレードの設定例	142

例：SMU の追加、アクティブ化、コミット 142

例：SMU のロールバック、非アクティブ化、または削除 145

ソフトウェア メンテナンス アップグレードのその他の参考資料 148

ソフトウェア メンテナンス アップグレードの機能情報 148



# 第 1 章

## 最初にお読みください

### Cisco IOS XE 16 に関する重要な情報

現行の Cisco IOS XE Release 3.7.0E (Catalyst スイッチング用) および Cisco IOS XE Release 3.17S (アクセスおよびエッジルーティング用) の2つのリリースは、単一バージョンのコンバインドリリース Cisco IOS XE 16 に進化 (マージ) しました。これにより、スイッチングおよびルーティングポートフォリオにおける広範なアクセス製品およびエッジ製品を1つのリリースでカバーします。

### 機能情報

機能のサポート、プラットフォームのサポート、およびシスコソフトウェアイメージのサポートに関する情報を検索するには、[Cisco Feature Navigator](#) を使用します。Cisco.com のアカウントは必要ありません。

### 参考資料

- [Cisco IOS コマンドリファレンス、全リリース](#)

### マニュアルの入手方法およびテクニカル サポート

- シスコからタイムリーな関連情報を受け取るには、[Cisco Profile Manager](#) でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、[シスコサービス](#) にアクセスしてください。
- サービス リクエストを送信するには、[シスコ サポート](#) にアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、[Cisco Marketplace](#) にアクセスしてください。
- 一般的なネットワーキング、トレーニング、認定関連の出版物を入手するには、[Cisco Press](#) にアクセスしてください。
- 特定の製品または製品ファミリの保証情報を探すには、[Cisco Warranty Finder](#) にアクセスしてください。





## 第 2 章

# コンフィギュレーション ファイルの管理

コンフィギュレーション ファイルを作成、ロード、維持することで、ユーザ設定のコマンド セットを生成し、現在のシスコ製ルーティングデバイスの機能性をカスタマイズできます。コンフィギュレーション ファイル管理コマンドの詳細については、『*Cisco IOS Configuration Fundamentals Command Reference*』を参照してください。

- [機能情報の確認 \(3 ページ\)](#)
- [コンフィギュレーション ファイルの管理の前提条件 \(3 ページ\)](#)
- [コンフィギュレーション ファイルの管理の制約事項 \(4 ページ\)](#)
- [コンフィギュレーション ファイルの管理について \(4 ページ\)](#)
- [コンフィギュレーション ファイル情報の管理方法 \(10 ページ\)](#)
- [その他の参考資料 \(31 ページ\)](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーション ファイルの管理の前提条件

- ユーザには、少なくとも Cisco IOS 環境とコマンドライン インターフェイスに関する基本的な知識が必要です。
- システムでは、少なくとも最小限の設定が実行されていることが必要です。`setup` コマンドを使用して基本的なコンフィギュレーション ファイルを作成できます (詳細について

は、「セットアップモードを使用したシスコ ネットワーキング デバイスの設定」を参照してください。

## コンフィギュレーションファイルの管理の制約事項

- このドキュメントで説明されている Cisco IOS コマンドの多くは、ルータの特定のコンフィギュレーションモードでのみ使用可能であり機能します。

## コンフィギュレーションファイルの管理について

### コンフィギュレーションファイルのタイプ

コンフィギュレーションファイルには、現在のシスコ製ルーティングデバイス（ルータ、アクセスサーバ、スイッチなど）の機能をカスタマイズするために使用される、Cisco IOS ソフトウェア コマンドが含まれています。コマンドは、システムを起動したとき（startup-config ファイルから）、またはコンフィギュレーションモードで CLI にコマンドを入力したときに、Cisco IOS ソフトウェアによって解析（変換および実行）されます。

スタートアップコンフィギュレーションファイル（startup-config）は、ソフトウェアを設定するためにシステムの起動時に使用されます。実行コンフィギュレーションファイル（running-config）には、ソフトウェアの現在の設定が含まれています。2つのコンフィギュレーションファイルは別々の設定にできます。たとえば、コンフィギュレーションを永続的ではなく短期間で変更する場合があります。その場合は、**configure terminal EXEC** コマンドを使用して実行コンフィギュレーションを変更しますが、そのコンフィギュレーションは **copy running-config startup-config EXEC** コマンドを使用して保存しません。

実行コンフィギュレーションを変更するには、「CLIでのコンフィギュレーションファイルの変更」の説明に従って、**configure terminal** コマンドを使用します。Cisco IOS コンフィギュレーションモードの使用時には、通常コマンドはすぐに実行され、入力直後またはコンフィギュレーションモードを終了した時点で実行コンフィギュレーションファイルに保存されます。

スタートアップコンフィギュレーションファイルを変更するには、**copy running-config startup-config EXEC** コマンドを使用してスタートアップコンフィギュレーションに実行コンフィギュレーションファイルを保存するか、ファイルサーバからスタートアップコンフィギュレーションにコンフィギュレーションファイルをコピーします（詳細については、「TFTP サーバからルータへのコンフィギュレーションファイルのコピー」を参照してください）。

## コンフィギュレーションモードおよびコンフィギュレーションソースの選択

ルータ上でコンフィギュレーションモードを開始するには、特権 EXEC プロンプトで **configure** コマンドを入力します。Cisco IOS ソフトウェアは次のプロンプトで応答し、端末、メモリ、

またはネットワークサーバ（ネットワーク）上に格納されたファイルのいずれかを、コンフィギュレーション コマンドのソースとして指定するように要求されます。

Configuring from terminal, memory, or network [terminal]?

端末からの設定では、コマンドラインにコンフィギュレーション コマンドを入力できます（次の項を参照してください）。メモリからの設定では、スタートアップコンフィギュレーション ファイルがロードされます。詳細については、「スタートアップコンフィギュレーション ファイルでのコンフィギュレーション コマンドの再実行」の項を参照してください。ネットワークからの設定では、ネットワーク経由でコンフィギュレーション コマンドをロードして実行できます。詳細については、「TFTP サーバからルータへのコンフィギュレーション ファイルのコピー」の項を参照してください。

## CLI を使用したコンフィギュレーション ファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れます。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブ コピーにも格納されないため、**show running-config or more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。**show startup-config** または **more nvram:startup-config EXEC** モード コマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがルータにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバ上に格納されているコンフィギュレーション ファイルのコメントのリストは表示できます。CLI を使用してソフトウェアは設定するときは、ユーザの入力に従ってソフトウェアによりコマンドが実行されます。

## コンフィギュレーション ファイルの場所

コンフィギュレーション ファイルは、次の場所に格納されます。

- 実行コンフィギュレーションは RAM に格納されます。
- クラス A フラッシュ ファイル システム プラットフォーム以外のすべてのプラットフォーム上では、スタートアップ コンフィギュレーションは不揮発性 RAM (NVRAM) に格納されます。
- クラス A フラッシュ ファイル システム プラットフォーム上では、スタートアップ コンフィギュレーションは CONFIG\_FILE 環境変数で指定された場所に格納されます（詳細については、の項を参照してください）。CONFIG\_FILE 変数は、デフォルトでは NVRAM になりますが、次のファイル システムのファイルも指定できます。
  - **nvram:** (NVRAM)

## ネットワーク サーバからルータへのコンフィギュレーションファイルのコピー

TFTP、rcp、またはFTP サーバからルータの実行コンフィギュレーションまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルをコピーできます。この機能は、次のいずれかの理由により実行する場合があります。

- バックアップコンフィギュレーションファイルを復元するため。
- 別のルータにコンフィギュレーションファイルを使用するため。たとえば、別のルータをネットワークに追加して、そのルータのコンフィギュレーションを元のルータと同様にする場合です。新しいルータにファイルをコピーすることにより、ファイル全体を再作成するのではなく、該当部分を変更できます。
- 同一のコンフィギュレーションコマンドをネットワーク内のすべてのルータにロードして、すべてのルータのコンフィギュレーションを同様にするため。

コマンドラインにコマンドを入力した場合と同様に、**copy {ftp:|rcp:|tftp:} system:running-config EXEC** コマンドはルータにコンフィギュレーションファイルをロードします。コマンドを追加する前に、ルータにより既存の実行コンフィギュレーションが消去されることはありません。コピーされたコンフィギュレーションファイル内のコマンドによって既存のコンフィギュレーションファイル内のコマンドが置き換えられた場合、既存のコマンドは消去されます。たとえば、コピーされたコンフィギュレーションファイルに含まれている特定のコマンドの IP アドレスが、既存のコンフィギュレーションと異なる場合は、コピーされたコンフィギュレーション内の IP アドレスが使用されます。ただし、既存のコンフィギュレーション内の一部のコマンドには、置き換えられたり無効になったりしないものもあります。このような場合、既存のコンフィギュレーションファイルとコピーされたコンフィギュレーションファイルで混成されたコンフィギュレーションファイルが作成され、コピーされたコンフィギュレーションファイルが優先されます。

コンフィギュレーションファイルをサーバ上に格納されているファイルの正確なコピーとして復元するには、そのコンフィギュレーションファイルをスタートアップコンフィギュレーションに直接コピーし (**copy ftp:|rcp:|tftp:} nvram:startup-config** コマンドを使用)、ルータをリロードする必要があります。

サーバからルータへコンフィギュレーションファイルをコピーするには、次の項で説明する作業を実行します。

使用するプロトコルは、使用中のサーバのタイプに応じて異なります。FTP および rcp のトランスポートメカニズムは、TFTP よりも高速でデータ配信の信頼性も優れています。これらの改善は、FTP および rcp のトランスポートメカニズムがコネクション型の TCP/IP スタック上に構築されており、これを使用しているために可能になりました。

## ルータから TFTP サーバへのコンフィギュレーションファイルのコピー

一部の TFTP 実装では、TFTP サーバ上にダミーファイルを作成し、読み取り、書き込み、および実行を許可してから、ダミーファイルを上書きする形でファイルをコピーする必要があります。詳細については、ご使用の TFTP のマニュアルを参照してください。



## ルータから FTP サーバへのコンフィギュレーションファイルのコピー

ルータから FTP サーバへコンフィギュレーションファイルのコピーできます。

### FTP ユーザ名およびパスワードの概要

FTP プロトコルでは、FTP 要求ごとにリモートユーザ名およびパスワードを、クライアントがサーバに送信する必要があります。FTP を使用してルータからサーバへコンフィギュレーションファイルのコピーする場合、Cisco IOS ソフトウェアは次の順番で最初に発見した有効なユーザ名を送信します。

1. **copy EXEC** コマンドで指定されたユーザ名（ユーザ名が指定されている場合）。
2. **ip ftp username** グローバル コンフィギュレーション コマンドで設定されたユーザ名（コマンドが設定されている場合）。
3. Anonymous

ルータは次の順番で最初に発見した有効なパスワードを送信します。

1. **copy** コマンドで指定されたパスワード（パスワードが指定されている場合）。
2. **ip ftp password** コマンドで設定されたパスワード（コマンドが設定されている場合）。
3. ルータは、*username @routename .domain* というパスワードを生成します。変数 *username* は現在のセッションに関連付けられたユーザ名、*routename* は設定済みのホスト名、*domain* はルータのドメインです。

ユーザ名およびパスワードは、FTP サーバのアカウントに関連付けられている必要があります。サーバに書き込む場合、ルータ上のユーザからの FTP 書き込み要求を受け入れるように、FTP サーバを適切に設定する必要があります。

このサーバがディレクトリ構造をとっている場合、コンフィギュレーションファイルまたはイメージは、サーバ上のユーザ名と関連付けられたディレクトリに書き込まれるか、そのディレクトリからコピーされます。たとえば、システム イメージがサーバ上のユーザのホーム ディレクトリにある場合は、そのユーザの名前をリモート ユーザ名として指定します。

詳細については、ご使用の FTP サーバのマニュアルを参照してください。

すべてのコピー操作に使用するユーザ名およびパスワードを指定するには、**ip ftp username** および **ip ftp password** グローバルコンフィギュレーションコマンドを使用します。特定のコピー操作にのみ使用するユーザ名を指定する場合は、**copy EXEC** コマンド内でユーザ名を指定します。

### VRF によるファイルのコピー

**copy** コマンドで指定した VRF インターフェイス経由でファイルをコピーできます。**copy** コマンドで VRF を指定するほうが、変更リクエストを使用して設定を変更しなくても送信元インターフェイスを直接変更でき、簡単で効率的です。

次の例に、**copy** コマンドを使用して VRF 経由でファイルをコピーする方法を示します。

```
Device# copy scp: slot0: vrf test-vrf
Device# copy scp: slot0: vrf test-vrf
Address or name of remote host [10.1.2.3]?
Source username [ScpUser]?
Source filename [/auto/tftp-server/ScpUser/vrf_test.txt]?
Destination filename [vrf_test.txt]?
Getting the vrf name as test-vrf
Password:
Sending file modes: C0644 10 vrf_test.txt
!
223 bytes copied in 22.740 secs (10 bytes/sec)
```

## NVRAM より大きいコンフィギュレーションファイル

NVRAM より大きいコンフィギュレーションファイルを維持管理するには、以降の項の情報を知っておく必要があります。

### コンフィギュレーションファイルの圧縮

**servicecompress-config** グローバルコンフィギュレーションコマンドは、コンフィギュレーションファイルを圧縮して NVRAM に格納することを指定します。コンフィギュレーションファイルが圧縮されると、ルータは正常に機能します。システムの起動時に、システムはコンフィギュレーションファイルが圧縮されていることを認識し、圧縮されたコンフィギュレーションファイルを展開して、正常に処理を進めます。**morenvram:startup-config EXEC** コマンドにより、コンフィギュレーションが展開されてから表示されます。

コンフィギュレーションファイルを圧縮する前に、適切なハードウェアのインストールおよびメンテナンスマニュアルを参照してください。ご利用のシステムの ROM がファイル圧縮をサポートしていることを確認します。サポートしていない場合、ファイル圧縮をサポートしている新しい ROM をインストールできます。

コンフィギュレーションのサイズは、NVRAM のサイズの 3 倍を超えてはいけません。NVRAM のサイズが 128 KB の場合、展開できる最大のコンフィギュレーションファイルのサイズは 384 KB です。

**servicecompress-config** グローバルコンフィギュレーションコマンドは、Cisco IOS ソフトウェア Release 10 以降のブート ROM を使用している場合に限り実行できます。新しい ROM のインストールは 1 回限りの操作で、ROM に Cisco IOS Release 10 がない場合にのみ必要です。ブート ROM が圧縮コンフィギュレーションを認識しない場合は、次のメッセージが表示されます。

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

### ネットワークからのコンフィギュレーションコマンドのロード

コンフィギュレーションが大きい場合は、FTP、TFTP のいずれかのサーバに格納しておき、システムの起動時にダウンロードすることもできます。ネットワークサーバを大きいコンフィギュレーションの保存に使用するためのコマンドの詳細については、「ルータから TFTP サーバへのコンフィギュレーションファイルのコピー」および「コンフィギュレーションファイルをダウンロードするためのルータの設定」の各項を参照してください。

## パーサー キャッシュの制御

Cisco IOS ソフトウェアの Cisco IOS コマンドライン パーサーは、コマンドラインを変換および実行（解析）します。パーサー キャッシュ機能は、大きいコンフィギュレーション ファイルを迅速に処理するために開発されました。これにより、ロード時間が大幅に改善されます。

パーサー キャッシュ機能では、簡略化された解析グラフをダイナミックに作成、キャッシュ、および再使用することにより、コンフィギュレーションファイル内の、前回使用された設定行と微妙に異なる設定行（たとえば `pvc 0/100`、`pvc 0/101` など）が、迅速に認識および変換できるようになります。この改善は、主に同じようなコマンドを何百回、何千回と繰り返すコンフィギュレーションファイルに役立ちます。このようなコンフィギュレーションファイルには、サブインターフェイス用に何千もの仮想回線を設定する必要がある場合や、何百ものアクセスリストを設定する必要がある場合があります。数値の引数だけが異なる同一のコマンドが繰り返し使用されているファイルのほとんどで、性能が向上します。

パーサー キャッシュは、Cisco IOS Release 12.1(5)T 以降のリリースを使用するすべてのプラットフォームで、デフォルトでイネーブルにされています。ただし、大きいコンフィギュレーションファイルが必要としないシスコ デバイスを使用しているユーザの場合は、パーサー キャッシュをディセーブルにし、この機能で使用されるリソースを解放できます（この機能により使用されるメモリは、解析されるコンフィギュレーションファイルのサイズに依存しますが、通常は 512 KB 未満です）。

パーサー キャッシュを制御するには、いくつかの方法があります（これらはすべて任意です）。

- パーサー キャッシュのクリア：リソースを解放するか、またはパーサー キャッシュのメモリをリセットするために、パーサー キャッシュ機能に格納されている解析エントリおよびヒット数とミス数の統計情報をクリアすることもできます。
- パーサー キャッシュのディセーブル化：パーサー キャッシュ機能は、デフォルトでイネーブルされています。パーサー キャッシュ機能をディセーブルにするには、グローバル コンフィギュレーション モードで `no parser cache` コマンドを使用します。パーサー キャッシュがディセーブルになると、`noparsercache` コマンドラインが実行コンフィギュレーションファイルに書き込まれます。システム リソースを解放するためにパーサー キャッシュをディセーブルにする場合は、`noparsercache` コマンドを発行する前にパーサー キャッシュをクリアする必要があります。パーサー キャッシュをディセーブルにした後は、パーサー キャッシュをクリアできません。
- パーサー キャッシュの再イネーブル化：パーサー キャッシュ機能をディセーブルにした後、再度イネーブルにするには、グローバルコンフィギュレーションモードで `parsercache` コマンドを使用します。
- パーサーのモニタリング：最後に解析されたコンフィギュレーションファイルに関する統計情報は、パーサー キャッシュ機能により解析されたコマンドのヒット数とミス数の統計情報とともにシステム メモリに格納されます。「hits（ヒット数）」および「misses（ミス数）」は、前回使用された類似するコマンドに対し、コンフィギュレーションセッション中にパーサー キャッシュが検出した一致数を示しています。一致したコマンド（「hits」）は、より効率的に解析されます。一致しなかったコマンド（「misses」）の解析時間は、パーサー キャッシュにより改善されることはありません。

## コンフィギュレーションファイルをダウンロードするルータの設定

システムの起動時に1つまたは2つのコンフィギュレーションファイルをロードするようにルータを設定できます。コンフィギュレーションファイルは、コマンドラインにコマンドを入力した場合と同様に、メモリにロードされ読み込まれます。したがって、ルータのコンフィギュレーションは、元のスタートアップコンフィギュレーションと1つまたは2つのダウンロードされたコンフィギュレーションファイルで混成されたものになります。

### ネットワークとホストのコンフィギュレーションファイル

歴史的な理由から、ルータが最初にダウンロードするファイルは、ネットワークコンフィギュレーションファイルと呼ばれます。ルータが2番目にダウンロードするファイルは、ホストコンフィギュレーションファイルと呼ばれます。2つのコンフィギュレーションファイルは、ネットワーク上のすべてのルータが、同一コマンドの多くを使用する場合に使用できます。ネットワークコンフィギュレーションファイルには、すべてのルータを設定するために使用される標準コマンドが含まれます。ホストコンフィギュレーションファイルには、特定の1つのホストに固有のコマンドが含まれます。2つのコンフィギュレーションファイルをロードする場合、ホストコンフィギュレーションファイルを、もう1つのファイルより優先させる必要があります。ネットワークコンフィギュレーションファイルおよびホストコンフィギュレーションファイルは、両方ともTFTP、rtp、FTPのいずれかを介して到達可能なネットワークサーバ上にあり、読み取り可能である必要があります。

## コンフィギュレーションファイル情報の管理方法

### コンフィギュレーションファイル情報の表示

コンフィギュレーションファイルに関する情報を表示するには、このセクションの手順を実行します。

#### 手順の概要

1. **enable**
2. **show boot**
3. **more file-url**
4. **show running-config**
5. **show startup-config**

#### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	<b>show boot</b> 例： Device# show boot	BOOT 環境変数の内容、CONFIG_FILE 環境変数によって指定されているコンフィギュレーションファイルの名前、およびBOOTLDR 環境変数の内容を示します。
ステップ 3	<b>more file-url</b> 例： Device# more 10.1.1.1	指定されたファイルの内容を表示します。
ステップ 4	<b>show running-config</b> 例： Device# show running-config	実行コンフィギュレーションファイルの内容を表示します（ <b>more system:running-config</b> コマンドのコマンドエイリアスです）。
ステップ 5	<b>show startup-config</b> 例： Device# show startup-config	スタートアップ コンフィギュレーション ファイルの内容を表示します。（ <b>more nvram:startup-config</b> コマンドのコマンドエイリアスです）。  クラス A フラッシュ ファイル システム プラットフォーム以外のすべてのプラットフォーム上では、通常、デフォルトの startup-config ファイルは NVRAM に格納されます。クラス A フラッシュ ファイル システム プラットフォーム上では、CONFIG_FILE 環境変数はデフォルトの startup-config ファイルを指定します。CONFIG_FILE 変数のデフォルトは NVRAM になります。

## CLI でのコンフィギュレーションファイルの変更

Cisco IOS ソフトウェアは、1 行につき 1 つのコンフィギュレーション コマンドを受け入れません。コンフィギュレーション コマンドは、必要なだけ入力できます。コンフィギュレーション ファイルには、入力したコマンドを説明するコメントを追加できます。コメントの先頭には、感嘆符 (!) を付けます。コメントは NVRAM にもコンフィギュレーション ファイルのアクティブ コピーにも格納されないため、**show running-config or more system:running-config EXEC** コマンドでアクティブな設定のリストを表示しても、コメントは表示されません。

**show startup-config** または **more nvram:startup-config EXEC** モード コマンドでスタートアップ コンフィギュレーションのリストを表示しても、コメントは表示されません。コメントは、コンフィギュレーション ファイルがルータにロードされたときにコンフィギュレーション ファイルから削除されます。ただし、ファイル転送プロトコル (FTP)、リモートコピープロトコル (RCP)、または Trivial File Transfer Protocol (TFTP) サーバ上に格納されているコンフィギュレーション ファイルのコメントのリストは表示できます。CLI を使用してソフトウェアは設定するときは、ユーザの入力に従ってソフトウェアによりコマンドが実行されます。CLI を

使用してソフトウェアを設定するには、特権EXECモードを開始して次のコマンドを使用します。

手順の概要

1. **enable**
2. **configure terminal**
3. 次のいずれかを実行します。
  - **end**
  - **^Z**
4. **copy system:running-config nvram:startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します（要求された場合）。</li> </ul>
ステップ 2	<p><b>configure terminal</b></p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>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。必要なコンフィギュレーションコマンドを入力します。Cisco IOS マニュアルセットに、テクノロジー別に編成されたコンフィギュレーションコマンドが説明されています。</p>
ステップ 3	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• <b>^Z</b></li> </ul> <p>例 :</p> <pre>Device(config)# end</pre>	<p>コンフィギュレーションセッションを終了し、EXEC モードに戻ります。</p> <p>(注) Ctrl キーと Z キーを同時に押すと、画面に ^Z と表示されます。</p>
ステップ 4	<p><b>copy system:running-config nvram:startup-config</b></p> <p>例 :</p> <pre>Device# copy system:running-config nvram:startup-config</pre>	<p>実行コンフィギュレーション ファイルをスタートアップ コンフィギュレーション ファイルとして保存します。</p> <p><b>copy running-config startup-config</b> コマンドエイリアスも使用できますが、このコマンドは精度が高くないため、注意する必要があります。ほとんどのプラットフォーム上では、このコマンドによりコンフィギュレーションはNVRAMに保存されます。クラス A フラッシュ ファイル システムのプラットフォーム上では、この手順によりコンフィギュレーションは CONFIG_FILE 環境変数によって指定され</p>

	コマンドまたはアクション	目的
		た場所に保存されます (デフォルトの CONFIG_FILE 変数では、ファイルの保存先は NVRAM に指定されています)。

**例**

次の例では、デバイスのデバイスプロンプト名を設定しています。感嘆符 (!) で示されたコメント行では、いずれのコマンドも実行されません。hostname コマンドを使用して、デバイス名を Device から new\_name に変更しています。Ctrl-Z (^Z) キーを押すか、end コマンドを入力すると、コンフィギュレーションモードが終了します。copy system:running-config nvram:startup-config コマンドにより、現在のコンフィギュレーションがスタートアップ コンフィギュレーションに保存されます。

```
Device# configure terminal
Device(config) # !The following command provides the Device host name.
Device(config) # hostname new_name
new_name(config) # end
new_name# copy system:running-config nvram:startup-config
```

スタートアップ コンフィギュレーションが NVRAM にある場合は、スタートアップ コンフィギュレーションには現在の設定情報がコンフィギュレーションコマンドとしてテキスト形式で格納され、デフォルト以外の設定だけが記録されます。破損データから保護するために、メモリはチェックサム算出されます。



(注) 一部の特定のコマンドは、NVRAM に保存されない場合があります。これらのコマンドは、マシンをリブートしたときに再入力する必要があります。これらのコマンドは、マニュアルに記載されています。リブート後にすばやくデバイスを再設定できるように、これらの設定のリストを保管しておくことを推奨します。

## ルータからTFTPサーバへのコンフィギュレーションファイルのコピー

TFTP ネットワーク サーバ上の設定をコピーするには、以下の手順を実行します。

**手順の概要**

1. enable
2. copy system:running-config tftp: [[[//location ]/directory ]/filename ]
3. copy nvram:startup-config tftp: [[[//location ]/directory ]/filename ]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>copy system:running-config tftp: [[[//location ]/directory ]/filename ]</b> 例： Device# copy system:running-config tftp: //server1/topdir/file10	TFTP サーバへ実行コンフィギュレーション ファイルをコピーします。
ステップ 3	<b>copy nvram:startup-config tftp: [[[//location ]/directory ]/filename ]</b> 例： Device# copy nvram:startup-config tftp: //server1/lstdir/file10	TFTPサーバへスタートアップコンフィギュレーション ファイルをコピーします。

例

次に、デバイスからから TFTP サーバへコンフィギュレーション ファイルをコピーする例を示します。

```
Tokyo# copy system:running-config tftp://172.16.2.155/tokyo-config
Write file tokyo-config on host 172.16.2.155? [confirm] y
Writing tokyo-config!!! [OK]
```

次の作業

**copy** コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **fileprompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

## ルータから FTP サーバへのコンフィギュレーションファイルのコピー

ルータから FTP サーバへスタートアップコンフィギュレーションファイルまたは実行コンフィギュレーション ファイルをコピーするには、以下のタスクを実行します。

手順の概要

1. **enable**
2. **configure terminal**



3. **ip ftp username** *username*
4. **ip ftp password** *password*
5. **end**
6. 次のいずれかを実行します。
  - **copy system:running-config ftp:** [[[/[*username* [:*password* ]@]*location/directory* ]/*filename* ]
  - 
  - 
  - **copy nvram:startup-config ftp:** [[[/[*username* [:*password* ]@]*location/directory* ]/*filename* ]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>ip ftp username</b> <i>username</i> 例： Device(config)# ip ftp username user1	(任意) デフォルトのリモートユーザ名を指定します。
ステップ 4	<b>ip ftp password</b> <i>password</i> 例： Device(config)# ip ftp password guessme	(任意) デフォルトのパスワードを指定します。
ステップ 5	<b>end</b> 例： Device(config)# end	(任意) グローバル コンフィギュレーション モードを終了します。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 2 および 3 を参照）。
ステップ 6	次のいずれかを実行します。 • <b>copy system:running-config ftp:</b> [[[/[ <i>username</i> [: <i>password</i> ]@] <i>location/directory</i> ]/ <i>filename</i> ] • • • <b>copy nvram:startup-config ftp:</b> [[[/[ <i>username</i> [: <i>password</i> ]@] <i>location/directory</i> ]/ <i>filename</i> ]	FTP サーバへ実行コンフィギュレーションまたはスタートアップ コンフィギュレーション ファイルをコピーします。

	コマンドまたはアクション	目的
	例 :  Device# copy system:running-config ftp://user1:guessme@company.com /dir10/file1	

## 例

### FTP サーバへの実行コンフィギュレーションファイルの格納

次に、rtr2-config という名前の実行コンフィギュレーションファイルを、IP アドレスが 172.16.101.101 のリモート ホスト上の netadmin1 ディレクトリにコピーする例を示します。

```
Device# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/Rtr2-config
Write file rtr2-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

### FTP サーバへのスタートアップコンフィギュレーションファイルの格納

次に、FTP を使用してファイルをコピーすることによって、サーバ上にスタートアップコンフィギュレーションファイルを格納する例を示します。

```
Rtr2# configure terminal

Rtr2(config)# ip ftp username netadmin2

Rtr2(config)# ip ftp password mypass

Rtr2(config)# end

Rtr2# copy nvram:startup-config ftp:

Remote host[]? 172.16.101.101

Name of configuration file to write [rtr2-config]?
Write file rtr2-config on host 172.16.101.101?[confirm]
![OK]
```

## 次の作業

**copy EXEC** コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **fileprompt** グローバルコンフィギュレーションコマンドの現在の設定によって異なります。

## TFTPサーバからルータへのコンフィギュレーションファイルのコピー

TFTP サーバからデバイスへコンフィギュレーションファイルをコピーするには、以下のタスクを実行します。

## 手順の概要

1. **enable**
2. **copy tftp: [[[//location ]/directory ]/filename ] system:running-config**
3. **copy tftp: [[[//location ]/directory ]/filename ] nvram:startup-config**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例：  Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>copy tftp: [[[//location ]/directory ]/filename ] system:running-config</b> 例：  Device# copy tftp://server1/dir10/datasource system:running-config	TFTP サーバから実行コンフィギュレーションへコンフィギュレーション ファイルをコピーします。
ステップ 3	<b>copy tftp: [[[//location ]/directory ]/filename ] nvram:startup-config</b> 例：  Device# copy tftp://server1/dir10/datasource nvram:startup-config	TFTP サーバからスタートアップ コンフィギュレーションへコンフィギュレーション ファイルをコピーします。

### 例

次に、IP アドレス 172.16.2.155 にある、tokyo-config という名前のファイルからソフトウェアを設定する例を示します。

```
Device1# copy tftp://172.16.2.155/tokyo-config system:running-config
Configure using tokyo-config from 172.16.2.155? [confirm] y
Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

## 次の作業

**copy EXEC** コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **fileprompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

# FTP サーバからルータへのコンフィギュレーションファイルのコピー

FTPサーバから実行コンフィギュレーションまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルのコピーするには、以下のタスクを実行します。

## 手順の概要

1. **enable**
2. **configure terminal**
3. **ip ftp username *username***
4. **ip ftp password *password***
5. **end**
6. **copy ftp: [[[//[*username* [:*password* ]@]*location* ]/*directory* ]/*filename* ]system:running-config**

## 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	（任意）グローバル コンフィギュレーション モードを開始できます。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 2 および 3 を参照）。
ステップ 3	<b>ip ftp username <i>username</i></b> 例： Device(config)# ip ftp username user1	（任意）デフォルトのリモートユーザ名を指定します。
ステップ 4	<b>ip ftp password <i>password</i></b> 例： Device(config)# ip ftp password guessme	（任意）デフォルトのパスワードを指定します。
ステップ 5	<b>end</b> 例： Device(config)# end	（任意）グローバル コンフィギュレーション モードを終了します。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 2 および 3 を参照）。
ステップ 6	<b>copy ftp: [[[//[<i>username</i> [:<i>password</i> ]@]<i>location</i> ]/<i>directory</i> ]/<i>filename</i> ]system:running-config</b> 例： 例：	FTPを使用して、ネットワークサーバから実行メモリまたはスタートアップコンフィギュレーションへコンフィギュレーションファイルのコピーします。

	コマンドまたはアクション	目的
	<pre> or 例： <b>copy ftp:</b>[[[//[username [:password ]@]location/directory ]/filename ] <b>nvr</b>am:startup-config 例： Device# copy ftp://user1:guessme@company.com /dir10/datasource nvr:startup-config                     </pre>	

## 例

### FTP の Running-Config のコピー

次に、host1-config という名前のホスト コンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモート サーバ上の netadmin1 ディレクトリからコピーし、デバイスでコマンドをロードして実行する例を示します。

```

Device# copy rcp://netadmin1:mypass@172.16.101.101/host1-config system:running-config

Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:[OK]
Device#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
    
```

### FTP の Startup-Config のコピー

次に、リモートユーザ名 netadmin1 を指定する例を示します。次に host2-config という名前のコンフィギュレーション ファイルを、IP アドレスが 172.16.101.101 のリモート サーバ上の netadmin1 ディレクトリからスタートアップ コンフィギュレーションへコピーします。

```

Rtr2# configure terminal
Rtr2(config)# ip ftp username
netadmin1
Rtr2(config)# ip ftp password
mypass
Rtr2(config)# end
Rtr2# copy ftp: nvr:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:[OK]
[OK]
Rtr2#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
    
```

## 次の作業

**copy EXEC** コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **fileprompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

## NVRAM より大きいコンフィギュレーションファイルの保守

NVRAM のサイズを超えるコンフィギュレーションファイルを保守するには、以降の項で説明する作業を実行します。

### コンフィギュレーションファイルの圧縮

コンフィギュレーションファイルを圧縮するには、このセクションの手順を実行してください。

#### 手順の概要

1. **enable**
2. **configure terminal**
3. **service compress-config**
4. **end**
5. 次のいずれかを実行します。
  - 新しいコンフィギュレーションをコピーするには、FTP、rcp、または TFTP を使用します。
  - **configure terminal**
6. **copy system:running-config nvram:startup-config**

#### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例：  Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例：  Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>service compress-config</b> 例：  Device(config)# service compress-config	コンフィギュレーションファイルを圧縮することを指定します。

	コマンドまたはアクション	目的
ステップ 4	<b>end</b> 例： Device(config)# end	グローバル コンフィギュレーション モードを終了します。
ステップ 5	次のいずれかを実行します。 <ul style="list-style-type: none"> <li>• 新しいコンフィギュレーションをコピーするには、FTP、rcp、または TFTP を使用します。</li> <li>• <b>configure terminal</b></li> </ul> 例： Device# configure terminal	新しいコンフィギュレーションを入力します。 <ul style="list-style-type: none"> <li>• NVRAM のサイズの 3 倍以上のコンフィギュレーションをロードしようとする、次のエラー メッセージが表示されます。</li> </ul> 「[buffer overflow -file-size /buffer-size bytes]。」
ステップ 6	<b>copy system:running-config nvram:startup-config</b> 例： Device(config)# copy system:running-config nvram:startup-config	実行コンフィギュレーションの変更が終わったら、新しいコンフィギュレーションを保存します。

**例**

次に、129KB のコンフィギュレーションファイルを 11KB に圧縮する例を示します。

```

Device# configure terminal

Device(config)# service compress-config

Device(config)# end

Device# copy tftp://172.16.2.15/tokyo-config system:running-config

Configure using tokyo-config from 172.16.2.155? [confirm] y

Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
Device# copy system:running-config nvram:startup-config

Building configuration...
Compressing configuration from 129648 bytes to 11077 bytes
[OK]
    
```

## パーサー キャッシュの管理

パーサーキャッシュ機能を制御するには、次の項で説明する作業を実行します。これらの作業はすべて任意です。

## パーサー キャッシュのクリア

パーサー キャッシュ機能によって格納された情報をクリアするには、このセクションのタスクを実行します。

### 手順の概要

1. **enable**
2. **clear parser cache**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>clear parser cache</b> 例： Device# clear parser cache	パーサー キャッシュ機能に格納されている解析キャッシュエントリおよびヒット数とミス数の統計情報をクリアします。

## パーサー キャッシュのディセーブル化

パーサー キャッシュ機能は、デフォルトでイネーブルにされています。パーサー キャッシュ機能を無効にするには、このセクションのタスクを実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **no parser cache**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。



	コマンドまたはアクション	目的
ステップ 3	<p><b>no parser cache</b></p> <p>例 :</p> <pre>Device(config)# no parser cache</pre>	<p>パーサー キャッシュ機能をディセーブルにします。</p> <ul style="list-style-type: none"> <li>パーサー キャッシュがディセーブルになると、<b>noparsercache</b> コマンドラインが実行コンフィギュレーション ファイルに書き込まれます。</li> <li>システム リソースを解放するためにパーサー キャッシュをディセーブルにする場合は、<b>noparsercache</b> コマンドを発行する前にパーサー キャッシュをクリアする必要があります。パーサー キャッシュをディセーブルにした後は、パーサー キャッシュをクリアできません。</li> </ul>

## パーサー キャッシュの再イネーブル化

パーサー キャッシュ機能を無効にした後に再度有効にするには、このセクションのタスクを実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **parser cache**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>パスワードを入力します (要求された場合)。</li> </ul>
ステップ 2	<p><b>configure terminal</b></p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>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。</p>
ステップ 3	<p><b>parser cache</b></p> <p>例 :</p> <pre>Device(config)# parser cache</pre>	<p>パーサー キャッシュ機能をイネーブルにします。</p>

## 次の作業

**showparserstatistics** コマンドにより、次の 2 セットのデータが表示されます。

- コンフィギュレーションファイル内のコマンドのうち、最後に実行コンフィギュレーションにコピーされたコマンドの数、およびシステムがこれらのコマンドを解析するために要した時間（コンフィギュレーションファイルはシステムの起動時または **copysourcerunning-config EXEC** コマンドなどのコマンドを発行することによって実行コンフィギュレーションにロードされます）。
- パーサーキャッシュのステータス（イネーブルまたはディセーブル）、およびシステムの起動以降またはパーサーキャッシュのクリア以降に一致したコマンドの数（ヒット数またはミス数）。

## フラッシュメモリからスタートアップまたは実行コンフィギュレーションへのコンフィギュレーションファイルのコピー

フラッシュメモリから現在の NVRAM にあるスタートアップコンフィギュレーションまたは実行コンフィギュレーションへコンフィギュレーションファイルを直接コピーするには、ステップ 2 のいずれかのコマンドを入力します。

### 手順の概要

1. **enable**
2. 次のいずれかを実行します。
  - **copy filesystem** : [partition-number:][filename] **nvrām:startup-config**
  - **copy filesystem** : [partition-number:][filename] **system:running-config**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します（要求された場合）。</li> </ul>
ステップ 2	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <li>• <b>copy filesystem</b> : [partition-number:][filename] <b>nvrām:startup-config</b></li> <li>• <b>copy filesystem</b> : [partition-number:][filename] <b>system:running-config</b></li> </ul> <p>例 :</p> <pre>Device# copy slot0:4:ios-upgrade-1 nvrām:startup-config</pre>	<p>NVRAM にコンフィギュレーションファイルを直接ロードします。</p> <p>または</p> <p>現在の実行コンフィギュレーションにコンフィギュレーションファイルをコピーします。</p>

例

次に、スロット 0 にあるフラッシュメモリ PC カードのパーティション 4 からルータのスタートアップコンフィギュレーションへ `ios-upgrade-1` という名前のファイルをコピーする例を示します。

```
Device# copy slot0:4:ios-upgrade-1 nvram:startup-config

Copy '
ios-upgrade-1
' from flash device
  as 'startup-config' ? [yes/no] yes

[OK]
```

## FTP サーバからフラッシュメモリ デバイスへのコンフィギュレーションファイルのコピー

FTP サーバからフラッシュメモリ デバイスへコンフィギュレーションファイルをコピーするには、以下のタスクを実行します。

手順の概要

1. `enable`
2. `configure terminal`
3. `ip ftp username username`
4. `ip ftp password password`
5. `end`
6. `copy ftp: [[[//[username:password@]location ]/directory ]/filename ] flash-filesystem:[partition-number:][filename ]`

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<code>enable</code> 例：  Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<code>configure terminal</code> 例：  Device# configure terminal	(任意) 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。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 3 および 4 を参照）。
ステップ 3	<code>ip ftp username username</code> 例：	(任意) リモートユーザ名を指定します。

	コマンドまたはアクション	目的
	Device(config)# ip ftp username user1	
ステップ 4	<b>ip ftp password <i>password</i></b> 例 : Device(config)# ip ftp password guessme	(任意) リモート パスワードを指定します。
ステップ 5	<b>end</b> 例 : Device(config)# end	(任意) コンフィギュレーションモードを終了します。このステップが必要になるのは、デフォルトのリモート ユーザ名を上書きする場合のみです (ステップ 3 および 4 を参照)。
ステップ 6	<b>copy ftp: [[[/[username:password@]location ]/directory ]/filename ]flash-filesystem:[partition-number:][filename ]</b> 例 : Device> copy ftp:router-config slot0:new-config	FTP を使用してネットワーク サーバからフラッシュ メモリ デバイスへコンフィギュレーション ファイルをコピーします。

## 次の作業

**copy EXEC** コマンドを発行した後、追加情報またはアクションの確認を求めるプロンプトが表示される場合があります。表示されるプロンプトは、**copy** コマンドで入力した情報量および **fileprompt** グローバル コンフィギュレーション コマンドの現在の設定によって異なります。

## rcp サーバからフラッシュ メモリ デバイスへのコンフィギュレーション ファイルのコピー

rcp サーバからフラッシュ メモリ デバイスへコンフィギュレーション ファイルをコピーするには、以下のタスクを実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username *username***
4. **end**
5. **copy rcp: [[[/[username@]location ]/directory ]/filename ] flash-filesystem:[partition-number:][filename ]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	（任意）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。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 3 および 4 を参照）。
ステップ 3	<b>ip rcmd remote-username <i>username</i></b> 例： Device(config)# ip rcmd remote-username user1	（任意）リモート ユーザ名を指定します。
ステップ 4	<b>end</b> 例： Device(config)# end	（任意）コンフィギュレーションモードを終了します。この手順は、デフォルトのリモートユーザ名またはパスワードを上書きする場合にだけ必要です（ステップ 3 および 4 を参照）。
ステップ 5	<b>copy rcp: [[[/<i>username@</i>]<i>location</i> ]/<i>directory</i> ]/<i>filename</i> ] flash-filesystem:[<i>partition-number</i>]:[<i>filename</i> ]</b> 例： Device# copy rcp:router-config slot0:new-config	rcp を使用してネットワーク サーバからフラッシュメモリ デバイスへコンフィギュレーション ファイルをコピーします。追加情報または確認を要求するルータからのプロンプトに対し応答します。このプロンプトは、 <b>copy</b> コマンドで入力した情報量および <b>fileprompt</b> コマンドの現在の設定によって異なります。

## TFTPサーバからフラッシュメモリデバイスへのコンフィギュレーションファイルのコピー

TFTP サーバからフラッシュ メモリ デバイスへコンフィギュレーション ファイルをコピーするには、以下のタスクを実行します。

手順の概要

1. **enable**
2. **copy tftp: [[[/*location* ]/*directory* ]/*filename* ] flash-filesystem:[*partition-number*]:[*filename* ]**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>
ステップ 2	<p><b>copy tftp: [[[//location ]/directory ]/filename ] flash-filesystem:[partition-number:][filename ]</b></p> <p>例 :</p> <pre>Device# copy tftp:router-config slot0:new-config</pre>	<p>TFTP サーバからフラッシュ メモリ デバイスへファイルをコピーします。追加情報または確認を要求するデバイスからのプロンプトに対し応答します。このプロンプトは、<b>copy</b> コマンドで入力した情報量および <b>fileprompt</b> コマンドの現在の設定によって異なります。</p>

例

次の例は、TFTP サーバから Cisco 7500 シリーズ デバイスのネットワーク処理エンジン (NPE) またはルート スイッチ プロセッサ (RSP) カードのスロット 0 に挿入されたフラッシュメモリカードへ Device-config という名前のコンフィギュレーション ファイルをコピーする例を示します。コピーされたファイルの名前は new-config に変更されます。

```
Device# copy tftp:router-config slot0:new-config
```

## スタートアップコンフィギュレーション ファイルでのコンフィギュレーション コマンドの再実行

スタートアップコンフィギュレーションファイルのコマンドを再実行するには、このセクションのタスクを実行します。

手順の概要

1. **enable**
2. **configure memory**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>

	コマンドまたはアクション	目的
ステップ 2	<b>configure memory</b> 例 : Device# configure memory	スタートアップコンフィギュレーションファイルにあるコンフィギュレーションコマンドを再実行します。

## スタートアップコンフィギュレーションのクリア

スタートアップコンフィギュレーションから設定情報を消去できます。スタートアップコンフィギュレーションなしでルータをリブートした場合は、ルータを最初から設定できるように、ルータは **Setup** コマンドファシリティに移行します。スタートアップコンフィギュレーションの内容を消去するには、このセクションのタスクを実行します。

### 手順の概要

1. **enable**
2. **erase nvram**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例 : Device> enable	特権 EXEC モードをイネーブルにします。 <ul style="list-style-type: none"> <li>• パスワードを入力します（要求された場合）。</li> </ul>
ステップ 2	<b>erase nvram</b> 例 :	スタートアップコンフィギュレーションの内容をクリアします。

	コマンドまたはアクション	目的
	<pre>Device# erase nvram</pre>	<p>(注) クラス A フラッシュ ファイル システムのプラットフォーム以外のすべてのプラットフォームでは、このコマンドにより NVRAM が消去されます。スタートアップ コンフィギュレーション ファイルは、いったん削除すると復元できません。クラス A フラッシュ ファイル システムのプラットフォーム上では、<b>erasestartup-configEXEC</b> コマンドを使用すると、CONFIG_FILE 環境変数により指定されたコンフィギュレーションが、デバイスにより削除されます。この変数が NVRAM を指定している場合は、デバイスにより NVRAM が消去されます。CONFIG_FILE 環境変数がフラッシュ メモリ デバイスとコンフィギュレーション ファイル名を指定している場合は、デバイスによりコンフィギュレーション ファイルが削除されます。つまり、そのコンフィギュレーション ファイルはデバイスにより消去されるのではなく、「削除済み」としてマークされます。この機能では、削除されたファイルを回復できます。</p>

## 指定されたコンフィギュレーション ファイルの削除

特定のフラッシュ デバイスの指定された設定を削除するには、このセクションのタスクを実行します。

### 手順の概要

1. **enable**
2. **delete flash-filesystem : filename**

### 手順の詳細

	コマンドまたはアクション	目的
<p>ステップ 1</p>	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>
<p>ステップ 2</p>	<p><b>delete flash-filesystem : filename</b></p> <p>例 :</p>	<p>指定されたフラッシュ デバイス上の指定されたコンフィギュレーション ファイルを削除します。</p>



	コマンドまたはアクション	目的
	Device# delete slot0:myconfig	<p>(注) クラス A および B フラッシュ ファイル システムでは、フラッシュ メモリ内の特定のファイルを削除すると、そのファイルは削除済みとしてシステムによりマークされます。これにより、<b>undelete EXEC</b> コマンドを使用して、削除したファイルを後で回復できるようになります。消去されたファイルは回復できません。コンフィギュレーション ファイルを完全に消去するには、<b>squeeze EXEC</b> コマンドを使用します。クラス C フラッシュ ファイル システムでは、削除されたファイルは回復できません。CONFIG_FILE 環境変数で指定されたコンフィギュレーション ファイルを消去または削除しようとした場合、システムにより削除の確認を求めるプロンプトが表示されます。</p>

## その他の参考資料

### 関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
Cisco IOS コンフィギュレーション コマンド	Cisco IOS Configuration Fundamentals Command Reference

### シスコのテクニカル サポート

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





## 第 3 章

# コンフィギュレーション生成のパフォーマンス拡張

コンフィギュレーション生成のパフォーマンス拡張機能は、実行中のコンフィギュレーションファイル情報の収集を高速化することでコンフィギュレーション管理を支援します。この機能は、多数のインターフェイスが構成された大規模なネットワークを管理する場合に特に便利です。

- [機能情報の確認 \(33 ページ\)](#)
- [コンフィギュレーション生成のパフォーマンス拡張に関する制限事項 \(34 ページ\)](#)
- [コンフィギュレーション生成のパフォーマンス拡張について \(34 ページ\)](#)
- [コンフィギュレーション生成のパフォーマンス強化の設定方法 \(35 ページ\)](#)
- [コンフィギュレーション生成のパフォーマンス強化の設定例 \(36 ページ\)](#)
- [その他の参考資料 \(36 ページ\)](#)
- [コンフィギュレーション生成のパフォーマンス強化の機能情報 \(38 ページ\)](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーション生成のパフォーマンス拡張に関する制限事項

コンフィギュレーション生成のパフォーマンス拡張機能を使用するデバイスには、大規模インターフェイス コンフィギュレーション ファイルを保存（キャッシュ保存）するための十分なメモリが必要です。たとえば、インターフェイスのコンフィギュレーションが 15KB のメモリを占有する場合、この機能を使用するには、さらに 15KB のメモリ領域を使用できる必要があります。

## コンフィギュレーション生成のパフォーマンス拡張について

### Cisco IOS XE ソフトウェアのコンフィギュレーションストレージ

Cisco IOS XE のソフトウェア コンフィギュレーション モデルでは、コンフィギュレーション状態は分散して維持され、各コンポーネントは独自のコンフィギュレーション状態を保持します。設定情報を取得するには、ソフトウェアは各コンポーネントをポーリングして、分散された情報を収集する必要があります。このコンフィギュレーション状態の取得操作は、不揮発生成（NVGEN）と呼ばれるプロセスによって実行され、実行中のシステム構成を表示またはコピーするためにコマンドラインインターフェイス（CLI）コマンド（**show running-configuration**、**write memory**、**copy system:running-configuration** など）で使用されます。NVGEN は、呼び出されると、各システムコンポーネントと、インターフェイスまたはその他の構成オブジェクトの各インスタンスを照会します。NVGEN がこれらのクエリーを実行しているシステムを通過するときに、実行コンフィギュレーションファイルが作成されます。

### コンフィギュレーション生成のパフォーマンス強化の利点

コンフィギュレーション生成のパフォーマンス強化機能が導入される前は、NVGEN は常にシステム全体を照会する必要があり、全体のコンフィギュレーションしか生成できませんでした。NVGEN 操作の完了には数分かかることがあるため、実行コンフィギュレーションの処理に必要な時間が原因となり、コンフィギュレーション管理上のパフォーマンスの問題が生じます。

コンフィギュレーション生成のパフォーマンス拡張機能は NVGEN 処理の実行時間を短縮し、特に多数のインターフェイス コンフィギュレーションを含む大規模なコンフィギュレーションファイルの管理で有用です。この機能はシステムメモリのインターフェイス コンフィギュレーション情報をキャッシュに保存し、変更された設定情報だけを取得することで、実行中のシステム構成を処理するコマンドの実行を高速化します。

# コンフィギュレーション生成のパフォーマンス強化の設定方法

## コンフィギュレーション生成のパフォーマンス強化の設定

コンフィギュレーション生成のパフォーマンス拡張をイネーブルにする作業を実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **parser config cache interface**
4. **end**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例：  Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例：  Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>parser config cache interface</b> 例：  Device(config)# parser config cache interface	特に大規模コンフィギュレーションファイルの場合に、実行中のシステム構成を管理するコマンドを CLI で実行するのに要する時間を短縮します。
ステップ 4	<b>end</b> 例：  Device(config)# end	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

# コンフィギュレーション生成のパフォーマンス強化の設定例

## コンフィギュレーション生成のパフォーマンス強化の設定例

次の例は、コンフィギュレーション生成のパフォーマンス強化機能を有効にする方法を示しています。

```
Device(config)# parser config cache interface
```

## コンフィギュレーション生成のパフォーマンス強化の確認例

システム コンフィギュレーション ファイルのコマンドをチェックして、**parserconfigcacheinterface** コマンドがイネーブルになっていることを確認できます。これは **showrunning-configuration EXEC** コマンドを入力すると表示されます。



- (注) 初めてコンフィギュレーション ファイルを表示する場合は、インターフェイス キャッシュが少ないため、それほどパフォーマンスの改善は見られません。ただし、**showrunning-config EXEC** コマンドなどの後続の NVGEN タイプのコマンドを入力すると、パフォーマンスが向上することがわかります。インターフェイスの構成が変更されるたびに、指定したインターフェイスのキャッシュがフラッシュされます。その他のインターフェイスデータはそのままキャッシュに残ります。インターフェイス コンフィギュレーションの修正後に NVGEN タイプのコマンドを入力すると、次回の NVGEN タイプのコマンドが入力されるまで改善はほとんど見られません。

```
Device# show running-config
!
!
parser config cache interface
!
!
```

## その他の参考資料

次の項に、コンフィギュレーション パーティショニング機能に関する参考資料を示します。

関連資料

関連項目	マニュアル タイトル
実行コンフィギュレーションのパフォーマンス強化：インターフェイスの <b>parserconfigcache</b>	コンフィギュレーション生成のパフォーマンス拡張
カスタマーサービスのプロビジョニング、コンフィギュレーションロールバック、コンフィギュレーションロッキング、およびコンフィギュレーションアクセスコントロール	『Contextual Configuration Diff Utility』
コンフィギュレーション管理：コンフィギュレーション変更およびロギング	コンフィギュレーション変更通知およびロギング
コンフィギュレーション管理：コンフィギュレーション変更およびロギングのクイック保存： <sup>1</sup>	コンフィギュレーションロガー永続性
Cisco IOS ソフトウェア コンフィギュレーションアクセス制御およびコンフィギュレーションセッションロック（「Config ロック」）。	排他的設定変更アクセスとアクセスセッションロック

<sup>1</sup> 「コンフィギュレーションロガー永続性」機能により、スタートアップ コンフィギュレーション全体を保存するのではなく、最後の startup-config ファイルが生成された時点から入力されたコマンドだけを保存できます。

標準

標準	タイトル
この機能に関連付けられている規格はありません。	--

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	--

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

テクニカル サポート

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。</p>	<p><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></p>

## コンフィギュレーション生成のパフォーマンス強化の機能情報

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

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

表 1: コンフィギュレーション生成のパフォーマンス強化機能の機能情報

機能名	リリース	機能情報
コンフィギュレーション生成のパフォーマンス拡張		<p>コンフィギュレーション生成のパフォーマンス拡張機能は、実行中のコンフィギュレーション ファイル情報の収集を高速化することでコンフィギュレーション管理を支援します。この機能は、多数のインターフェイスが構成された大規模なネットワークを管理する場合に特に便利です。</p> <p>この機能に関連付けられたコマンド：</p> <ul style="list-style-type: none"> <li>• <b>parser config cache interface</b></li> <li>• <b>parser config partition</b></li> <li>• <b>parser cache</b></li> </ul>





## 第 4 章

# 排他的設定変更アクセスとアクセスセッションロック

排他的設定変更アクセス機能（「コンフィギュレーションロック」機能とも呼びます）を使用すると、Cisco IOS XE の実行コンフィギュレーションに排他的に変更アクセスし、複数のユーザが同時に設定を変更するのを防ぐことができます。

この機能に対してアクセスセッション ロッキングを追加することで、排他的設定変更アクセス機能が拡張され、設定ロックを保持しているユーザが実行する **show** コマンドと **debug** コマンドの実行が常に優先されるようになります。他のユーザによって入力される **show** コマンドと **debug** コマンドは、設定ロックの所有者が開始したプロセスが終了した後でしか実行を許可されません。

排他的設定変更アクセス機能（「コンフィギュレーションロック」）は、コンフィギュレーションの置換とロールバック機能（「ロールバック ロック」）を補完するロック機構です。

- [機能情報の確認（39 ページ）](#)
- [設定のロックについて（40 ページ）](#)
- [排他的設定変更アクセスとアクセスセッションロックの設定方法（41 ページ）](#)
- [コンフィギュレーションのロックの設定例（44 ページ）](#)
- [その他の参考資料（44 ページ）](#)
- [排他的設定変更アクセスとアクセスセッションロックの機能情報（46 ページ）](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## 設定のロックについて

### 排他的設定変更アクセスとアクセス セッション ロック

Cisco IOS ソフトウェアが動作するデバイスは、デバイスのコンフィギュレーション状態を決定する実行コンフィギュレーションを保持しています。実行コンフィギュレーションを変更すると、デバイスの動作が変わります。Cisco IOS ソフトウェアでは、複数のユーザがデバイス CLI（デバイス コンソール、telnet セキュア シェル (SSH) など）を介して実行コンフィギュレーションを変更することが可能です。そのため運用環境によっては、複数のユーザが同時に Cisco IOS の実行コンフィギュレーションに変更を加えるのを防ぐと役立ちます。Cisco IOS の実行コンフィギュレーションへのアクセスを一時的に制限することにより、不注意による競合や、2人のユーザが実行コンフィギュレーションの同じ部分を設定しようとするのを防ぐことができます。

排他的設定変更アクセス機能（「コンフィギュレーションロック」機能とも呼びます）を使用すると、Cisco IOS の実行コンフィギュレーションに排他的に変更アクセスし、複数のユーザが同時に設定を変更するのを防ぐことができます。

この機能により、**configure terminal** コマンドを使用してグローバル コンフィギュレーション モードを開始した時点から、Cisco IOS の実行コンフィギュレーションへの排他的な変更アクセスが提供されます。これにより、「コンフィギュレーションロック」の効果が得られ、他のユーザによる Cisco IOS の実行コンフィギュレーションの変更を防止できます。コンフィギュレーションロックは、Cisco IOS コンフィギュレーション モードを終了すると自動的に解除されます。

排他的設定変更アクセス機能をイネーブルにするには、グローバル コンフィギュレーション モードで **configuration mode exclusive** コマンドを使用します。排他的設定変更アクセスは **auto** に設定できます。その場合は、他のユーザが **configure terminal** コマンドを使用するたびに Cisco IOS のコンフィギュレーションモードがロックされます。または **manual** に設定すると、**configure terminal lock** コマンドが発行されたときのみ Cisco IOS のコンフィギュレーションモードがロックされます。

排他的設定変更アクセス機能は、Cisco IOS リリース 12.2(25)S および 12.3(7)T で導入されたコンフィギュレーションの置換とロールバック機能を補完するロック機構です。

### アクセス セッション ロック

アクセスセッション ロッキング機能は、設定のロックを保持しているユーザが入力した **show** コマンドと **debug** コマンドの実行が常に優先されるように、排他的設定変更アクセス機能を拡張します。この機能は、同時設定アクセスを防ぐとともに、別のユーザが入力した **show** コマンドのように、他のコンフィギュレーションコマンドの実行中に同時に処理が実行されるのを防ぐためのオプションも提供します。この機能をイネーブルにすると、設定ロックを保持しているユーザが入力したコマンド（コンフィギュレーションコマンドなど）が、他のユーザが入力したコマンドよりも常に優先されます。

# 排他的設定変更アクセスとアクセスセッションロックの設定方法

## 排他的設定変更アクセスとアクセスセッションロックの有効化



- (注) Cisco IOS リリース 12.2(33)SRE から、排他的設定変更アクセスおよびアクセスセッションロック機能は、Cisco IOS ソフトウェアで使用できなくなりました。この機能の代わりに、パーサーの並行処理およびロックの改善機能を使用してください。詳細については、「パーサーの同時実行とロックの改善の有効化」を参照してください。

排他的設定変更アクセスとアクセスセッションロック機能を有効にするには、次のタスクを実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **configuration mode exclusive**
4. **end**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Router# 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	<b>configuration mode exclusive</b> 例： Router(config)# configuration mode exclusive	排他的設定変更アクセス（設定ロック機能）をイネーブルにします。 • コマンドがイネーブルになると、コンフィギュレーションセッションがシングルユーザ（排他）モードで実行されます。

	コマンドまたはアクション	目的
ステップ 4	<b>end</b> 例：  Router(config)# end	コンフィギュレーションセッションを終了して、CLI を特権 EXEC モードに戻します。

## 排他的設定変更アクセスの取得

### 手順の概要

1. **enable**
2. **configure terminal**
3. **configure terminal lock**
4. 変更を実行コンフィギュレーションに入力してシステムを設定します。
5. 次のいずれかを実行します。
  - **end**
  - または
  - **exit**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例：  Router> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例：  Router# 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	<b>configure terminal lock</b> 例：  Router(config)# configure terminal lock	（任意）Cisco IOS ソフトウェアを排他（シングルユーザ）モードでロックします。  • このコマンドは、 <b>configuration mode exclusive</b> コマンドを使用して設定ロックをイネーブルにしてある場合にだけ使用できます。  • このコマンドは、Cisco IOS リリース 12.3(14)T 以降のリリースで使用できます。
ステップ 4	変更を実行コンフィギュレーションに入力してシステムを設定します。	--

	コマンドまたはアクション	目的
ステップ 5	<p>次のいずれかを実行します。</p> <ul style="list-style-type: none"> <li>• <b>end</b></li> <li>• または</li> <li>• <b>exit</b></li> </ul> <p>例 :</p> <pre>Router(config)# end</pre> <p>例 :</p> <p>例 :</p> <pre>Router(config)# exit</pre>	<p>コンフィギュレーションセッションを終了し、ステップ 1 で取得したセッションロックを解放し、特権 EXEC モードに戻ります。</p> <p>(注) <b>end</b> コマンド、<b>exit</b> コマンド、Ctrl+Z のキーの組み合わせのいずれかで設定ロックを解放します。<b>end</b> コマンドの使用をお勧めします。</p>

## 設定ロックのモニタリングとトラブルシューティング

排他的設定変更アクセスおよびアクセスセッションロック機能モニタリングまたはトラブルシューティングするには、この作業のいずれかの手順または両方の手順を実行します。

### 手順の概要

1. **show configuration lock**
2. **debug configuration lock**

### 手順の詳細

#### ステップ 1 show configuration lock

現在の設定ロックのステータスと詳細（所有者、ユーザー、端末、ロック状態、ロッククラスなど）を表示するには、次のコマンドを使用します。

グローバル コンフィギュレーション モードを開始できない場合は、このコマンドを使用して、コンフィギュレーションセッションが別のユーザによってロックされているかどうか、およびそのユーザーが誰なのかを調べることができます。

例 :

#### ステップ 2 debug configuration lock

CiscoIOS 設定ロックのデバッグをイネーブルにするには、このコマンドを使用します（公開クラスロックまたはロールバック クラスロック）。

例 :

```
Router# debug configuration lock
```

```

Session1 from console
=====
Router# configure terminal lock
Configuration mode locked exclusively. The lock will be cleared once you exit out of configuration
mode using end/exit
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
Parser : LOCK REQUEST in EXCLUSIVE mode
Parser: <configure terminal lock> - Config. Lock requested by process <3> client <PARSER Client>
Parser: <configure terminal lock> - Config. Lock acquired successfully !
Router(config)#

```

## コンフィギュレーションのロックの設定例

### 自動モードでの排他的ロックの設定例

次に、**configurationmodeexclusive** コマンドを使用し、シングルユーザ自動コンフィギュレーションモードに対して、自動モードで排他的ロックをイネーブルにする例を示します。Cisco IOS コンフィギュレーションファイルが排他的にロックされたら、**showconfigurationlock** コマンドを使用してこのコンフィギュレーションを確認できます。

```

Router# configure terminal
Router(config)#
Router(config)# exit
Router# configure terminal
! Locks configuration mode exclusively.
Router# show configuration lock
Parser Configure Lock
Owner PID      : 10
User           : User1
TTY            : 3
Type           : EXCLUSIVE
State          : LOCKED
Class          : Exposed
Count          : 0
Pending Requests : 0
User debug info : 0

```

### 手動モードでの排他的ロックの設定例

## その他の参考資料

ここでは、設定のロックに関する関連資料について説明します。

## 関連資料

関連項目	マニュアル タイトル
コンフィギュレーション ファイルを管理するためのコマンド	『Cisco IOS Configuration Management Command Reference』
コンフィギュレーション ファイルの管理についての情報	コンフィギュレーション ファイルの管理

## 標準

標準	タイトル
この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。	--

## MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a>

## RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

## テクニカル サポート

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	<a href="http://www.cisco.com/en/US/support/index.html">http://www.cisco.com/en/US/support/index.html</a>

## 排他的設定変更アクセスとアクセスセッションロックの機能情報

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

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



表 2: 排他的設定変更アクセスとアクセスセッションロックの機能情報

機能名	リリース	機能情報
排他的設定変更アクセスとアクセスセッションロック	12.3(14)T 12.0(31)S 12.2(33)SRA 12.4(11)T 12.2(33)SXH 12.2(33)SB	<p>排他的設定変更アクセス機能（「コンフィギュレーションロック」機能とも呼びます）を使用すると、Cisco IOSの実行コンフィギュレーションに排他的に変更アクセスし、複数のユーザが同時に設定を変更するのを防ぐことができます。</p> <p>この機能に対してアクセスセッション ロッキングを追加することで、排他的設定変更アクセス機能が拡張され、設定ロックを保持しているユーザが実行する <b>show</b> コマンドと <b>debug</b> コマンドの実行が常に優先されるようになります。他のユーザによって入力される <b>show</b> コマンドと <b>debug</b> コマンドは、設定ロックの所有者が開始したプロセスが終了した後でしか実行を許可されません。</p> <p>排他的設定変更アクセス機能は、コンフィギュレーションの置換とロールバック機能（「ロールバック ロック」）を補完するロック機構です。</p> <p>設定ロック機能はリリース 12.0S に統合され、アクセスセッションロック機能の拡張が実装されました。 <b>configuration mode exclusive</b> コマンドが、キーワードオプション <b>config_wait</b>、<b>expire</b>、<b>interleave</b>、<b>lock-show</b>、<b>retry_wait</b>、および <b>terminate</b> を含むように拡張されました。 <b>show configuration lock</b> コマンドの出力が改良されました。</p> <p>拡張機能は、リリース 12.2(33)SRA、12.4(11)T、12.2(33)SXH、および 12.2(33)SB に統合されました。</p> <p>この機能に関する詳細については、次の各項を参照してください。</p> <ul style="list-style-type: none"> <li>• 設定のロックについて</li> <li>• 設定ロックの設定方法</li> </ul> <p>次のコマンドが導入または変更されました：<b>clear configuration lock</b>、<b>configuration mode exclusive</b>、<b>configure terminal lock</b>。</p>

機能名	リリース	機能情報
パーサーの並行処理およびロッキングの改善	12.2(33)SRE 15.1(1)T	<p>パーサーの並行処理およびロッキングの改善機能は、要求したプロセスに対して排他的なアクセスを許可し、他のプロセスが Cisco IOS の設定に同時にアクセスできないようにするための、共通のインターフェイスを提供します。この機能により、ロックを保持しているユーザーにのみアクセスが許可され、他のクライアントがコンフィギュレーションにアクセスできなくなります。</p> <p>この機能に関する詳細については、次の各項を参照してください。</p> <ul style="list-style-type: none"> <li>• パーサーの並行処理およびロッキングの改善</li> <li>• パーサーの並行処理およびロッキングの改善のイネーブル化</li> </ul> <p>次のコマンドが導入または変更されました：<b>parser command serializer</b>、<b>test parser session-lock</b>。</p>



## 第 5 章

# コンフィギュレーションの置換とロールバック

コンフィギュレーションの置換とロールバック機能により、現在の実行コンフィギュレーションを、保存しておいた Cisco IOS コンフィギュレーションファイルで置換することができます。この機能は、コンフィギュレーションを保存しておいた状態へ戻すために使用でき、そのコンフィギュレーションファイルが保存された後にどのような変更が加えられても、効果的にロールバックさせることができます。

- [機能情報の確認 \(49 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの前提条件 \(50 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの制約事項 \(50 ページ\)](#)
- [コンフィギュレーションの置換とロールバックについて \(51 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの使用方法 \(54 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの設定例 \(61 ページ\)](#)
- [その他の参考資料 \(64 ページ\)](#)
- [コンフィギュレーションの置換とロールバックの機能情報 \(66 ページ\)](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリースノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーションの置換とロールバックの前提条件

コンフィギュレーションの置換とロールバックの機能に対する入力となるコンフィギュレーションファイルの形式は、標準の Cisco ソフトウェア コンフィギュレーションファイルの、次に示すインデント規則に準拠している必要があります。

- 新しい行のすべてのコマンドは、コマンドがコンフィギュレーションサブモードにない限り、インデントなしで開始します。
- レベル1 コンフィギュレーションサブモード内のコマンドは、スペース1個分インデントします。
- レベル2 コンフィギュレーションサブモード内のコマンドは、スペース2個分インデントします。
- 以下、続くサブモード内のコマンドは、同じようにインデントします。

これらのインデント規則には、ソフトウェアが **show running-config** や **copy running-config destination-url** などのコマンドのコンフィギュレーションファイルを作成する方法が記述されています。シスコデバイスで生成されるコンフィギュレーションファイルは、いずれもこうした規則に従います。

2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリが必要です。

## コンフィギュレーションの置換とロールバックの制約事項

デバイスに、2つのコンフィギュレーションファイル（現在の実行コンフィギュレーションと、保存された置換用コンフィギュレーション）を合わせたサイズより大きな空きメモリがない場合、コンフィギュレーション置換操作は実行されません。

ネットワークデバイスの物理コンポーネント（物理インターフェイスなど）に関連する特定の Cisco コンフィギュレーション コマンドは、実行コンフィギュレーションについて追加または削除することはできません。たとえば、コンフィギュレーション置換操作を行っても、そのインターフェイスがデバイス上に物理的に存在する場合、現在の実行コンフィギュレーションから **interface ethernet 0** コマンド行を削除することはできません。同様に、**interface ethernet 1** コマンド行は、そのようなインターフェイスがデバイス上に物理的に存在しない場合、実行コンフィギュレーションに追加することはできません。コンフィギュレーション置換操作でこのタイプの変更を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

非常にまれなケースですが、ルータをリロードしないと特定の Cisco コンフィギュレーション コマンドを実行コンフィギュレーションから削除できないことがあります。コンフィギュレーション置換操作でこのタイプのコマンドの削除を試行すると、その特定のコマンド行が失敗したことを示すエラーメッセージが表示されます。

## コンフィギュレーションの置換とロールバックについて

### コンフィギュレーションアーカイブ

Cisco IOS コンフィギュレーションアーカイブは、**configurereplace** コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、Cisco IOS コンフィギュレーションファイルのアーカイブの保存、整理、管理を行うことを目的としたメカニズムです。この機能の導入前にも、実行コンフィギュレーションのコピーを **copyrunning-configdestination-url** コマンドを使用して保存し、ローカルやリモートに置換ファイルを保管できました。ただし、この方法ではファイルの自動管理を行うことはできませんでした。一方、コンフィギュレーションの置換とロールバック機能では、実行コンフィギュレーションファイルを自動的に Cisco IOS コンフィギュレーションアーカイブに保存できます。アーカイブされたファイルはコンフィギュレーションのチェックポイントとして参照することができ、**configurereplace** コマンドを使用して以前のコンフィギュレーション状態に戻すために利用できます。

**archiveconfig** コマンドを使用すると、Cisco IOS コンフィギュレーションをコンフィギュレーションアーカイブに保存できます。その場合、標準のディレクトリとファイル名のプレフィクスが使用され、バージョン番号（およびオプションでタイムスタンプ）が自動的に付加されます。バージョン番号は連続したファイルを保存するごとに、1 つずつ大きくなります。この機能により、保存した Cisco IOS コンフィギュレーションファイルを一貫して識別できます。アーカイブに保存する実行コンフィギュレーションの数は指定することができます。アーカイブ内のファイル数が上限値に達すると、次に最新のファイルが保存されるときに、最も古いファイルが自動的に消去されます。**showarchive** コマンドを使用すると、Cisco IOS コンフィギュレーションアーカイブに保存されているすべてのコンフィギュレーションファイルに関する情報が表示されます。

コンフィギュレーションファイルを保存する Cisco IOS コンフィギュレーションアーカイブは、**configurereplace** コマンドで使用することによって、次のファイルシステムに配置できます。

- お使いのプラットフォームが `disk0--disk 0: disk1: ftp: pram: rcv: slavedisk0: slavedisk1:` または `tftp:`
- `disk0` が不在プラットフォーム : `ftp:`、`http:`、`pram:`、`rcv:`、`tftp:`。

### コンフィギュレーションの置換

**configurereplace** コマンドにより、現在の実行コンフィギュレーションを、保存しておいた Cisco IOS コンフィギュレーションファイルで置換することができます。この機能は、コンフィ

ギュレーションを保存しておいた状態へ戻すために使用することができ、そのコンフィギュレーション状態が保存された後にどのような変更が加えられても、効果的にロールバックさせることができます。

**configurereplace** コマンドを使用するときは、現在の実行コンフィギュレーションと置換するための、保存された Cisco IOS コンフィギュレーションファイルを指定する必要があります。置換ファイルは、Cisco IOS デバイスによって作成された完全なコンフィギュレーション (**copyrunning-configdestination-url** コマンドによって作成されたものなど) であることが必要です。あるいは、置換ファイルを外部的に作成する場合は Cisco IOS デバイスが作成するファイル形式に完全に準拠していなければなりません。**configurereplace** コマンドを入力すると、現在の実行コンフィギュレーションが指定された置換コンフィギュレーションと比較され、一連の diff が生成されます。2つのファイルの比較に使用されるアルゴリズムは、**showarchiveconfigdifferences** コマンドで使用されるものと同じです。置換コンフィギュレーションの状態になるよう、diffの結果が Cisco IOS パーサーによって適用されます。diffのみが適用されるため、現在の実行コンフィギュレーション上にすでに存在していた設定コマンドを再適用することにより生じる、潜在的なサービスの中断を避けられます。このアルゴリズムでは、順序に依存するコマンド (アクセスリストなど) へのコンフィギュレーション変更を、複数のパスプロセスを通して効果的に実行します。通常的环境下では、コンフィギュレーション置換操作の完了に必要なパスは3つまでであり、ループ動作を防ぐためのパスは最大5つまでに制限されます。

Cisco IOS **copysource-urlrunning-config** コマンドは、保存された Cisco IOS コンフィギュレーションファイルを実行コンフィギュレーションへコピーするためによく使用されます。

**copysource-urlrunning-config** コマンドを **configurereplacetarget-url** コマンドの代わりに使用する場合、主な相違点として次の点に注意が必要です。

- **copysource-urlrunning-config** コマンドはマージ動作であり、ソースファイルと現在の実行コンフィギュレーションの両方のコマンドをすべて保持します。このコマンドでは、現在の実行コンフィギュレーションにのみ含まれ、ソースファイルには存在しないコマンドが削除されることはありません。これに対して、**configurereplacetarget-url** コマンドでは、置換ファイルに存在しないコマンドが現在の実行コンフィギュレーションから削除され、追加する必要のあるコマンドが現在の実行コンフィギュレーションに追加されます。
- **copysource-urlrunning-config** コマンドでは、現在の実行コンフィギュレーションにすでに存在しているかどうかにかかわらず、ソースファイル中のすべてのコマンドが適用されます。このアルゴリズムは効率的でない上、場合によってはサービスの停止が発生します。これに対して、**configurereplacetarget-url** コマンドでは適用が必要なコマンドのみを適用し、現在の実行コンフィギュレーションに存在しているコマンドは再適用されません。
- **copysource-urlrunning-config** コマンドでは部分的なコンフィギュレーションファイルもコピー元として使用できますが、**configurereplacetarget-url** コマンドの置換ファイルとして使用できるのは、完全な Cisco IOS コンフィギュレーションファイルのみです。

Cisco IOS リリース 12.2(25)S および 12.3(14)T では、コンフィギュレーション置換操作にロック機能が導入されました。**configurereplace** コマンドが使用されると、コンフィギュレーション置換の動作中、デフォルトで実行コンフィギュレーションファイルがロックされます。このロックメカニズムによって、置換動作の実行中に他のユーザが実行コンフィギュレーションを変更しようとしたために、置換動作の不正終了が発生することを防止できます。**nolock** キー

ワードを **configurereplace** コマンドの実行時に使用すると、実行コンフィギュレーションのロックをディセーブルにできます。

実行コンフィギュレーションのロックは、コンフィギュレーションの置換動作終了時に自動的にクリアされます。 **showconfigurationlock** コマンドを使用すると、現在実行コンフィギュレーションに適用されているロックをすべて表示できます。



- (注) IOS から供給されていないコンフィギュレーション（カスタムに記述したコンフィギュレーションなど）を使用してコンフィギュレーションを置換するシナリオでは、ログインバナーに EXT 文字（ASCII コード 003）ではない区切り記号がある場合、バナーのコンフィギュレーションは拒否され、置換後のコンフィギュレーションには含まれません。正常に動作しない区切り文字には、^C、%、#、CC があります。

## コンフィギュレーション ロールバック

ロールバックの概念は、データベースの操作ではトランザクションプロセスモデルに由来します。データベーストランザクションでは、あるデータベースのテーブルに一連の変更を加えることがあります。その後、変更を実行する（変更を恒久的に適用する）か、変更をロールバックする（変更を破棄してテーブルを以前の状態に戻す）かを選択することになります。ここでロールバックが意味するのは、変更のログを含んだジャーナルファイルが破棄され、何の変更も加えられないということです。ロールバック操作の結果として、加えた変更が適用される前の状態に戻ります。

**configurereplace** コマンドを使用することで、以前のコンフィギュレーション状態へ戻ることが可能になり、コンフィギュレーション状態の保存後に加えた変更を効率的にロールバックさせることができます。Cisco IOS コンフィギュレーション ロールバックは、適用された一連の変更をもとにロールバック動作を行うのではなく、保存された Cisco コンフィギュレーションファイルに基づいた特定のコンフィギュレーション状態へ戻るといったコンセプトを採用しています。このコンセプトは、チェックポイント（データベースの保存されたバージョン）に特定の状態を保存しておくという、データベースの考え方に類似しています。

コンフィギュレーションのロールバック機能が必要な場合、コンフィギュレーションの変更前先立って Cisco IOS 実行コンフィギュレーションを保存する必要があります。次に、コンフィギュレーションを変更した後に (**configurereplace target-url** コマンドを使用し) 保存したコンフィギュレーションファイルを使って変更をロールバックします。さらに、保存された Cisco IOS コンフィギュレーションファイルならどれでも置換コンフィギュレーションとして指定できるため、ジャーナルファイルによるロールバック モデルの一部のように、ロールバックの数が制限されることもありません。

### コンフィギュレーション ロールバック変更確認の操作

コンフィギュレーションロールバック変更確認機能は、コンフィギュレーションの変更を確認条件を追加できる機能です。この機能により、要求された変更の確認が設定済みの時間枠以内に受信されない場合にロールバックを行うことができます。コマンドの失敗を、コンフィギュレーションのロールバックをトリガーするように設定することもできます。

次に、このプロセスを実施するための手順の概要を示します。

1. 新しいオプションを使用すると、コンフィギュレーションの変更の確認を要求できます（確認の時間制限を指定する必要があります）。
2. 確認コマンドを入力する必要があります。要求された制限時間内に確認を入力しないと、コンフィギュレーションは以前の状態に戻ります。

## コンフィギュレーションの置換とロールバックの利点

- コンフィギュレーションの変更を効率的にロールバックさせて、以前のコンフィギュレーション状態へ戻ることが可能。
- ルータをリロードしたり、CLIで実行コンフィギュレーションファイルに加えた変更を手動で元に戻したりすることなく、現在の実行コンフィギュレーションファイルをスタートアップコンフィギュレーションファイルと置換できるため、システムのダウンタイムが減少。
- 保存しておいたどの Cisco IOS コンフィギュレーション状態に戻すことも可能。
- 追加や削除が必要なコマンドだけが影響される場合、ルータに完全なコンフィギュレーションファイルを適用することができるため、コンフィギュレーションの変更がシンプルに。
- **configure replace** コマンドを **copy source-url running-config** コマンドの代用として使用すると、現在の実行コンフィギュレーションにある既存のコマンドが再度適用されないため、効率が向上し、サービス停止のリスクが回避されます。

## コンフィギュレーションの置換とロールバックの使用方法

### コンフィギュレーションアーカイブの作成

**configurereplace** コマンドを使用するうえで前提条件となる設定はありません。**configurereplace** コマンドと、Cisco IOS コンフィギュレーションアーカイブおよび **archiveconfig** コマンドとの併用は任意ですが、コンフィギュレーションロールバックのシナリオでは大きな利点があります。**archiveconfig** コマンドを使用する前に、コンフィギュレーションアーカイブを設定しておく必要があります。コンフィギュレーションアーカイブの特性を設定するには、次の作業を実行します。

#### 手順の概要

1. **enable**
2. **configure terminal**



3. **archive**
4. **path** *url*
5. **maximum** *number*
6. **time-period** *minutes*
7. **end**
8. **archive config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 ・パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>archive</b> 例： Device(config)# archive	アーカイブ コンフィギュレーション モードを開始します。
ステップ 4	<b>path</b> <i>url</i> 例： Device(config-archive)# path flash:myconfig	Cisco IOS コンフィギュレーション アーカイブの場所と、ファイル名のプレフィックスを指定します。 (注) パスのところでファイルの代わりにディレクトリを指定する場合、ディレクトリ名は <b>path flash:/directory/</b> のように後ろにスラッシュを付ける必要があります。このスラッシュはファイル名の後ろでは必要ありません。ディレクトリを指定する場合にだけ使います。
ステップ 5	<b>maximum</b> <i>number</i> 例： Device(config-archive)# maximum 14	(任意) Cisco IOS コンフィギュレーション アーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を設定します。 ・ <i>number</i> 引数は、Cisco IOS コンフィギュレーション アーカイブに保存される実行コンフィギュレーションのアーカイブファイル数の上限値を示します。有効な値は 1 ～ 14 で、デフォルトは 10 です。

	コマンドまたはアクション	目的
		(注) このコマンドを使用する前に、 <b>path</b> コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 6	<b>time-period</b> <i>minutes</i> 例： <pre>Device(config-archive)# time-period 10</pre>	(任意) Cisco IOS コンフィギュレーションアーカイブに実行コンフィギュレーションのアーカイブファイルを自動保存する間隔を設定します。 <ul style="list-style-type: none"> <li>• Cisco IOS コンフィギュレーションアーカイブに現在の実行コンフィギュレーションのアーカイブファイルをどれほどの頻度で自動保存するかを、<b>minutes</b> 引数により分単位で指定します。</li> </ul> (注) このコマンドを使用する前に、 <b>path</b> コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 7	<b>end</b> 例： <pre>Device(config-archive)# end</pre>	特権 EXEC モードに戻ります。
ステップ 8	<b>archive config</b> 例： <pre>Device# archive config</pre>	現在の実行設定ファイルを設定アーカイブに保存します。 (注) このコマンドを使用する前に、 <b>path</b> コマンドを設定する必要があります。

## コンフィギュレーションの置換やロールバック操作の実行

保存された Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションファイルを置換するには、次の作業を実行します。



- (注) この手順の前に、コンフィギュレーションアーカイブを作成しておく必要があります。詳細については、コンフィギュレーションアーカイブの作成を参照してください。次に、現在の実行コンフィギュレーションで問題が生じた場合に、アーカイブしておいたコンフィギュレーションに戻す手順の詳細を示します。

手順の概要

1. **enable**
2. **configure replace** *target-url* [**nolock**] [**list**] [**force**] [**ignorecase**]  
[**reverttrigger**[**error**]/**timerminutes**]|**timeminutes**]
3. **configure revert** {**now** |**timer**{*minutes*|*idleminutes*}}
4. **configure confirm**
5. **exit**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>
ステップ 2	<p><b>configure replace</b> <i>target-url</i> [<b>nolock</b>] [<b>list</b>] [<b>force</b>] [<b>ignorecase</b>] [<b>reverttrigger</b>[<b>error</b>]/<b>timerminutes</b>] <b>timeminutes</b>]</p> <p>例 :</p> <pre>Device# configure replace flash:myconfig-1 list time 30</pre>	<p>保存しておいた Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションファイルを置換します。</p> <ul style="list-style-type: none"> <li>• <b>target-url</b> 引数は、<b>archiveconfig</b> コマンドで作成されたコンフィギュレーションファイルなど、現在の実行コンフィギュレーションと置換する、保存された Cisco IOS コンフィギュレーションファイルの URL です (Cisco IOS ファイルシステムでアクセス可能なもの)。</li> <li>• <b>list</b> キーワードは、コンフィギュレーション置換動作のパスごとに、Cisco IOS ソフトウェアパーサーによって適用されるコマンドラインのリストを表示します。実行されたパスの総数も表示されます。</li> <li>• <b>force</b> キーワードは、現在の実行コンフィギュレーションから指定した Cisco IOS コンフィギュレーションファイルへの置換を、確認プロンプトを出さずに実行します。</li> <li>• <b>timeminutes</b> キーワードおよび引数は、現在の実行コンフィギュレーションファイルの置換確認のために <b>configureconfirm</b> コマンドを入力しなければならない制限時間 (分単位) を指定します。<b>configureconfirm</b> コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます (つまり、現在の実行コンフィギュレーションファイ</li> </ul>

	コマンドまたはアクション	目的
		<p>ルが <b>configurereplace</b> コマンド入力以前のコンフィギュレーション状態へと回復されます)。</p> <ul style="list-style-type: none"> <li>• <b>nolock</b> キーワードは、コンフィギュレーション置換操作中に他のユーザが実行コンフィギュレーションを変更しないように実行コンフィギュレーションファイルをロックする機能をオフにします。</li> <li>• <b>reverttrigger</b> キーワードは、元のコンフィギュレーションへ戻すトリガーを次の内容から設定します。 <ul style="list-style-type: none"> <li>• <b>error</b> : エラー時に元のコンフィギュレーションに戻します。</li> <li>• <b>timerminutes</b> : 指定した時間が過ぎると元のコンフィギュレーションに戻します。</li> </ul> </li> <li>• <b>ignorecase</b> キーワードで、コンフィギュレーションに確認コマンドの大文字と小文字の区別を無視させることができます。</li> </ul>
<p>ステップ 3</p>	<p><b>configure revert {now  timer{minutes}idleminutes}}</b></p> <p>例 :</p> <pre>Device# configure revert now</pre> <p>例 :</p>	<p>(任意) 時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、特権 EXEC モードで <b>configurerevert</b> コマンドを使用します。</p> <ul style="list-style-type: none"> <li>• <b>now</b> : ロールバックをただちにトリガーします。</li> <li>• <b>timer</b> : コンフィギュレーションを元に戻すタイマーをリセットします。 <ul style="list-style-type: none"> <li>• 元に戻す時間を分単位で新たに指定するには、<i>minutes</i> 引数を <b>timer</b> キーワードとともに使用します。</li> <li>• 保存されたコンフィギュレーションに戻すまでに、操作が行われないアイドル時間を最大どれほど長く許容できるかを設定するには、分単位の時間とともに <b>idle</b> キーワードを使用します。</li> </ul> </li> </ul>

	コマンドまたはアクション	目的
ステップ 4	<p><b>configure confirm</b></p> <p>例 :</p> <pre>Device# configure confirm</pre>	<p>(任意) 保存しておいた Cisco IOS コンフィギュレーションファイルの現在の実行コンフィギュレーションファイルへの置換を確認します。</p> <p>(注) このコマンドは、<b>configurereplace</b> コマンドの <b>timeseconds</b> キーワードおよび引数が指定されている場合にのみ使用します。</p>
ステップ 5	<p><b>exit</b></p> <p>例 :</p> <pre>Device# exit</pre>	<p>ユーザ EXEC モードに戻ります。</p>

## 機能のモニタリングおよびトラブルシューティング

コンフィギュレーションの置換とロールバック機能をモニタおよびトラブルシューティングするには、この手順を実行します。

### 手順の概要

1. **enable**
2. **show archive**
3. **debug archive versioning**
4. **debug archive config timestamp**
5. **exit**

### 手順の詳細

#### ステップ 1 enable

このコマンドを使用して、特権 EXEC モードをイネーブルにします。パスワードを入力します（要求された場合）。次に例を示します。

例 :

```
Device> enable
Device#
```

#### ステップ 2 show archive

Cisco IOS コンフィギュレーションアーカイブに保存されているファイルに関する情報を表示するには、次のコマンドを使用します。次に例を示します。

例 :

```
Device# show archive
There are currently 1 archive configurations saved.
```

```
The next archive file will be named flash:myconfig-2
Archive #  Name
0
1          flash:myconfig-1 <- Most Recent
2
3
4
5
6
7
8
9
10
11
12
13
14
```

次に、実行コンフィギュレーションのアーカイブ ファイルをいくつか保存した状態で **showarchive** コマンドを使用した場合の出力例を示します。この例では、保存されるアーカイブ ファイルの最大数が 3 に設定されています。

例：

```
Device# show archive
There are currently 3 archive configurations saved.
The next archive file will be named flash:myconfig-8
Archive #  Name
0
1          :Deleted
2          :Deleted
3          :Deleted
4          :Deleted
5          flash:myconfig-5
6          flash:myconfig-6
7          flash:myconfig-7 <- Most Recent
8
9
10
11
12
13
14
```

### ステップ 3 debug archive versioning

このコマンドを使用して、Cisco IOS コンフィギュレーションアーカイブのアクティビティのデバッグを有効にして、コンフィギュレーションの置換とロールバックをモニタおよびトラブルシューティングします。次に例を示します。

例：

```
Device# debug archive versioning
Jan  9 06:46:28.419:backup_running_config
Jan  9 06:46:28.419:Current = 7
Jan  9 06:46:28.443:Writing backup file flash:myconfig-7
Jan  9 06:46:29.547: backup worked
```

### ステップ 4 debug archive config timestamp

このコマンドを使用して、コンフィギュレーション置換操作の各必須段階の処理時間、および操作中のコンフィギュレーションファイルのサイズのデバッグをイネーブルにします。次に例を示します。

例：

```
Device# debug archive config timestamp
Device# configure replace flash:myconfig force
Timing Debug Statistics for IOS Config Replace operation:
  Time to read file slot0:sample_2.cfg = 0 msec (0 sec)
  Number of lines read:55
  Size of file          :1054
Starting Pass 1
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:93
  Size of file          :2539
  Time taken for positive rollback pass = 320 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for negative incremental diffs pass = 59 msec (0 sec)
  Time taken by PI to apply changes = 0 msec (0 sec)
  Time taken for Pass 1 = 380 msec (0 sec)
Starting Pass 2
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:55
  Size of file          :1054
  Time taken for positive rollback pass = 0 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for Pass 2 = 0 msec (0 sec)
Total number of passes:1
Rollback Done
```

## ステップ5 exit

このコマンドを使用して、ユーザ EXEC モードに戻ります。次に例を示します。

例：

```
Device# exit
Device>
```

# コンフィギュレーションの置換とロールバックの設定例

## コンフィギュレーションアーカイブの作成例

次の例は、Cisco IOS コンフィギュレーションアーカイブの初期設定を実行する方法を示しています。この例では、flash:myconfig がコンフィギュレーションアーカイブの保存位置およびファイル名のプレフィックスとして設定され、保存するアーカイブファイルが最大 10 個に設定されます。

```
configure terminal
!
archive
 path flash:myconfig
```

```
maximum 10
end
```

## 保存された Cisco IOS コンフィギュレーション ファイルでの現在の実行コンフィギュレーションの置換：例

次の例では、flash:myconfig という名前で保存された Cisco IOS コンフィギュレーションファイルで現在の実行コンフィギュレーションを置換する方法を示します。configure replace コマンドでは、確認プロンプトでインタラクティブに操作を進めます。

```
Device# configure replace flash:myconfig
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
```

次の例では、コンフィギュレーション置換操作中に適用されるコマンドラインを表示するために、list キーワードを指定しています。

```
Device# configure replace flash:myconfig list
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
!Pass 1
!List of Commands:
no snmp-server community public ro
snmp-server community mystring ro

end
Total number of passes: 1
Rollback Done
```

## スタートアップコンフィギュレーション ファイルの復元：例

次の例に、configure replace コマンドを使用して Cisco IOS スタートアップコンフィギュレーションファイルへ復元する方法を示します。この例は、オプションの force キーワードを使用して、インタラクティブユーザプロンプトをオーバーライドする方法を示しています。

```
Device# configure replace nvram:startup-config force
Total number of passes: 1
Rollback Done
```



## 例：configure confirm コマンドを使用したコンフィギュレーション置換操作の実行

次に、**configure replace** コマンドを **time minutes** キーワードおよび引数とともに使用する例を示します。現在の実行コンフィギュレーションファイルの置換を実行するには、指定の制限時間内に **configure confirm** コマンドを入力する必要があります。**configure confirm** コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルが **configure replace** コマンド入力以前のコンフィギュレーション状態へと回復されます）。

```
Device# configure replace nvram:startup-config time 120
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
Device# configure confirm
```

次に、**configure revert** コマンドを **timer** キーワードとともに使用する例を示します。時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、**configure revert** コマンドを入力する必要があります。

```
Device# configure revert timer 100
```

## コンフィギュレーションロールバック操作の実行：例

次の例は、現在実行中のコンフィギュレーションへの変更を行い、その変更をロールバックする方法を示しています。コンフィギュレーションロールバック操作の一部として、ファイルに変更を加える前に現在の実行コンフィギュレーションを保存する必要があります。この例では、現在の実行コンフィギュレーションの保存に **archiveconfig** コマンドが使用されています。**configurereplace** コマンドで生成された出力は、ロールバック操作を完了するために1つのパスのみが実行されたことを示します。



(注) **archiveconfig** コマンドを使用する前に、**path** コマンドを設定して Cisco IOS コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。

次のように、設定アーカイブの現在実行中のコンフィギュレーションを保存します。

```
archive config
```

それから、次の例に示すようにコンフィギュレーションの変更を入力します。

```
configure terminal
!
user netops2 password rain
```

```
user netops3 password snow
exit
```

実行コンフィギュレーションファイルに変更を加えた後、それらの変更をロールバックさせて、変更前のコンフィギュレーションに戻したくなるとします。**showarchive** コマンドは、交換ファイルとして使用される設定のバージョンを確認するために使用されます。次の例に示すように、**configurereplace** コマンドは交換コンフィギュレーションファイルへ戻すために使用されます。

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfig-2
Archive # Name
0
1 flash:myconfig-1 <- Most Recent
2
3
4
5
6
7
8
9
10
Device# configure replace flash:myconfig-1
Total number of passes: 1
Rollback Done
```

## その他の参考資料

次の項に、コンフィギュレーションの置換とロールバック機能に関する参考資料を示します。

### 関連資料

関連項目	マニュアルタイトル
設定ロック	『 <i>Exclusive Configuration Change Access and Access Session Locking</i> 』
コンフィギュレーションファイルを管理するためのコマンド	<i>Cisco IOS Configuration Fundamentals Command Reference</i>
コンフィギュレーションファイルの管理についての情報	コンフィギュレーションファイルの管理
コンフィギュレーションのコンテキスト差分ユーティリティ機能の使用	『 <i>Contextual Configuration Diff Utility</i> 』

**標準**

標準	タイトル
この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。	--

**MIB**

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャセットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a>

**RFC**

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

**テクニカル サポート**

説明	リンク
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	<a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a>

# コンフィギュレーションの置換とロールバックの機能情報

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

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

表 3: コンフィギュレーションの置換とロールバックの機能情報

機能名	リリース	機能情報
コンフィギュレーションの置換とロールバック		<p>コンフィギュレーションの置換とロールバック機能により、現在の実行コンフィギュレーションを、保存しておいた Cisco IOS コンフィギュレーションファイルで置換することができます。この機能は、コンフィギュレーションを保存しておいた状態へ戻すために使用でき、そのコンフィギュレーションファイルが保存された後にどのような変更が加えられても、ロールバックさせることができます。</p> <p>機能情報について、次の項で説明します。</p> <p>この機能により、次のコマンドが変更されました。 <b>archive config</b>、<b>configure confirm</b>、<b>configure replace</b>、<b>debug archive config timestamp</b>、<b>debug archive versioning</b>、<b>maximum</b>、<b>path (archive configuration)</b>、<b>show archive</b>、<b>show configuration lock</b>、<b>time-period</b></p>
コンフィギュレーションのバージョン管理		<p>コンフィギュレーションのバージョン管理機能により、Cisco IOS 実行コンフィギュレーションのコピーをデバイス上やデバイス外で維持および管理することができます。コンフィギュレーション置換機能では、実行コンフィギュレーションの保存されたコピーへのロールバックを行うためにコンフィギュレーションバージョン管理機能を使用します。</p>

機能名	リリース	機能情報
排他的設定変更アクセス		<p>排他的設定変更アクセス機能（「コンフィギュレーションロック」機能とも呼びます）を使用すると、Cisco IOS の実行コンフィギュレーションに排他的に変更アクセスし、複数のユーザが同時に設定を変更するのを防ぐことができます。</p> <p>この機能により、<b>show configuration lock</b> コマンドが変更され、コンフィギュレーションの置換とロールバック機能に適用されます。</p> <p>詳しくは、別のモジュール『Exclusive Configuration Change Access and Access Session Locking』を参照してください。</p>
コンフィギュレーションロールバック変更確認		<p>コンフィギュレーションロールバック変更確認機能により、コンフィギュレーション変更の実行に際して確認を要求するようオプションで設定できます。</p> <p>この確認が受信できない場合、コンフィギュレーションは変更が適用される前の状態に戻されます。</p> <p>このメカニズムは、ネットワークデバイスとユーザまたは管理アプリケーションとの接続に、誤ったコンフィギュレーション変更に起因する切断を防止するものです。</p> <p>この機能に関する詳細については、次の各項を参照してください。</p> <p>この機能により、次のコマンドが変更されました。<b>configure confirm</b>、<b>configure replace</b>、<b>configure revert</b>、<b>configure terminal</b></p>





## 第 6 章

# コンフィギュレーションのコンテキスト差分ユーティリティ

コンフィギュレーションのコンテキスト差分ユーティリティ機能は、2つのコンフィギュレーションファイル（Cisco IOS XE Integrated File System（IFS）を通じてアクセス可能）を行ごとに比較し、その間の違いの一覧を生成する機能を提供します。生成される出力には、追加、変更、または削除されたコンフィギュレーション行に関する情報と、変更されたコンフィギュレーション行が存在するコンフィギュレーションモードが含まれます。

- [機能情報の確認（69 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティの前提条件（70 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティの制限事項（70 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティについて（70 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティの使い方（72 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティの設定例（73 ページ）](#)
- [その他の参考資料（77 ページ）](#)
- [コンフィギュレーションのコンテキスト差分ユーティリティの機能情報（79 ページ）](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリースノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーションのコンテキスト差分ユーティリティの前提条件

コンフィギュレーションのコンテキスト差分ユーティリティ機能で使用されるコンフィギュレーションファイルの形式は、次に示す標準的な Cisco IOS XE コンフィギュレーションファイルのインデントルールに準拠している必要があります。

- 新しい行のすべてのコマンドは、コマンドがコンフィギュレーションサブモードにない限り、インデントなしで開始します。
- レベル1 コンフィギュレーションサブモード内のコマンドは、スペース1個分インデントします。
- レベル2 コンフィギュレーションサブモード内のコマンドは、スペース2個分インデントします。
- 以下、続くサブモード内のコマンドは、同じようにインデントします。

ルータには、比較対象の2つのコンフィギュレーションファイルの合計サイズよりも大きい連続したメモリブロックが必要です。

## コンフィギュレーションのコンテキスト差分ユーティリティの制限事項

比較対象の2つのコンフィギュレーションファイルの合計サイズよりも大きい連続したメモリブロックがデバイスにない場合、diff 操作は失敗します。

## コンフィギュレーションのコンテキスト差分ユーティリティについて

### コンフィギュレーションのコンテキスト差分ユーティリティの利点

コンフィギュレーションのコンテキスト差分ユーティリティ機能は、2つのコンフィギュレーションファイル（Cisco IOS XE File System (IFS) を通じてアクセス可能）を行ごとに比較し、その間の違いの一覧を生成する機能を提供します。生成される出力には、次の項目に関する情報が含まれます。

- 追加、変更、削除された設定行。
- 変更された設定行が存在するコンフィギュレーションモード。



- 順序に依存する設定行の場所の変更。たとえば、**ip access-list** コマンドと **community-lists** コマンドは、コンフィギュレーションファイル内での、同じ種類の他の Cisco IOS コマンドとの相対的な順序による影響を受けます。

## コンフィギュレーションのコンテキスト差分ユーティリティの出力形式

### diff 操作

コンフィギュレーションのコンテキスト差分ユーティリティ機能は、2つのコンフィギュレーションファイルのファイル名を入力として使用します。**show archive config differences** コマンドの使用により、指定されたファイルに対して **diff** 操作を実行し、2つのファイル間の差分のリストを出力として生成します。出力の解釈は、コマンドで指定される2つのファイルの順序に依存します。ここでは、最初に入力されたファイルのファイル名を **file1**、2番目に入力されたファイルのファイル名を **file2** と仮定します。生成される出力リストの各エントリの前には、見つかった差分の種類を示す固有のテキスト記号が付与されます。テキスト記号とその意味は次のとおりです。

- マイナス記号 (-) は、設定行が **file1** に存在するが **file2** には存在しないことを示します。
- プラス記号 (+) は、設定行が **file2** に存在するが **file1** には存在しないことを示します。
- 感嘆符 (!) と説明用のコメントは、順序による影響を受ける設定行の場所が、**file1** と **file2** で異なることを示します。

### 差分比較操作

アプリケーションによっては、**diff** 操作で生成される出力に、変更されていない（つまり、マイナス記号やプラス記号のない）コンフィギュレーション行を含める必要があります。これらのアプリケーションでは、指定されたコンフィギュレーションファイルを実行コンフィギュレーションファイルと比較する **show archive config incremental-diffs** コマンドを使用して、増分の **diff** 操作を実行できます。

増分 **diff** 操作が実行されると、実行コンフィギュレーションファイルに出現しない設定行（つまり、実行コンフィギュレーションと比較して指定されたファイルにのみ出現する設定行）のリストが出力として生成されます。感嘆符 (!) と説明用のコメントは、順序による影響を受ける設定行の場所が、指定されたコンフィギュレーションファイルと実行コンフィギュレーションファイルで異なることを示します。

# コンフィギュレーションのコンテキスト差分ユーティリティの使い方

## コンフィギュレーションのコンテキスト差分ユーティリティを使用した行ごとのファイル比較の実行

### 手順の概要

1. **enable**
2. 次のいずれか1つを入力します。
  - **show archive config differences** *[file1 [file2]]*
  - **show archive config incremental-diffs** *file*
3. **exit**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>
ステップ 2	<p>次のいずれか1つを入力します。</p> <ul style="list-style-type: none"> <li>• <b>show archive config differences</b> <i>[file1 [file2]]</i></li> <li>• <b>show archive config incremental-diffs</b> <i>file</i></li> </ul> <p>例 :</p> <pre>Device# show archive config differences running-config startup-config</pre> <p>例 :</p> <pre>Device# show archive config incremental-diffs nvram:startup-config</pre>	<p>2つのコンフィギュレーションファイル (Cisco IOS File System を通じてアクセス可能) を行ごとに比較し、その間の差分の一覧を生成します。</p> <p>または</p> <p>実行コンフィギュレーションファイルに対して、指定されたコンフィギュレーションファイルの行ごとの比較を実行し、実行コンフィギュレーションファイルに出現しないコンフィギュレーション行のリストを生成します。</p>
ステップ 3	<p><b>exit</b></p> <p>例 :</p> <pre>Device# exit</pre>	<p>ユーザ EXEC モードに戻ります。</p>

# コンフィギュレーションのコンテキスト差分ユーティリティの設定例

## 差分操作の例

この例では、実行コンフィギュレーションファイルとスタートアップコンフィギュレーションに対して比較操作を行います。次の表は、この例で使用しているコンフィギュレーションファイルを示しています。

表 4: diff 操作で使用するコンフィギュレーションファイルの例

実行コンフィギュレーションファイル	スタートアップコンフィギュレーションファイル
no ip subnet-zero	ip subnet-zero
ip cef	ip cef
interface FastEthernet1/0	ip name-server 10.4.4.4
ip address 10.7.7.7 255.0.0.0	voice dnis-map 1
no ip route-cache	dnis 111
no ip mroute-cache	interface FastEthernet1/0
duplex half	no ip address
no ip classless	no ip route-cache
snmp-server community public RO	no ip mroute-cache
	shutdown
	duplex half
	ip default-gateway 10.5.5.5
	ip classless
	access-list 110 deny ip any host 10.1.1.1
	access-list 110 deny ip any host 10.1.1.2
	access-list 110 deny ip any host 10.1.1.3
	snmp-server community private RW

次に、**show archive config differences** コマンドの出力例を示します。この出力例は、以下の表のコンフィギュレーションファイルに対して実行された **diff** 操作の結果を示しています。

```
Device# show archive config differences system:running-config nvram:startup-config

+ip subnet-zero

+ip name-server 10.4.4.4

+voice dnis-map 1
```

```

+dnis 111

interface FastEthernet1/0

+no ip address

+shutdown

+ip default-gateway 10.5.5.5

+ip classless

+access-list 110 deny ip any host 10.1.1.1

+access-list 110 deny ip any host 10.1.1.2

+access-list 110 deny ip any host 10.1.1.3

+snmp-server community private RW

-no ip subnet-zero

interface FastEthernet1/0

-ip address 10.7.7.7 255.0.0.0

-no ip classless

-snmpp-server community public RO

```

## 差分比較操作の例

この例では、スタートアップ コンフィギュレーション ファイルと実行コンフィギュレーション ファイルに対して増分 **diff** 操作を行います。次の表は、この例で使用しているコンフィギュレーション ファイルを示しています。

表 5: 差分比較操作の例で使用するコンフィギュレーションファイル

スタートアップコンフィギュレーションファイル	実行コンフィギュレーションファイル
ip subnet-zero	no ip subnet-zero
ip cef	ip cef
ip name-server 10.4.4.4	interface FastEthernet1/0
voice dnis-map 1	ip address 10.7.7.7 255.0.0.0
dnis 111	no ip route-cache
interface FastEthernet1/0	no ip mroute-cache
no ip address	duplex half
no ip route-cache	no ip classless
no ip mroute-cache	snmp-server community public RO
shutdown	
duplex half	
ip default-gateway 10.5.5.5	
ip classless	
access-list 110 deny ip any host 10.1.1.1	
access-list 110 deny ip any host 10.1.1.2	
access-list 110 deny ip any host 10.1.1.3	
snmp-server community private RW	

次に、**show archive config incremental-diffs** コマンドの出力例を示します。この出力例は、以下の表のコンフィギュレーションファイルに対して実行された増分 diff 操作の結果を示しています。

```
Device# show archive config incremental-diffs startup-config

ip subnet-zero

ip name-server 10.4.4.4
```

```

voice dnis-map 1

dnis 111

interface FastEthernet1/0

no ip address

shutdown

ip default-gateway 10.5.5.5

ip classless

access-list 110 deny ip any host 10.1.1.1

access-list 110 deny ip any host 10.1.1.2

access-list 110 deny ip any host 10.1.1.3

snmp-server community private RW
    
```

## その他の参考資料

次の項に、コンフィギュレーションパーティショニング機能に関する参考資料を示します。

### 関連資料

関連項目	マニュアルタイトル
実行コンフィギュレーションのパフォーマンス強化：インターフェイスの <b>parserconfigcache</b>	コンフィギュレーション生成のパフォーマンス拡張
カスタマーサービスのプロビジョニング、コンフィギュレーションロールバック、コンフィギュレーションロック、およびコンフィギュレーションアクセスコントロール	コンフィギュレーションのコンテキスト差分ユーティリティ
コンフィギュレーション管理：コンフィギュレーション変更およびロギング	コンフィギュレーション変更通知およびロギング
コンフィギュレーション管理：コンフィギュレーション変更およびロギングのクイック保存： <a href="#">2</a> 。	コンフィギュレーション ロガー永続性
Cisco IOS ソフトウェア コンフィギュレーション アクセス制御およびコンフィギュレーションセッションロック（「Config ロック」）。	排他的設定変更アクセスとアクセスセッションロック

- <sup>2</sup> 「コンフィギュレーション ロガー永続性」機能により、スタートアップ コンフィギュレーション全体を保存するのではなく、最後の startup-config ファイルが生成された時点から入力されたコマンドだけを保存できます。

**標準**

標準	タイトル
この機能に関連付けられている規格はありません。	--

**MIB**

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	--

**RFC**

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

**テクニカル サポート**

説明	Link
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。</p>	<p><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></p>



# コンフィギュレーションのコンテキスト差分ユーティリティの機能情報

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

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

表 6: コンフィギュレーションのコンテキスト差分ユーティリティの機能情報

機能名	リリース	機能情報
コンフィギュレーションのコンテキスト差分ユーティリティ	Cisco IOS XE リリース 2.1	<p>コンフィギュレーションのコンテキスト差分ユーティリティ機能は、2つのコンフィギュレーションファイルを行ごとに比較し、その間の違いの一覧を生成する機能を提供します。生成される出力には、追加、変更、または削除されたコンフィギュレーション行に関する情報と、変更されたコンフィギュレーション行が存在するコンフィギュレーション モードが含まれます。</p> <p>この機能は、Cisco IOS XE Release 2.1 で、Cisco ASR 1000 シリーズ ルータに導入されました。</p> <p>この機能により、次のコマンドが変更されました。<b>show archive config differences</b>、<b>show archive config incremental-diffs</b></p>





## 第 7 章

# コンフィギュレーション変更通知およびロギング

コンフィギュレーション変更通知およびロギング（コンフィギュレーションログアーカイブ）機能を使用すると、アーカイブ機能を実装することにより、設定変更をセッションごとおよびユーザごとに追跡できます。このアーカイブでは、適用された各コンフィギュレーションコマンド、コマンドを適用した人、コマンドの Parser Return Code（PRC）、コマンドを適用した時刻を追跡する「設定ログ」が保存されます。また、この機能により、設定ログが変化したときに非同期通知を登録されたアプリケーションに送信する、通知メカニズムも追加されます。

コンフィギュレーション変更通知およびロギング機能が導入されるまでは、シスコソフトウェアの設定が変更されたかどうかを判断するための唯一の方法は、実行コンフィギュレーションとスタートアップコンフィギュレーションのコピーをローカルコンピュータに保存し、行単位で比較することでした。この比較方法では、変更を特定できますが、変更が行われた順序や、変更に責任を持つ人は特定できません。

- [機能情報の確認（81 ページ）](#)
- [コンフィギュレーション変更通知およびロギングの制約事項（82 ページ）](#)
- [コンフィギュレーション変更通知およびロギングについて（82 ページ）](#)
- [コンフィギュレーション変更通知およびロギングの設定方法（84 ページ）](#)
- [コンフィギュレーション変更通知およびロギングの設定例（90 ページ）](#)
- [その他の参考資料（90 ページ）](#)
- [コンフィギュレーション変更通知およびロギングの機能情報（91 ページ）](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリースノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーション変更通知およびロギングの制約事項

- コンフィギュレーション モードでの完全なコマンド入力のみがログに記録されます。
- **copy** コマンドを使用して適用されたコンフィギュレーション ファイルの一部であるコマンドは、ログに記録されません。

## コンフィギュレーション変更通知およびロギングについて

### 設定ログ

コンフィギュレーション変更通知およびロギング機能は、設定ログを保持することで、シスコ ソフトウェアの実行コンフィギュレーションに加えられた変更を追跡します。この設定ログは、CLI または HTTP のみを介して開始される変更を追跡します。アクションルーチンの呼び出しが発生する完全なコマンドが記録されます。次の種類の入力はログに記録されません。

- 結果的に構文エラー メッセージが表示されるコマンド
- デバイス ヘルプ システムを呼び出す一部のコマンド

実行される各設定コマンドでは次の情報が記録されます。

- 実行されたコマンド
- コマンドが実行されたコンフィギュレーション モード
- コマンドを実行したユーザーの名前
- コマンドが実行された時間
- 設定変更のシーケンス番号
- コマンドへのパーサー返還コード

設定ログの情報を表示するには、**show archive log config** コマンドを使用します。ただし、Parser Return Code は、シスコ アプリケーションの内部だけで使用されるため、除外されます。

## コンフィギュレーション変更通知およびコンフィギュレーション変更ロギング

設定変更の通知をソフトウェアシステムロギング (syslog) プロセスに送信するように、コンフィギュレーション変更通知およびロギング機能を設定できます。syslog 通知機能を使用すると、ポーリングや情報収集作業を実行しなくても、設定ログ情報をモニタリングできます。

コンフィギュレーション変更通知およびロギング機能では、セッションごとまたはユーザごとにユーザが入力した設定変更を追跡できます。管理者はこのツールを使用して、ソフトウェアの実行コンフィギュレーションに加えられた設定変更をすべて追跡し、その変更を実行したユーザを特定できます。

### EAL4+ 認証用のコンフィギュレーション ロガーの機能強化

Evaluation Assurance Level 4+ (EAL4+) 認定のためのコンフィギュレーション ロガー機能拡張により、ロギングプロセスが Conformance to Common Criteria, EAL4+ Firewall Protection Profiles で規定されている要件を満たすことが保証されます。これらの機能拡張には、次の要件を満たすための変更が含まれています。

- ロギングパラメータを変更すると、それらの変更がログに記録されます。これは、実行コンフィギュレーションに対する各変更に対し、コピー操作 (**copy source running-config** など) から syslog メッセージを送信することで実現されます。
- 管理ユーザグループに対する変更がログに記録されます。たとえば、特権 EXEC モード (「イネーブル」モード) へのアクセスの失敗が記録されます。



(注) EALの認定はシスコが要求するものではありません。これらの機能拡張は、将来の認定に備えた土台となるものです。

前述のロギングアクションは、デフォルトでは無効になっています。これらのロギング特性を有効にするには、「コンフィギュレーション変更通知およびロギング」機能モジュールの「コンフィギュレーション変更通知およびロギング機能の設定」セクションに記載されているタスクを実行します。

# コンフィギュレーション変更通知およびロギングの設定方法

## コンフィギュレーション変更通知およびロギングの設定

### 手順の概要

1. **enable**
2. **configure terminal**
3. **archive**
4. **log config**
5. **logging enable**
6. **logging size *entries***
7. **hidekeys**
8. **notify syslog**
9. **end**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>archive</b> 例： Device(config)# archive	アーカイブ コンフィギュレーション モードを開始します。
ステップ 4	<b>log config</b> 例： Device(config-archive)# log config	設定変更ロガー コンフィギュレーション モードを開始します。
ステップ 5	<b>logging enable</b> 例：	設定変更のロギングをイネーブルにします。

	コマンドまたはアクション	目的
	Device(config-archive-log-config)# logging enable	<ul style="list-style-type: none"> <li>• コンフィギュレーション変更のロギングは、デフォルトでは無効になっています。</li> </ul>
ステップ 6	<b>logging size entries</b> 例： Device(config-archive-log-config)# logging size 200	(任意) 設定ログに保持する最大エントリ数を指定します。 <ul style="list-style-type: none"> <li>• <i>entries</i> 引数の有効な値の範囲は、1 ~ 1000 です。デフォルト値は 100 エントリです。</li> <li>• 設定ログがいっぱいになると、新しいエントリが追加されるたびに最も古いエントリが削除されます。</li> </ul> (注) 現在のログ サイズよりも小さいログ サイズが新たに指定された場合、ログ エントリの経過時間にかかわらず、新しいログ サイズになるまで最も古いログ エントリがすぐに削除されます。
ステップ 7	<b>hidekeys</b> 例： Device(config-archive-log-config)# hidekeys	(任意) パスワード情報が設定ログファイルに表示されないようにします。 (注) <b>hidekeys</b> コマンドを有効にすると、設定ログファイルにパスワード情報が表示されなくなり、セキュリティが向上します。
ステップ 8	<b>notify syslog</b> 例： Device(config-archive-log-config)# notify syslog	(任意) 設定変更の通知をリモート syslog に送信できるようにします。
ステップ 9	<b>end</b> 例： Device(config-archive-log-config)# end	特権 EXEC モードに戻ります。

## 設定ログ エントリおよび統計の表示

設定ログのエントリまたは設定ログのメモリ使用量に関する統計情報を表示するには、ここに示す作業を実行します。コマンドは任意の順序で入力できます。

設定ログ エントリを表示し、設定ログのメモリ使用量を監視するために、コンフィギュレーション変更通知およびロギング機能に **show archive log config** コマンドが用意されています。

## 手順の概要

1. **enable**
2. **show archive log config number [end-number]**
3. **show archive log config all provisioning**
4. **show archive log config statistics**
5. **exit**

## 手順の詳細

### ステップ1 enable

このコマンドを使用して、特権 EXEC モードをイネーブルにします。パスワードを入力します（要求された場合）。次に例を示します。

例：

```
Device> enable
```

### ステップ2 show archive log config number [end-number]

このコマンドを使用して、設定ログ エントリをレコード番号ごとに表示します。オプションの *end-number* を指定すると、*number* 引数で入力した値から *end-number* 引数で入力した値までの範囲のレコード番号を持つすべてのログ エントリが表示されます。次に例を示します。

```
Device# show archive log config 1 2

idx  sess  user@line      Logged command
  1    1    user1@console  logging enable
  2    1    user1@console  logging size 200
```

例：

この例では、設定ログ エントリ番号 1 と 2 が表示されています。*number* 引数と *end-number* 引数の範囲は 1 ~ 2147483647 です。

### ステップ3 show archive log config all provisioning

すべての設定ログ ファイルを、表形式ではなくコンフィギュレーションファイルでの表示形式で表示するには、このコマンドを使用します。次に例を示します。

例：

```
Device# show archive log config all provisioning

archive
log config
logging enable
logging size 200
```

この表示では、ログに記録されたコマンドを正しく適用するために必要な、コンフィギュレーション モードを変更するために使用したコマンドも表示されています。

### ステップ4 show archive log config statistics



コンフィギュレーションのメモリ使用量の情報を表示するには、このコマンドを使用します。次に例を示します。

例：

```
Device# show archive log config statistics

Config Log Session Info:
  Number of sessions being tracked: 1
  Memory being held: 3910 bytes
  Total memory allocated for session tracking: 3910 bytes
  Total memory freed from session tracking: 0 bytes
Config Log log-queue Info:
  Number of entries in the log-queue: 3
  Memory being held in the log-queue: 671 bytes
  Total memory allocated for log entries: 671 bytes
  Total memory freed from log entries:: 0 bytes
```

### ステップ5 exit

このコマンドを使用して、ユーザ EXEC モードに戻ります。次に例を示します。

例：

```
Device# exit
Device>
```

## 設定ログ エントリのクリア

設定ログのエントリは、2つのうちいずれかの方法でクリアできます。**logging size** コマンドを使用して設定ログのサイズを縮小するか、または **logging enable** コマンドを使用して設定ログを無効にしてから再び有効にすることができます。

### ログサイズのリセットによる設定ログの消去

このタスクでは、**logging size** コマンドを2回入力して、ログサイズを1に減らしてから、ログサイズを目的の値にリセットする方法を示します。

#### 手順の概要

1. **enable**
2. **configure terminal**
3. **archive**
4. **log config**
5. **logging size entries**
6. **logging size entries**
7. **end**

設定ログをディセーブルすることによる設定ログのクリア

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>archive</b> 例： Device(config)# archive	アーカイブ コンフィギュレーション モードを開始します。
ステップ 4	<b>log config</b> 例： Device(config-archive)# log config	設定変更ロガー コンフィギュレーション モードを開始します。
ステップ 5	<b>logging size entries</b> 例： Device(config-archive-log-config)# logging size 1	設定ログに保持する最大エン트리数を指定します。 (注) 設定ログのサイズを1に設定すると、最新のエン트리以外はすべて消去されます。
ステップ 6	<b>logging size entries</b> 例： Device(config-archive-log-config)# logging size 200	設定ログに保持する最大エン트리数を指定します。 (注) 設定ログを消去した後、設定ログのサイズを目的の値にリセットする必要があります。
ステップ 7	<b>end</b> 例： Device(config-archive-log-config)# end	特権 EXEC モードに戻ります。

設定ログをディセーブルすることによる設定ログのクリア

手順の概要

1. enable
2. configure terminal
3. archive

4. **log config**
5. **no logging enable**
6. **logging enable**
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Device> enable	特権 EXEC モードをイネーブルにします。  • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Device# configure terminal	グローバル コンフィギュレーション モードを開始します。
ステップ 3	<b>archive</b> 例： Device(config)# archive	アーカイブ コンフィギュレーション モードを開始します。
ステップ 4	<b>log config</b> 例： Device(config-archive)# log config	設定変更ロガー コンフィギュレーション モードを開始します。
ステップ 5	<b>no logging enable</b> 例： Device(config-archive-log-config)# no logging enable	コンフィギュレーション変更のロギングを無効にします。  (注) 設定ログを無効にすると、すべてのレコードが消去されます。
ステップ 6	<b>logging enable</b> 例： Device(config-archive-log-config)# logging enable	設定変更のロギングをイネーブルにします。
ステップ 7	<b>end</b> 例： Device(config-archive-log-config)# end	特権 EXEC モードに戻ります。

# コンフィギュレーション変更通知およびロギングの設定例

## 例：コンフィギュレーション変更通知およびロギングの設定

次に、設定ログの最大エントリ数を 200 にして設定ロギングをイネーブルにする例を示します。この例では、**hidekeys** コマンドを使用して設定ログレコード内のパスワード情報の表示を抑止することでセキュリティを向上させ、**notify syslog** コマンドで **syslog** 通知を有効にしています。

```
configure terminal
archive
  log config
  logging enable
  logging size 200
  hidekeys
  notify syslog
```

## その他の参考資料

### 関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『 <a href="#">Cisco IOS Master Command List, All Releases</a> 』
コンフィギュレーション ファイルの管理についての情報	『コンフィギュレーションファイルの管理コンフィギュレーションガイド』の「コンフィギュレーションファイルの管理」モジュール
コンフィギュレーション ファイルを管理するためのコマンド	『 <a href="#">Cisco IOS Configuration Fundamentals Command Reference</a> 』

シスコのテクニカル サポート

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

## コンフィギュレーション変更通知およびロギングの機能情報

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

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

表 7: コンフィギュレーション変更通知およびロギングの機能情報

機能名	リリース	機能情報
コンフィギュレーション変更通知およびロギング		<p>コンフィギュレーション変更通知およびロギング（コンフィギュレーション ロギング）機能を使用すると、設定ログを実装することで、セッションごとまたはユーザごとに設定変更を追跡できます。設定ログには、適用された各コンフィギュレーション コマンド、コマンドを適用した人、コマンドの Parser Return Code (PRC)、および、コマンドを適用した時刻が記録されます。また、この機能により、設定ログが変化したときに非同期通知を登録されたアプリケーションに送信する、通知メカニズムも追加されます。</p> <p>次のコマンドが導入または変更されました。 <b>archive</b>、<b>hidekeys</b>、<b>log config</b>、<b>logging enable</b>、<b>logging size</b>、<b>notify syslog</b>、<b>show archive log config</b></p>





## 第 8 章

# コンフィギュレーションパーティショニング

コンフィギュレーションパーティショニング機能によって実行コンフィギュレーション状態をモジュール化（「パーティショニング」）して、Cisco IOS ソフトウェアで実行コンフィギュレーションに柔軟にアクセスできるようにします。

この機能が搭載された Cisco IOS ソフトウェア イメージではデフォルトでオンになっています。

デバイスのコンフィギュレーション状態は、**showrunning-config** コマンドがユーザによって実行されると動的に取得されます。コンフィギュレーションパーティショニング機能がイネーブルの場合、システムによってデバイスのコンフィギュレーション状態が分割され、グループ化されます（「パーティション」と呼ばれます）。これにより、実行コンフィギュレーションで表示されるコマンドリストの生成時にユーザが確認したいコンフィギュレーション状態のみを取得できます。この機能により、システムのコンフィギュレーション状態全体が処理される従来の処理方法とは異なり、実行コンフィギュレーションコマンドのリストの生成時に実行コンフィギュレーション状態の一部のみが処理されるため、コンフィギュレーションが複雑なハイエンドシステムのパフォーマンスを向上できます。

デフォルトのコンフィギュレーションパーティションはこの機能を導入することで提供されません。将来のリリースでは、他の Cisco IOS ソフトウェア機能によって独自のコマンドパーティションが提供される可能性があります。

- [機能情報の確認](#) (94 ページ)
- [コンフィギュレーションパーティショニングについて](#) (94 ページ)
- [コンフィギュレーションパーティショニング機能を使用するには](#) (95 ページ)
- [コンフィギュレーションパーティショニングするためのコンフィギュレーション例](#) (99 ページ)
- [その他の参考資料](#) (108 ページ)
- [コンフィギュレーションパーティショニングの機能情報](#) (110 ページ)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーションパーティショニングについて

### システム実行コンフィギュレーション

Cisco IOS ソフトウェアベース デバイスのコンフィギュレーション管理には、不揮発性メモリに格納されたスタートアップコンフィギュレーション (`startup-config`) およびシステムに適用されているすべてのコンフィギュレーション オプションである実行コンフィギュレーション (`running-config`) を管理する必要があります。通常、スタートアップコンフィギュレーション ファイルはシステム起動時にロードされ、コマンドラインインターフェイス (CLI) を使用して適用されたシステムに対する実行コンフィギュレーションの変更は、実行コンフィギュレーションをコンフィギュレーションファイルにコピーすることで保存されます (ローカルまたはネットワーク上)。ファイルは、起動時にデバイスをコンフィギュレーションする場合、または他のデバイスをコンフィギュレーションする場合に使用されます。

### 実行コンフィギュレーションを取得して表示またはコピーする

Cisco IOS のソフトウェア コンフィギュレーション モデルでは、コンフィギュレーション状態は分散して維持され、各コンポーネントは独自のコンフィギュレーション状態を保持します。グローバルコンフィギュレーション情報を取得するには、ソフトウェアは各コンポーネントをポーリングして、分散された情報を収集する必要があります。このコンフィギュレーション状態の取得処理は不揮発性生成 (NVGEN) として知られる処理によって実行され、現在のコンフィギュレーション状態を表示する `showrunning-config` などのコマンドや、実行コンフィギュレーションをファイルにコピーして保存する `copysystem:running-configuration` コマンドによって呼び出されます。取得処理が呼び出されると、NVGEN 処理によって各システムコンポーネント、各インターフェイスインスタンス、およびその他すべてのコンフィギュレーションされたコンポーネント オブジェクトが標準の順序でクエリーされます。NVGEN がこれらのクエリーを実行しているシステムを通過するときに、実行コンフィギュレーションファイルが作成されます。表示およびコピーには作成された「仮想ファイル」が使用されます。



## 実行コンフィギュレーションをパーティショニングする利点

コンフィギュレーションパーティショニング機能は、Cisco IOS ソフトウェアに追加された一連のコンフィギュレーション生成のパフォーマンス拡張機能の最新機能です（関連する機能については、「関連ドキュメント」セクションを参照してください）。この機能によって、**showrunning-config** コマンドの実行時に表示したいシステムコンポーネントのみがクエリーされるため、システム応答時間が短縮されます。

コンフィギュレーションパーティショニング機能がイネーブルの場合、システムによってデバイスのコンフィギュレーション状態が分割され、グループ化されます（「パーティション」と呼ばれます）。これにより、仮想実行コンフィギュレーションファイル（コンフィギュレーションコマンドのリスト）が生成されます。新しいコマンド **showrunning-configpartition** を使用すると、一度に実行コンフィギュレーションをすべて表示したり、特定のストリングに一致する行のみを表示するのではなく、検証したい実行コンフィギュレーションの部分のみを表示することができます。

この機能は、ユーザが表示したいシステムコンポーネントのグループ（特定のインターフェイスなど）のみの NVGEN 処理をシステムで実行してシステムのパフォーマンスを向上できることが主な利点であると言えます。この特徴は、システムコンポーネントをすべて処理した後に生成されたリストをフィルタ処理するだけの **showrunning-config** コマンドのその他の拡張とは対照的です。

実行コンフィギュレーションを部分的に生成するため、システムのコンフィギュレーション状態を選択的に処理することを「コンフィギュレーションパーティショニング」と呼びます。

コンフィギュレーション情報に柔軟にアクセスできることで、サイズの大きいコンフィギュレーションファイルがあるハイエンドなルーティングプラットフォームにパフォーマンスの重大な利点をもたらし、同時に詳細なコンフィギュレーション機能を細かに実装することでコンフィギュレーション管理を強化します。詳細なコンフィギュレーションオプションには、Cisco IOS ソフトウェアのカスタマーサービスのプロビジョニング、コンフィギュレーションロールバック、コンフィギュレーションロック、およびコンフィギュレーションアクセスコントロールのサポートが含まれます。

## コンフィギュレーションパーティショニング機能を使用するには

### コンフィギュレーションパーティションの表示

この機能を活用するには、主に特権 EXEC モードで **showrunning-configpartitionpart** コマンドを使用します。このコマンドは、**showrunning-config** コマンド専用の拡張です。



(注) **partitionpart** コマンドの拡張は、**more:systemrunning-config** コマンドでは利用できません。

この機能は既存のコマンドのパフォーマンスを向上するので、この機能が搭載された Cisco IOS ソフトウェアイメージではデフォルトでオンになっています。お使いのシステムでサポートおよび実行されているかどうかを簡単に判断するには、特権 EXEC モードで **showrunning-configpartition?** コマンドを実行します。

## 手順の概要

1. **show running-config partition ?**
2. **show running-config partition part**

## 手順の詳細

### ステップ 1 show running-config partition ?

このコマンドを実行すると、システムに表示できる実行コンフィギュレーションの部分が表示されます。コンフィギュレーションパーティショニング機能がシステムでサポートされており、イネーブルの場合は、ヘルプ出力の 1 行目に「**config partition is TRUE**」というストリングが表示されます。

ここに示すコマンド構文を入力するとエラーメッセージが表示される場合は、この機能はシステムでサポートされていません。実行コンフィギュレーションの部分のみを表示できる他のリリースで利用可能な **showrunning-config** コマンドの既存の拡張については、コマンドのマニュアルを参照してください。

(注) 利用できるコンフィギュレーションの部分は、ソフトウェアイメージによって異なり、コンフィギュレーションされている機能に依存します。

例 :

```
Router# show running-config partition ?
config partition is TRUE
access-list      All access-list configurations
boot             All boot configurations
class-map        All class-map configurations
common           All remaining unregistered configurations
global-cdp       All global cdp configurations
interface        All Interface specific Configurations
ip-as-path       All IP as-path configurations
ip-community     All IP community list configurations
ip-domain-list  All ip domain list configurations
ip-prefix-list  All ip prefix-list configurations
ip-static-routes All IP static configurations
line             All line mode configurations
policy-map       All policy-map configurations
route-map        All route-map configurations
router           All routing configurations
snmp             All SNMP configurations
tacacs           All TACACS configurations
```

表示する実行コンフィギュレーションの部分を選択して、ステップ 2 で関連キーワードを **part** 引数として使用します。

### ステップ 2 show running-config partition part

たとえば、システムで NVGEN 処理を実行コンフィギュレーション状態の `access-list` 部分に関連するコンポーネントのみで実行して、`access-list` に関連するコンポーネントのみを表示する場合は、**showrunning-configpartitionaccess-list** コマンドを入力します。

例：

```
Router# show running-config partition access-list
Building configuration...
Current configuration : 127 bytes
!
Configuration of Partition access-list
!
!
!
access-list 90 permit 0.0.0.0 1.2.3.5
access-list 100 permit 10 any any
!
end
```

(注) このコマンドを使用すると、NVGEN 処理を実行して、特定のインターフェイスに関する結果出力を表示します。複数のインターフェイスがアクティブなシステムで使用できる設計のこの動作がコンフィギュレーションパーティショニング機能の主な役割です。

次の例では、メインのコンフィギュレーションパーティションはインターフェイスコンフィギュレーションです。生成される特定のコンフィギュレーション部分は、ファストイーサネットインターフェイス 0/0 のコンフィギュレーションです。

例：

```
Router# show running-config partition interface fastethernet0/0
Building configuration...
Current configuration : 213 bytes
!
Configuration of Partition interface FastEthernet0/0
!
!
!
interface FastEthernet0/0
ip address 10.4.2.39 255.255.255.0
no ip route-cache cef
no ip route-cache
duplex half
ipv6 enable
no cdp enable
!
!
end
```

## コンフィギュレーションパーティショニング機能をディセーブルにする

この機能は既存のコマンドのパフォーマンスを向上させるので、この機能が搭載された Cisco IOS ソフトウェアイメージではデフォルトでオンになっています。しかし、この機能は少量のシステムリソース（メモリおよび CPU）を消費するため不要な場合、ディセーブルにしたい

次の作業

場合があります。コンフィギュレーションパーティショニングをディセーブルにするには、次の手順を実行してください。手順はユーザ EXEC モードで起動されていることを前提としています。

手順の概要

1. **enable**
2. **configure terminal**
3. **no parser config partition**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Router# 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	<b>no parser config partition</b> 例： Router(config)# no parser config partition 例： Disabling config partitioning 例： Router(config)#	コンフィギュレーションパーティショニング機能をディセーブルにします。

次のタスク

次の作業

機能をディセーブルにした後、イネーブルにするには、グローバル コンフィギュレーション モードで **parserconfigpartition** コマンドを使用します。



(注) この機能はデフォルトでイネーブルになっているので、実行コンフィギュレーションファイルには、**no** 形式のみが表示されます。または、**copyrunning-configstartup-config** コマンドを実行するとスタートアップ コンフィギュレーション ファイルに書き込まれます。

# コンフィギュレーションパーティショニングするための コンフィギュレーション例

ここでは、**show running-config partition** コマンドを使用してコンフィギュレーションパーティションを表示する例を示します。

## コンフィギュレーションパーティションの表示例

この例では、管理者が特定のインターフェイスの状態、およびシステムの他のコンポーネントの一部のコンフィギュレーションを確認するために実行する一連の手順で

**showrunning-configpartition** と関連コマンドと一緒に使用しています。標準の **showrunning-config** コマンド (例: **showrunning-config|includeaccess-list**) による、同等のフィルタされた出力もデモとして含まれます。



(注) *part* 引数には **showrunning-configpartrouterreigrp1** のように複数のパーティション名キーワードを含めることができます。

```
gt3-7200-3# show running-config partition ?
access-list      All access-list configurations
boot             All boot configurations
class-map        All class-map configurations
global-cdp       All global cdp configurations
interface        All Interface specific Configurations
ip-as-path       All IP as-path configurations
ip-community     All IP community list configurations
ip-domain-list   All ip domain list configurations
ip-static-routes All IP static configurations
line             All line mode configurations
policy-map       All policy-map configurations
route-map        All route-map configurations
router           All routing configurations
service          All service configurations
snmp             All SNMP configurations
gt3-7200-3# show running-config partition access-list

Building configuration...
Current configuration : 87 bytes
!
!
!
!
access-list 90 permit 0.0.0.0 1.2.3.5
access-list 100 permit 10 any any
!
end
gt3-7200-3# show running-config | include access-list

access-list 90 permit 0.0.0.0 1.2.3.5
access-list 100 permit 10 any any
gt3-7200-3#
```

```

gt3-7200-3# show running-config partition boot

Building configuration...
Current configuration : 51 bytes
!
boot network tftp://service_config.txt
!
!
end
gt3-7200-3# show running-config partition class-map

Building configuration...
Current configuration : 78 bytes
!
!
!
class-map match-all abc
  match any
class-map match-all xyz
!
!
!
end
gt3-7200-3# show running-config | begin class-map

class-map match-all abc
  match any
class-map match-all xyz
!
!
gt3-7200-3# show running-config partition global-cdp

Building configuration...
Current configuration : 43 bytes
!
!
!
cdp timer 20
cdp holdtime 100
!
end
gt3-7200-3# show running-config | include

global-cdp

cdp timer 20
cdp holdtime 100
gt3-7200-3#
gt3-7200-3# show ip interface brief
Interface                IP-Address      OK? Method Status          Protocol
FastEthernet0/0          unassigned     YES NVRAM  administratively down  down
Ethernet2/0              10.4.2.32      YES NVRAM  up              up
Ethernet2/1              unassigned     YES NVRAM  administratively down  down
Ethernet2/2              unassigned     YES NVRAM  administratively down  down
Ethernet2/3              unassigned     YES NVRAM  administratively down  down
Serial3/0                 unassigned     YES NVRAM  administratively down  down
Serial3/1                 unassigned     YES NVRAM  administratively down  down
Serial3/2                 unassigned     YES NVRAM  administratively down  down
Serial3/3                 unassigned     YES NVRAM  administratively down  down
Loopback0                 unassigned     YES NVRAM  administratively down  down
Loopback234              unassigned     YES NVRAM  administratively down  down
gt3-7200-3# show running-config partition interface fastethernet0/0
Building configuration...

```

```

Current configuration : 98 bytes
!
!
!
interface FastEthernet0/0
  no ip address
  no ip route-cache
  shutdown
  duplex half
!
!
end
gt3-7200-3# show running-config partition interface ethernet2/0

Building configuration...
Current configuration : 122 bytes
!
!
!
interface Ethernet2/0
  ip address 10.4.2.32 255.255.255.0
  no ip proxy-arp
  no ip route-cache
  duplex half
!
!
end
gt3-7200-3# show running-config partition interface ethernet2/1
Building configuration...
Current configuration : 94 bytes
!
!
!
interface Ethernet2/1
  no ip address
  no ip route-cache
  shutdown
  duplex half
!
!
end
gt3-7200-3# show running-config partition interface ethernet2/2

Building configuration...
Current configuration : 94 bytes
!
!
!
interface Ethernet2/2
  no ip address
  no ip route-cache
  shutdown
  duplex half
!
!
end
gt3-7200-3# show running-config partition interface ethernet2/3
Building configuration...
Current configuration : 94 bytes
!
!
!
interface Ethernet2/3
  no ip address

```

```

no ip route-cache
shutdown
duplex half
!
!
end
gt3-7200-3# show running-config partition interface serial3/0
Building configuration...
Current configuration : 103 bytes
!
!
!
interface Serial3/0
no ip address
no ip route-cache
shutdown
serial restart-delay 0
!
!
end
gt3-7200-3# show running-config partition interface serial3/1
Building configuration...
Current configuration : 103 bytes
!
!
!
interface Serial3/1
no ip address
no ip route-cache
shutdown
serial restart-delay 0
!
!
end
gt3-7200-3# show running-config partition interface serial3/2
Building configuration...
Current configuration : 103 bytes
!
!
!
interface Serial3/2
no ip address
no ip route-cache
shutdown
serial restart-delay 0
!
!
end
gt3-7200-3# show running-config partition interface serial3/3
Building configuration...
Current configuration : 103 bytes
!
!
!
interface Serial3/3
no ip address
no ip route-cache
shutdown
serial restart-delay 0
!
!
end
gt3-7200-3# show running-config partition interface loopback0
Building configuration...

```



```

Current configuration : 79 bytes
!
!
interface Loopback0
  no ip address
  no ip route-cache
  shutdown
!
!
end
gt3-7200-3# show running-config partition interface loopback1
                                     ^
% Invalid input detected at '^' marker.
gt3-7200-3# show running-config partition interface loopback234
Building configuration...
Current configuration : 81 bytes
!
!
!
interface Loopback234
  no ip address
  no ip route-cache
  shutdown
!
!
end
gt3-7200-3# configure terminal

Enter configuration commands, one per line.  End with CNTL/Z.
gt3-7200-3(config)# interface ethernet 2/0.1
gt3-7200-3(config-subif)# exit
gt3-7200-3(config)# exit
gt3-7200-3#
00:13:05: %SYS-5-CONFIG_I: Configured from console by console
gt3-7200-3# show running-config partition interface ethernet2/0.1
Building configuration...
Current configuration : 58 bytes
!
!
!
interface Ethernet2/0.1
  no ip route-cache
!
!
end
gt3-7200-3# show run partition ip?
ip-as-path ip-community ip-domain-list ip-static-routes
gt3-7200-3#sh run part ip-as
gt3-7200-3#sh run part ip-as-path

Building configuration...
Current configuration : 125 bytes
!
!
!
ip as-path access-list 2 permit $ABC
ip as-path access-list 2 permit $xyz*
ip as-path access-list 2 permit qwe*
!
end
gt3-7200-3# show running-config partition ip-community

Building configuration...

```

```

Current configuration : 92 bytes
!
!
!
ip community-list standard asd permit
ip community-list expanded qwe deny uio*
!
end
gt3-7200-3# show running-config | include ip community
ip community-list standard asd permit
ip community-list expanded qwe deny uio*
gt3-7200-3#
gt3-7200-3# show running-config partition ip-domain-list

Building configuration...
Current configuration : 70 bytes
!
!
!
ip domain-list iop
ip domain-list tyu
ip domain-list jkl
!
!
!
end
gt3-7200-3# show running-config partition
ip-static-routes

Building configuration...
Current configuration : 98 bytes
!
!
!
ip route 0.0.0.0 0.0.0.0 Ethernet2/0
ip route 171.69.1.129 255.255.255.255 10.4.29.1
!
!
end
gt3-7200-3# show running-config partition line
Building configuration...
Current configuration : 489 bytes
!
!
!
!
line con 0
  exec-timeout 0 0
  transport output lat pad v120 mop telnet rlogin udptn nasi
  stopbits 1
line aux 0
  transport output lat pad v120 mop telnet rlogin udptn nasi
  stopbits 1
line vty 0
  password lab
  login
  transport input lat pad v120 mop telnet rlogin udptn nasi
  transport output lat pad v120 mop telnet rlogin udptn nasi
line vty 1 4
  login
  transport input lat pad v120 mop telnet rlogin udptn nasi
  transport output lat pad v120 mop telnet rlogin udptn nasi
!
end
gt3-7200-3# show running-config partition policy-map
Building configuration...
Current configuration : 162 bytes

```

```

!
!
!
policy-map qwer
  description policy-map qwer.
  class xyz
    shape peak 8000 32 32
policy-map p1
policy-map sdf
  class abc
  set precedence 4
!
!
!
end
gt3-7200-3# show running-config partition route-map
Building configuration...
Current configuration : 65 bytes
!
!
!
route-map iop permit 10
!
route-map rty permit 10
!
!
end
gt3-7200-3#sh run part router bgp 1
Building configuration...
Current configuration : 111 bytes
!
!
!
router bgp 1
  no synchronization
  bgp log-neighbor-changes
  distance bgp 2 2 2
  no auto-summary
!
!
end
gt3-7200-3#sh run part router egp ?
<0-65535> Remote autonomous system number
gt3-7200-3#sh run part router egp 1
Building configuration...
Current configuration : 46 bytes
!
!
!
router egp 1
  timers egp 20 20
!
!
end
gt3-7200-3# show running-config partition router ?
bgp      Border Gateway Protocol (BGP)
egp      Exterior Gateway Protocol (EGP)
eigrp    Enhanced Interior Gateway Routing Protocol (EIGRP)
isis     ISO IS-IS
iso-igrp IGRP for OSI networks
mobile   Mobile routes
odr      On Demand stub Routes
ospf     Open Shortest Path First (OSPF)
rip      Routing Information Protocol (RIP)

```

```

gt3-7200-3# show running-config partition router eigrp ?
<1-65535> Autonomous system number
gt3-7200-3# show running-config partition router eigrp 1
Building configuration...
Current configuration : 13 bytes
!
!
!
!
end
gt3-7200-3#
gt3-7200-3# sh run part router eigrp 2

Building configuration...
Current configuration : 57 bytes
!
!
!
router eigrp 2
 variance 10
 auto-summary
!
!
end
gt3-7200-3# show running-config partition router ?
  bgp      Border Gateway Protocol (BGP)
  egp      Exterior Gateway Protocol (EGP)
  eigrp    Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis     ISO IS-IS
  iso-igrp IGRP for OSI networks
  mobile   Mobile routes
  odr      On Demand stub Routes
  ospf     Open Shortest Path First (OSPF)
  rip      Routing Information Protocol (RIP)
gt3-7200-3# show running-config partition router isis ?
  WORD ISO routing area tag
  |    Output modifiers
  <cr>
gt3-7200-3# show running-config partition router isis qwe
Building configuration...
Current configuration : 86 bytes
!
!
!
router isis qwe
 set-attached-bit route-map qwer
 use external-metrics
!
!
end
gt3-7200-3# show running-config partition router isis ?
  WORD ISO routing area tag
  |    Output modifiers
  <cr>
gt3-7200-3# show running-config partition router iso
gt3-7200-3# show running-config partition router iso-igrp ?
  WORD ISO routing area tag
  |    Output modifiers
  <cr>
gt3-7200-3# show running-config partition router iso-igrp

Building configuration...
Current configuration : 31 bytes
!

```

```

!
!
router iso-igrp
!
!
end
gt3-7200-3# show running-config | begin iso
router iso-igrp
!
router isis qwe
  set-attached-bit route-map qwer
  use external-metrics
!
router egp 1
  timers egp 20 20
!
router bgp 1
  no synchronization
  bgp log-neighbor-changes
  distance bgp 2 2 2
  no auto-summary
!

gt3-7200-3# show running-config partition router ?
  bgp      Border Gateway Protocol (BGP)
  egp      Exterior Gateway Protocol (EGP)
  eigrp    Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis     ISO IS-IS
  iso-igrp IGRP for OSI networks
  mobile   Mobile routes
  odr      On Demand stub Routes
  ospf     Open Shortest Path First (OSPF)
  rip      Routing Information Protocol (RIP)
gt3-7200-3# show running-config partition router mobile ?
  | Output modifiers
  <cr>
gt3-7200-3# show running-config partition router mobile

Building configuration...
Current configuration : 42 bytes
!
!
!
router mobile
  distance 20
!
!
end
gt3-7200-3# sh run | include router

router mobile
router odr
router eigrp 2
router ospf 4
router iso-igrp
router isis qwe
router egp 1
router bgp 1
gt3-7200-3# show running-config partition router ?
  bgp      Border Gateway Protocol (BGP)
  egp      Exterior Gateway Protocol (EGP)
  eigrp    Enhanced Interior Gateway Routing Protocol (EIGRP)
  isis     ISO IS-IS
  iso-igrp IGRP for OSI networks

```

```

mobile      Mobile routes
odr         On Demand stub Routes
ospf       Open Shortest Path First (OSPF)
rip        Routing Information Protocol (RIP)
gt3-7200-3# show running-config partition router ospf ?
<1-65535>  Process ID
gt3-7200-3# show running-config partition router ospf 4
Building configuration...
Current configuration : 64 bytes
!
!
!
router ospf 4
  log-adjacency-changes
  distance 4
!
!
end
gt3-7200-3# sh run part service

Building configuration...
Current configuration : 190 bytes
!
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service internal
service udp-small-servers
service tcp-small-servers
!
!
!
end
gt3-7200-3# sh run part snmp

Building configuration...
Current configuration : 84 bytes
!
!
!
snmp-server community user101 RW
snmp mib target list qwe host 0.0.0.0
!
end

```

## その他の参考資料

次の項に、コンフィギュレーションパーティショニング機能に関する参考資料を示します。

### 関連資料

関連項目	マニュアルタイトル
実行コンフィギュレーションのパフォーマンス強化： インターフェイスの <b>parserconfigcache</b>	コンフィギュレーション生成のパフォーマンス拡張

関連項目	マニュアルタイトル
カスタマーサービスのプロビジョニング、コンフィギュレーションロールバック、コンフィギュレーションロック、およびコンフィギュレーションアクセスコントロール	コンフィギュレーションのコンテキスト差分ユーティリティ
コンフィギュレーション管理：コンフィギュレーション変更およびロギング	コンフィギュレーション変更通知およびロギング
コンフィギュレーション管理：コンフィギュレーション変更およびロギングのクイック保存： <sup>3</sup>	コンフィギュレーション ロガー永続性
Cisco IOS ソフトウェア コンフィギュレーション アクセス制御およびコンフィギュレーションセッションロック（「Config ロック」）。	排他的設定変更アクセスとアクセスセッションロック

<sup>3</sup> 「コンフィギュレーション ロガー永続性」機能により、スタートアップ コンフィギュレーション全体を保存するのではなく、最後の startup-config ファイルが生成された時点から入力されたコマンドだけを保存できます。

#### 標準

標準	タイトル
この機能に関連付けられている規格はありません。	--

#### MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	--

#### RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

テクニカル サポート

説明	Link
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンラインリソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。</p>	<p><a href="http://www.cisco.com/techsupport">http://www.cisco.com/techsupport</a></p>

## コンフィギュレーションパーティショニングの機能情報

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

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

表 8: コンフィギュレーションパーティショニングの機能情報

機能名	リリース	機能情報
<p>コンフィギュレーションパーティショニング</p>	<p>12.2(33)SRB 12.2(33)SB 12.2(33)SXI</p>	<p>コンフィギュレーションパーティショニング機能によって実行コンフィギュレーション状態をモジュール化（「パーティショニング」）して、Cisco IOS ソフトウェアで実行コンフィギュレーションに柔軟にアクセスできるようにします。この機能が搭載された Cisco IOS ソフトウェア イメージではデフォルトでオンになっています。</p> <p>12.2(33)SB では、この機能が Cisco 10000 シリーズに実装されました。</p> <p>この機能に関する詳細については、次の各項を参照してください。</p> <ul style="list-style-type: none"> <li>• コンフィギュレーションパーティショニングについて</li> <li>• コンフィギュレーションパーティショニング機能を使用するには</li> </ul>





## 第 9 章

# コンフィギュレーションのバージョン管理

コンフィギュレーションのバージョン管理機能により、シスコの実行コンフィギュレーションのコピーをデバイス上やデバイス外で維持および管理することができます。コンフィギュレーション置換機能では、実行コンフィギュレーションの保存されたコピーへのロールバックを行うためにコンフィギュレーションバージョン管理機能を使用します。

- 機能情報の確認 (111 ページ)
- コンフィギュレーションのバージョン管理について (111 ページ)
- コンフィギュレーションのバージョン管理の設定方法 (112 ページ)
- コンフィギュレーションのバージョン管理の設定例 (117 ページ)
- その他の参考資料 (117 ページ)
- コンフィギュレーションのバージョン管理の機能情報 (118 ページ)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェアリリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーションのバージョン管理について

### コンフィギュレーションアーカイブ

シスコのコンフィギュレーションアーカイブは、**configure replace** コマンドにより提供されるコンフィギュレーションのロールバック機能を強化するために、シスコのコンフィギュレー

ションファイルのアーカイブを保存、整理、管理するメカニズムです。この機能の導入前にも、実行コンフィギュレーションのコピーを `copy running-config destination-url` コマンドを使用して保存し、ローカルやリモートに置換ファイルを保管できました。ただし、この方法ではファイルの自動管理を行うことはできませんでした。コンフィギュレーションの置換とロールバック機能により、実行コンフィギュレーションのコピーを自動的にコンフィギュレーションアーカイブに保存できます。アーカイブされたファイルはコンフィギュレーションのチェックポイントとして参照することができ、`configure replace` コマンドを使用してコンフィギュレーションを以前の状態に戻せます。

`archive config` コマンドを使用すると、シスコのコンフィギュレーションをコンフィギュレーションアーカイブに保存できます。その場合、標準のディレクトリとファイル名のプレフィクスが使用され、バージョン番号（およびオプションでタイムスタンプ）が自動的に付加されません。バージョン番号は連続したファイルを保存するごとに、1 つずつ大きくなります。この機能により、保存したコンフィギュレーションファイルを一貫して識別できます。アーカイブに保存する実行コンフィギュレーションの数は指定することができます。アーカイブ内のファイル数が上限値に達すると、次に最新のファイルが保存されるときに、最も古いファイルが自動的に消去されます。`show archive` コマンドを使用すると、コンフィギュレーションアーカイブに保存されているすべてのコンフィギュレーションファイルに関する情報が表示されます。

コンフィギュレーションファイルを保存するコンフィギュレーションアーカイブは、`configure replace` コマンドで使用することによって、お使いのプラットフォームに応じて次のファイルシステムに配置できます。

- `disk0` があるプラットフォーム : `disk0:`、`disk1:`、`ftp:`、`pram:`、`rcp:`、`slavedisk0:`、`slavedisk1:`、または `tftp:`
- `disk0` がないプラットフォーム : `bootflash:`、`ftp:`、`harddisk:`、`http:`、`pram:`、`rcp:`、`tftp:`、`usb0:`、または `usb1:`

## コンフィギュレーションのバージョン管理の設定方法

### 設定アーカイブの特性の設定

`archive config` コマンドを使用する前に、コンフィギュレーションアーカイブを設定しておく必要があります。コンフィギュレーションアーカイブの特性を設定するには、次の作業を実行します。

#### 手順の概要

1. `enable`
2. `configure terminal`
3. `archive`
4. `path url`
5. `maximum number`
6. `time-period minutes`

- 7. end
- 8. archive config

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します (要求された場合)。</li> </ul>
ステップ 2	<p><b>configure terminal</b></p> <p>例 :</p> <pre>Device# configure terminal</pre>	<p>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。</p>
ステップ 3	<p><b>archive</b></p> <p>例 :</p> <pre>Device(config)# archive</pre>	<p>アーカイブ コンフィギュレーション モードを開始します。</p>
ステップ 4	<p><b>path url</b></p> <p>例 :</p> <pre>Device(config-archive)# path bootflash:myconfig</pre>	<p>コンフィギュレーションアーカイブに、ファイルのディレクトリとファイル名プレフィックスを指定します。</p> <ul style="list-style-type: none"> <li>• ハードウェアプラットフォームによって、ファイルシステムの名前は、例に示しているものとは異なる可能性があります。</li> </ul> <p>(注) パスの部分でファイルの代わりにディレクトリを指定する場合、ディレクトリ名は <b>path flash:/directory/</b> のように後ろにスラッシュを付ける必要があります。このスラッシュはファイル名の後ろでは必要ありません。ディレクトリを指定する場合にだけ使います。</p>
ステップ 5	<p><b>maximum number</b></p> <p>例 :</p> <pre>Device(config-archive)# maximum 14</pre>	<p>(任意) 設定アーカイブに保存する実行設定のアーカイブ ファイルの最大数を指定します。</p> <ul style="list-style-type: none"> <li>• <b>number</b> 引数は、コンフィギュレーションアーカイブに保存される実行コンフィギュレーションのアーカイブ ファイル数の上限値を示します。指定できる範囲は 1 ~ 14 です。デフォルトは 10 です。</li> </ul>

	コマンドまたはアクション	目的
		(注) このコマンドを使用する前に、 <b>path</b> コマンドを設定して、コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 6	<b>time-period <i>minutes</i></b> 例： Device(config-archive)# time-period 10	(任意) コンフィギュレーションアーカイブに現在実行中のコンフィギュレーションのアーカイブファイルを自動保存する間隔を設定します。 • 設定アーカイブに現在の実行設定のアーカイブファイルをどれほどの頻度で自動保存するかを、 <i>minutes</i> 引数により分単位で指定します。 (注) このコマンドを使用する前に、 <b>path</b> コマンドを設定して、コンフィギュレーションアーカイブの位置とファイル名プレフィックスを指定しておく必要があります。
ステップ 7	<b>end</b> 例： Device(config-archive)# end	特権 EXEC モードに戻ります。
ステップ 8	<b>archive config</b> 例： Device# archive config	現在の実行設定ファイルを設定アーカイブに保存します。 (注) <b>archive config</b> コマンドを使用する前に、 <b>path</b> コマンドを設定する必要があります。

## コンフィギュレーションのモニタリングとトラブルシューティング

### 手順の概要

1. **enable**
2. **show archive**
3. **debug archive versioning**
4. **debug archive config timestamp**
5. **exit**

### 手順の詳細

#### ステップ 1 enable

このコマンドを使用して、特権 EXEC モードをイネーブルにします。パスワードを入力します（要求された場合）。次に例を示します。

例：

```
Device> enable
Device#
```

## ステップ 2 show archive

コンフィギュレーションアーカイブに保存されているファイルに関する情報を表示するには、次のコマンドを使用します。次に例を示します。

例：

```
Device# show archive

There are currently 1 archive configurations saved.
The next archive file will be named bootflash:myconfig-2
Archive # Name
0
1 bootflash:myconfig-1 <- Most Recent
2
3
4
5
6
7
8
9
10
11
12
13
14
```

次に、実行コンフィギュレーションのアーカイブ ファイルをいくつか保存した状態で **show archive** コマンドを使用した場合の出力例を示します。この例では、保存されるアーカイブ ファイルの最大数が 3 に設定されています。

例：

```
Device# show archive

There are currently 3 archive configurations saved.
The next archive file will be named bootflash:myconfig-8
Archive # Name
0
1 :Deleted
2 :Deleted
3 :Deleted
4 :Deleted
5 bootflash:myconfig-5
6 bootflash:myconfig-6
7 bootflash:myconfig-7 <- Most Recent
8
9
10
11
12
```

13  
14

### ステップ3 debug archive versioning

このコマンドを使用して、コンフィギュレーションアーカイブのアクティビティのデバッグを有効にして、コンフィギュレーションの置換とロールバックをモニタおよびトラブルシューティングします。次に例を示します。

例：

```
Device# debug archive versioning
Jan  9 06:46:28.419:backup_running_config
Jan  9 06:46:28.419:Current = 7
Jan  9 06:46:28.443:Writing backup file bootflash:myconfig-7
Jan  9 06:46:29.547: backup worked
```

### ステップ4 debug archive config timestamp

このコマンドを使用して、コンフィギュレーション置換操作の各必須段階の処理時間、および操作中のコンフィギュレーションファイルのサイズのデバッグをイネーブルにします。次に例を示します。

例：

```
Device# debug archive config timestamp
Device# configure replace bootflash:myconfig force
Timing Debug Statistics for IOS Config Replace operation:
  Time to read file slot0:sample_2.cfg = 0 msec (0 sec)
  Number of lines read:55
  Size of file      :1054
Starting Pass 1
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:93
  Size of file      :2539
  Time taken for positive rollback pass = 320 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for negative incremental diffs pass = 59 msec (0 sec)
  Time taken by PI to apply changes = 0 msec (0 sec)
  Time taken for Pass 1 = 380 msec (0 sec)
Starting Pass 2
  Time to read file system:running-config = 0 msec (0 sec)
  Number of lines read:55
  Size of file      :1054
  Time taken for positive rollback pass = 0 msec (0 sec)
  Time taken for negative rollback pass = 0 msec (0 sec)
  Time taken for Pass 2 = 0 msec (0 sec)
Total number of passes:1
Rollback Done
```

### ステップ5 exit

このコマンドを使用して、ユーザ EXEC モードに戻ります。次に例を示します。

例：

```
Device# exit
Device>
```

# コンフィギュレーションのバージョン管理の設定例

## 例：コンフィギュレーションアーカイブの作成

次の例は、コンフィギュレーションアーカイブの初期設定を実行する方法を示しています。この例では、`bootflash:myconfig` がコンフィギュレーションアーカイブの保存位置およびファイル名のプレフィックスとして設定され、保存するアーカイブファイルが最大 10 個に設定されます。ハードウェアプラットフォームによって、ファイルシステムの名前は、例に示しているものとは異なる可能性があります。

```
configure terminal
!
archive
 path bootflash:myconfig
 maximum 10
end
```

## その他の参考資料

### 関連資料

関連項目	マニュアルタイトル
Cisco IOS コマンド	<a href="#">『Cisco IOS Master Command List, All Releases』</a>
コンフィギュレーションファイルの管理についての情報	『コンフィギュレーションファイルの管理コンフィギュレーションガイド』の「コンフィギュレーションファイルの管理」モジュール
コンフィギュレーションファイルを管理するためのコマンド	<a href="#">『Cisco IOS Configuration Fundamentals Command Reference』</a>

### シスコのテクニカル サポート

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

## コンフィギュレーションのバージョン管理の機能情報

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

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



表 9: コンフィギュレーションのバージョン管理の機能情報

機能名	リリース	機能情報
コンフィギュレーションのバージョン管理	12.2(25)S 12.2(33)SRA 12.3(7)T Cisco IOS XE Release 2.1	<p>コンフィギュレーションのバージョン管理機能により、シスコの実行コンフィギュレーションのコピーをデバイス上やデバイス外で維持および管理することができます。コンフィギュレーション置換機能では、実行コンフィギュレーションの保存されたコピーへのロールバックを行うためにコンフィギュレーションバージョン管理機能を使用します。</p> <p>次のコマンドが導入または変更されました。 <b>archive config</b>、 <b>debug archive versioning</b>、 <b>log config</b>、 <b>maximum</b>、 <b>path</b> (アーカイブの設定)、 <b>show archive</b>、 <b>time-period</b>、 <b>write-memory</b></p>





## 第 10 章

# コンフィギュレーション ロールバック変更確認

コンフィギュレーションロールバック変更確認機能により、コンフィギュレーション変更の実行に際して確認を要求するようオプションで設定できます。この確認が受信できない場合、コンフィギュレーションは変更が適用される前の状態に戻されます。このメカニズムは、ネットワーク デバイスとユーザまたは管理アプリケーションとの接続において、コンフィギュレーション変更起因する切断を防止するものです。

- [機能情報の確認 \(121 ページ\)](#)
- [コンフィギュレーション ロールバック変更確認について \(122 ページ\)](#)
- [コンフィギュレーション ロールバック変更確認の設定方法 \(122 ページ\)](#)
- [コンフィギュレーション ロールバック変更確認の設定例 \(125 ページ\)](#)
- [その他の参考資料 \(125 ページ\)](#)
- [コンフィギュレーション ロールバック変更確認の機能情報 \(126 ページ\)](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

# コンフィギュレーションロールバック変更確認について

## コンフィギュレーション ロールバック変更確認の操作

コンフィギュレーションロールバック変更確認機能は、コンフィギュレーションの変更の確認条件を追加できる機能です。この機能により、要求された変更の確認が設定済みの時間枠以内に受信されない場合にロールバックを行うことができます。コマンドの失敗を、コンフィギュレーションのロールバックをトリガーするように設定することもできます。

次に、このプロセスを実施するための手順の概要を示します。

1. 新しいオプションを使用すると、コンフィギュレーションの変更の確認を要求できます（確認の時間制限を指定する必要があります）。
2. 確認コマンドを入力する必要があります。要求された制限時間内に確認を入力しないと、コンフィギュレーションは以前の状態に戻ります。

## コンフィギュレーションロールバック変更確認の設定方法

### コンフィギュレーションの置換またはコンフィギュレーションのロールバック操作の確認を伴う実行

現在の実行コンフィギュレーション ファイルを保存済みのシスコのコンフィギュレーション ファイルに置換するには、次のタスクを実行します。



- (注) この手順の前に、コンフィギュレーションアーカイブを設定しておく必要があります。手順の詳細については、『コンフィギュレーションファイルの管理コンフィギュレーションガイド』の「コンフィギュレーションアーカイブの特性の設定」モジュールを参照してください。次に、現在の実行コンフィギュレーションで問題が生じた場合に、アーカイブしておいたコンフィギュレーションに戻す手順の詳細を示します。

#### 手順の概要

1. `enable`
2. `configure replace target-url [nolock] [list] [force] [ignorecase] [revert trigger [error] [timer minutes] | time minutes]`
3. `configure revert {now | timer {minutes | idle minutes}}`
4. `configure confirm`

## 5. exit

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<p><b>enable</b></p> <p>例 :</p> <pre>Device&gt; enable</pre>	<p>特権 EXEC モードをイネーブルにします。</p> <ul style="list-style-type: none"> <li>• パスワードを入力します（要求された場合）。</li> </ul>
ステップ 2	<p><b>configure replace <i>target-url</i> [<b>nolock</b>] [<b>list</b>] [<b>force</b>] [<b>ignorecase</b>] [<b>revert trigger</b>] [<b>error</b>] [<b>timer <i>minutes</i></b>] [<b>time <i>minutes</i></b>]</b></p> <p>例 :</p> <pre>Device# configure replace bootflash:myconfig-1 list time 30</pre>	<p>現在の実行コンフィギュレーションファイルを保存済みのコンフィギュレーションファイルに置換します。</p> <ul style="list-style-type: none"> <li>• <b>target-url</b> : <b>archive config</b> コマンドで作成されたコンフィギュレーションファイルなど、現在の実行コンフィギュレーションを置き換える、保存済みのコンフィギュレーションファイルの URL を指定します（シスコのファイルシステムでアクセス可能なもの）。ハードウェアプラットフォームによって、ファイルシステムの名前は、例に示しているものとは異なる可能性があります。</li> <li>• <b>nolock</b> : コンフィギュレーション置換操作中に他のユーザが実行コンフィギュレーションを変更しないように実行コンフィギュレーションファイルをロックする機能をオフにします。</li> <li>• <b>list</b> : コンフィギュレーション置換動作のパスごとに、シスコのソフトウェアパーサーによって適用されるコマンドラインのリストを表示します。実行されたパスの総数も表示されます。</li> <li>• <b>force</b> : 現在の実行コンフィギュレーションファイルと指定した保存済みコンフィギュレーションファイルの交換を確認なしで実行します。</li> <li>• <b>ignorecase</b> : コンフィギュレーションに確認コマンドの大文字と小文字の区別を無視させることができます。</li> <li>• <b>time <i>minutes</i></b> : 現在の実行コンフィギュレーションファイルの置換確認のために <b>configure confirm</b> コマンドを入力しなければならない制限時間（分単位）を指定します。<b>configure confirm</b> コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作</li> </ul>

	コマンドまたはアクション	目的
		<p>は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルが <b>configure replace</b> コマンド入力以前のコンフィギュレーション状態へと回復されます）。</p> <ul style="list-style-type: none"> <li>• <b>revert trigger</b> : 元のコンフィギュレーションへ戻すトリガーを次の内容から設定します。 <ul style="list-style-type: none"> <li>• <b>error</b> : エラー時に元のコンフィギュレーションに戻します。</li> <li>• <b>timer minutes</b> : 指定した時間が過ぎると元のコンフィギュレーションに戻します。</li> </ul> </li> </ul>
<p>ステップ 3</p>	<p><b>configure revert {now   timer {minutes   idle minutes}}</b></p> <p>例 :</p> <pre>Device# configure revert now</pre>	<p>(任意) 時間指定ロールバックをキャンセルしてロールバックを即時トリガーするか、または時間指定ロールバックのパラメータをリセットします。</p> <ul style="list-style-type: none"> <li>• <b>now</b> : ロールバックをただちにトリガーします。</li> <li>• <b>timer</b> : コンフィギュレーションを元に戻すタイマーをリセットします。 <ul style="list-style-type: none"> <li>• 元に戻す時間を分単位で新たに指定するには、<i>minutes</i> 引数を <b>timer</b> キーワードとともに使用します。</li> <li>• 保存されたコンフィギュレーションに戻すまでに、操作が行われないアイドル時間を最大どれほど長く許容できるかを設定するには、分単位の時間とともに <b>idle</b> キーワードを使用します。</li> </ul> </li> </ul>
<p>ステップ 4</p>	<p><b>configure confirm</b></p> <p>例 :</p> <pre>Device# configure confirm</pre>	<p>(任意) 現在の実行コンフィギュレーションファイルが保存済みのコンフィギュレーションファイルに置換されることを確認します。</p> <p>(注) このコマンドは、<b>configure replace</b> コマンドの <b>time minutes</b> キーワードおよび引数が指定されている場合にのみ使用してください。</p>
<p>ステップ 5</p>	<p><b>exit</b></p> <p>例 :</p> <pre>Device# exit</pre>	<p>ユーザ EXEC モードに戻ります。</p>

# コンフィギュレーションロールバック変更確認の設定例

## 例：configure confirm コマンドを使用したコンフィギュレーション置換操作の実行

次に、**configure replace** コマンドを **time minutes** キーワードおよび引数とともに使用する例を示します。現在の実行コンフィギュレーションファイルの置換を実行するには、指定の制限時間内に **configure confirm** コマンドを入力する必要があります。**configure confirm** コマンドが指定の制限時間内に入力されない場合、コンフィギュレーション置換操作は自動的に戻されます（つまり、現在の実行コンフィギュレーションファイルが **configure replace** コマンド入力以前のコンフィギュレーション状態へと回復されます）。

```
Device# configure replace nvram:startup-config time 120
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
Device# configure confirm
```

次に、**configure revert** コマンドを **timer** キーワードとともに使用する例を示します。時間指定ロールバックをキャンセルしてロールバックを即時トリガーする、または時間指定ロールバックのパラメータをリセットするには、**configure revert** コマンドを入力する必要があります。

```
Device# configure revert timer 100
```

## その他の参考資料

### 関連資料

関連項目	マニュアルタイトル
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
コンフィギュレーションファイルの管理についての情報	『コンフィギュレーションファイルの管理コンフィギュレーションガイド』の「コンフィギュレーションファイルの管理」モジュール
コンフィギュレーションファイルを管理するためのコマンド	『Cisco IOS Configuration Fundamentals Command Reference』

シスコのテクニカル サポート

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

## コンフィギュレーションロールバック変更確認の機能情報

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

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



表 10:コンフィギュレーションロールバック変更確認の機能情報

機能名	リリース	機能情報
コンフィギュレーションロールバック変更確認	12.2(33)SB 12.2(33)SRC 12.2(33)SXI 12.4(20)T Cisco IOS XE Release 2.1	<p>コンフィギュレーションロールバック変更確認機能により、コンフィギュレーション変更の実行に際して確認を要求するようオプションで設定できます。</p> <p>このメカニズムは、ネットワークデバイスとユーザまたは管理アプリケーションとの接続に、誤ったコンフィギュレーション変更に起因する切断を防止するものです。</p> <p>次のコマンドが導入または変更されました。<b>configure confirm</b>、<b>configure replace</b>、<b>configure revert</b>、<b>configure terminal</b></p>





## 第 11 章

# コンフィギュレーション ロガー永続性

コンフィギュレーション ロガー永続性機能は「クイック保存」機能を実装することで、Cisco IOS コンフィギュレーションとプロビジョニングアクションの運用上の堅牢性を高めます。コンフィギュレーション ロガー永続性機能を設定すると、Cisco IOS ソフトウェアはスタートアップ コンフィギュレーション全体を保存するのではなく、最後の startup-config ファイルが生成された時点から入力されたコマンドだけを保存します。

- [機能情報の確認 \(129 ページ\)](#)
- [コンフィギュレーション ロガー永続性の前提条件 \(129 ページ\)](#)
- [コンフィギュレーション ロガー永続性について \(130 ページ\)](#)
- [コンフィギュレーション ロガー永続性機能を設定する方法 \(131 ページ\)](#)
- [コンフィギュレーション ロガー永続性機能の設定例 \(135 ページ\)](#)
- [その他の参考資料 \(135 ページ\)](#)
- [コンフィギュレーション ロガー永続性の機能情報 \(136 ページ\)](#)
- [用語集 \(137 ページ\)](#)

## 機能情報の確認

ご使用のソフトウェアリリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「[Bug Search Tool](#)」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

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

## コンフィギュレーション ロガー永続性の前提条件

コンフィギュレーション ロガー永続性機能をイネーブルにするには、disk0: を構成し、ルータ上に外部フラッシュ カードを挿入する必要があります。

コンフィギュレーション ロガー永続性機能の最適な結果を実現するためには、Cisco IOS Release 12.2(33)SRA、Release 12.4(11)T、Release 12.2(33)SXH、または Release 12.2(33)SB をシステムにインストールする必要があります。

## コンフィギュレーション ロガー永続性について

### コンフィギュレーション ロガー永続性を使用したコンフィギュレーション ファイルの保存

Cisco IOS ソフトウェアは `startup-config` コンフィギュレーション ファイルを使用して、リロード全体でルータ コンフィギュレーション コマンドを保存します。この単一のファイルには、ルータのリブート時に適用する必要があるすべてのコマンドが含まれています。スタートアップ コンフィギュレーション ファイルは、`writememory` コマンドまたは `copyurl/startup-config` コマンドを入力するたびに更新されます。`running-config` ファイルのサイズが大きくなると、`startup-config` ファイルを NVRAM ファイル システムに保存する時間が長くなります。スタートアップ コンフィギュレーション ファイルは、1 MB 以上にすることができます。このサイズのファイルの場合、`startup-config` ファイルの 1 行を変更すると、ほとんどのコンフィギュレーションが変更されていない場合でも、全体の `startup-config` ファイルを再度保存する必要があります。

コンフィギュレーション ロガー永続性機能は「クイック保存」機能を実装しています。この目的は、`startup-config` ファイルの変更を保存する時間が保存する (`startup-config` ファイルと相対した) 差分変更のサイズに比例する「コンフィギュレーション保存」のメカニズムを提供することです。

Cisco IOS コンフィギュレーション ロガーは、コマンドラインプロンプトで手動で入力されたすべての変更をログに記録します。この機能では、ログの変更が発生したときに登録済みのクライアントに通知します。設定ログの内容はランタイムメモリに保存されます。ログの内容は再起動後は保持されません。

コンフィギュレーション ロガー永続性機能は、リロード全体でユーザが入力したコンフィギュレーション コマンドを保持するメカニズムです。リロード後も保持されるのは、コマンドライン インターフェイス (CLI) で入力したコマンド (コンフィギュレーション モードで入力したコマンド) のみです。この機能は Cisco IOS のセキュア ファイル システムを使用して、生成されるコンフィギュレーション コマンドを保持します。



(注) Cisco IOS コンフィギュレーション ロガーはシステム メッセージ ロギング (syslog) 機能とは別のものです。Syslog はシステム メッセージを追跡するための一般的なログ ファシリティです。コンフィギュレーション ロガーは、CLI で入力された設定コマンドに関する情報を記録します。

## 保持されたコマンド

Cisco IOS コンフィギュレーション ロガーで保持されたコマンドはスタートアップ コンフィギュレーションの拡張として使用されます。これらの保存されたコマンドでは、クイック保存機能が提供されます。startup-config ファイル全体を保存するのではなく、Cisco IOS ソフトウェアは最後の startup-config ファイル生成以降入力されたコマンドだけを保存します。

ログ出力されたコマンドだけが保持されます。コンフィギュレーション ロガーの次の追加データは保持されません。

- コマンドを出力したユーザ
- ユーザがログインした IP アドレス
- ログに記録されたコマンドのセッションおよびログ インデックス
- コマンドが入力された時刻
- 入力されたコマンドに関連付けられている前後の NVGEN 出力
- 入力されたコマンドに対するパーサーからの戻りコード

コマンドを保持する主な目的は、startup-config ファイルのクイック保存の拡張として使用することです。コンフィギュレーション コマンドに関連付けられている追加情報はクイック保存目的では有用ではありません。（監査の目的で）再起動後に追加情報を保持する必要がある場合は、次の手順を実行します。

1. Syslog へのコンフィギュレーション ロガー通知をイネーブルにします。
2. Syslog 保持機能のイネーブル化

代わりに、Cisco Networking Services、CiscoView、または、Cisco IOS デバイスを管理して標準外のストレージソリューションのコンフィギュレーション変更を追跡するその他のネットワーク管理システムを使用できます。

デフォルトでは、リロード時に、保持されたコマンドが startup-config ファイルの末尾に追加されます。CLI コンフィギュレーション コマンドを使用して明示的にこの動作を設定した場合にだけこれらのコマンドが適用されます。

# コンフィギュレーション ロガー永続性機能を設定する方法

## コンフィギュレーション ロガー永続性機能のイネーブル化

コンフィギュレーション ロガー永続性機能はクイック保存メカニズムを実装するため、スタートアップ コンフィギュレーションの変更を保存するためにかかる時間が、保存する必要がある（スタートアップ コンフィギュレーションと相対した）差分変更のサイズに比例します。Cisco

IOS コンフィギュレーション ロガーで保持されたコマンドはスタートアップコンフィギュレーションの拡張として使用されます。保存されたコマンドは、スタートアップコンフィギュレーションの拡張として使用され、クイック保存の機能を提供します。startup-config ファイル全体を保存するのではなく、Cisco IOS ソフトウェアは最後の startup-config ファイル生成以降入力されたコマンドだけを保存します。

コンフィギュレーション ロガー永続性機能をイネーブルにするには、次の作業を実行します。

### 手順の概要

1. **enable**
2. **configure terminal**
3. **archive**
4. **log config**
5. **logging persistent auto manual**
6. **logging persistent reload**
7. **logging persistent size threshold**
8. **logging size entries**

### 手順の詳細

	コマンドまたはアクション	目的
ステップ 1	<b>enable</b> 例： Router> enable	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	<b>configure terminal</b> 例： Router# 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	<b>archive</b> 例： Router(config)# archive	アーカイブ コンフィギュレーション モードを開始します。
ステップ 4	<b>log config</b> 例： Router(config-archive)# log config	アーカイブ configuration-log コンフィギュレーション モードをイネーブルにします。
ステップ 5	<b>logging persistent auto manual</b> 例： Router(config-archive-log-cfg)# logging persistent auto	コンフィギュレーション ロガー永続性機能をイネーブルにします。 • <b>auto</b> キーワードは、各コンフィギュレーション コマンドが自動的に Cisco IOS セキュア ファイル システムに保存されることを指定します。

	コマンドまたはアクション	目的
		<p>• <b>manual</b> キーワードは、コンフィギュレーション コマンドを Cisco IOS セキュア ファイル システムにオンデマンドで保存できることを指定します。これを行うには、<b>archive log config persistent save</b> コマンドを使用する必要があります。</p> <p>(注) <b>logging persistent auto</b> コマンドをイネーブ ルにするには、<b>disk0:</b> を構成し、ルータに外部フラッシュ カードを挿入する必要があります。</p>
<p>ステップ 6</p>	<p><b>logging persistent reload</b></p> <p>例 :</p> <pre>Router(config-archive-log-cfg)# logging persistent reload</pre>	<p>続いて、リロード後に、コンフィギュレーション ロガー データベースに保存された (最後の <b>writememory</b> コマンド以降の) コンフィギュレーション コマンドを実行コンフィギュレーション ファイルに適用します。</p>
<p>ステップ 7</p>	<p><b>logging persistent size threshold</b></p> <p>例 :</p> <pre>Router(config-archive-log-cfg)# logging persistent size threshold</pre>	<p>コンフィギュレーション ロガー データベースにログメッセージを書き込むためのディスク領域のサイズを指定します。ログサイズがしきい値 (パーセントで指定) を超えると、コンソールまたは <b>syslog</b> サーバでアラートがトリガーされます。</p>
<p>ステップ 8</p>	<p><b>logging size entries</b></p> <p>例 :</p> <pre>Router(config-archive-log-cfg)# logging size 10</pre>	<p>設定ログに保持する最大エントリ数を指定します。</p> <ul style="list-style-type: none"> <li>有効な値の範囲は、1 ~ 1000 です。</li> <li>デフォルト値は 100 エントリです。</li> </ul>

## コンフィギュレーション ロガー永続性機能の検証とトラブルシューティング

3つのコマンドを使用して、設定ログの内容を検証、アーカイブ、クリアできます。トラブルシューティングでは、ステップ 4 のコマンドでデバッグをオンにします。

### 手順の概要

1. **show archive log config persistent**
2. **clear archive log config persistent**
3. **archive log config persistent save**
4. **debug archive log config persistent**

## 手順の詳細

### ステップ1 show archive log config persistent

このコマンドは設定ログに保持されたコマンドを表示します。このコマンドは `configlet` 形式で表示されます。次に、このコマンドの出力例を示します。

例：

```
Router# show archive log config persistent
!Configuration logger persistentarchive
 log config
 logging persistent auto
 logging persistent reload
archive
 log config
 logging size 10
 logging console
interface loop 101
 ip address 10.1.1.1 255.255.255.0
 ip address 10.2.2.2 255.255.255.0
 no shutdown
```

### ステップ2 clear archive log config persistent

このコマンドはコンフィギュレーション ロギング永続データベース エントリをクリアします。コンフィギュレーション ロギング データベース ファイルのエントリだけが削除されます。ファイル自体は、新しいエントリを記録するために使用されるため、削除されません。このコマンドを入力すると、アーカイブログがクリアされたことを示すメッセージが表示されます。

例：

```
Router# clear archive log config persistent
Purged the config log persist database entries successfully
Router#
```

### ステップ3 archive log config persistent save

このコマンドは Cisco IOS セキュア ファイル システムに設定ログを保存します。このコマンドを有効にするには、`archivelogconfigpersistentsave` コマンドを設定する必要があります。

### ステップ4 debug archive log config persistent

このコマンドはデバッグ機能をオンにします。デバッグがオンになっていることを示すメッセージが返されます。

例：

```
Router# debug archive log config persistent
debug archive log config persistent debugging is on
```



# コンフィギュレーション ロガー永続性機能の設定例

## Cisco 7200 シリーズ ルータでのコンフィギュレーション ロガー永続性機能の設定例

この例では、各コンフィギュレーション コマンドが自動的に Cisco IOS セキュア ファイル システムに保存され、（最後の **writememory** コマンドの実行以降）コンフィギュレーション ロガー データベースに保存されたコンフィギュレーション コマンドが実行コンフィギュレーション ファイルに適用され、設定ログに保持される最大エン트리数が 10 に設定されます。

```
Router> enable
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# archive
Router(config-archive)# log config
Router(config-archive-log-config)# logging persistent auto
configuration log persistency feature enabled. Building configuration... [OK]
Router(config-archive-log-config)# logging persistent reload
Router(config-archive-log-config)# logging persistent size 16384 threshold 10
Router(config-archive-log-config)# logging size 10
Router(config-archive-log-config)# archive log config persistent save
Router(config-archive-log-config)# end
Router#
```

## その他の参考資料

次の項に、コンフィギュレーション ロガー永続性機能に関する参考資料を示します。

### 関連資料

関連項目	マニュアル タイトル
包括的なコマンドリファレンス情報	<i>Cisco IOS Configuration Fundamentals Command Reference</i>

### 標準

標準	タイトル
この機能がサポートする新しい規格または変更された規格はありません。	--

### MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。	<p>選択したプラットフォーム、Cisco IOS リリース、およびフィーチャセットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。</p> <p><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></p>

### RFC

RFC	タイトル
この機能でサポートが追加または変更された RFC はありません。	--

### テクニカル サポート

説明	Link
<p>シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。</p> <p>お使いの製品のセキュリティ情報や技術情報入手するために、Cisco Notification Service (Field Notice からアクセス)、Cisco Technical Services Newsletter、Really Simple Syndication (RSS) フィードなどの各種サービスに加入できます。</p> <p>シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。</p>	<p><a href="http://www.cisco.com/en/US/support/index.html">http://www.cisco.com/en/US/support/index.html</a></p>

## コンフィギュレーション ロガー永続性の機能情報

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

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

表 11: コンフィギュレーション ロガー永続性の機能情報

機能名	リリース	機能情報
コンフィギュレーション ロガー永続性	12.2(33)SRA 12.4(11)T 12.2(33)SXH 12.2(33)SB Cisco IOS XE Release 3.9S	<p>コンフィギュレーション ロガー永続性機能は「クイック保存」機能を実装することで、シスコのコンフィギュレーションとプロビジョニングアクションの運用上の堅牢性を高めます。</p> <p>Cisco IOS Release 12.2(33)SRA、Release 12.4(11)T、Release 12.2(33)SXH、Release 12.2(33)SB で有効なシスコのソフトウェアは、スタートアップコンフィギュレーション全体を保存するのではなく、最後の startup-config ファイルが生成された時点から入力されたコマンドだけを保存します。</p> <p>この機能は、Cisco IOS XE Release 3.9S に統合されました。</p>

## 用語集

**API** : アプリケーション プログラミング インターフェイス。

**CAF** : コマンド アクション機能。

**CDP** : Cisco Discovery Protocol。

**CSB** : コマンド ステータス ブロック。

**HA** : 高可用性アーキテクチャ。

**MIB** : 管理情報ベース。

**NAF** : NVGEN アクション機能。

**NVGEN** : 不揮発生成。

**NVRAM** : 不揮発性ランダム アクセス メモリ

**parse chain** : Cisco IOS コマンドの構文を定義する一連の C 言語マクロ。

**RP** : ルート プロセッサ。

**SNMP** : 簡易ネットワーク管理プロトコル。

**XML** : 拡張マークアップ言語。





## 第 12 章

# ソフトウェア メンテナンス アップグレード

ソフトウェア メンテナンス アップグレード (SMU) は、システムにインストールしてパッチ修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。

- [ソフトウェア メンテナンス アップグレードについて \(139 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの制約事項 \(141 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの設定方法 \(141 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの設定例 \(142 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードのその他の参考資料 \(148 ページ\)](#)
- [ソフトウェア メンテナンス アップグレードの機能情報 \(148 ページ\)](#)

## ソフトウェア メンテナンス アップグレードについて

### ソフトウェア メンテナンス アップグレード

ソフトウェア メンテナンス アップグレード (SMU) は、システムにインストールしてパッチ修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。

SMU パッケージはリリースごとおよびコンポーネントごとに提供され、プラットフォームに固有です。

SMU はネットワークの問題に迅速に対応できるようにするとともに、必要なテストの時間と範囲を削減するため、従来の IOS ソフトウェアには多大なメリットがあります。Cisco IOS XE プラットフォームでは SMU の互換性を内部的に検証し、互換性のない SMU はインストールできません。

すべて SMU が後続の Cisco IOS XE ソフトウェア メンテナンス リリースに統合されています。SMU は独立した自己完結型パッケージであり、前提条件や依存関係はありません。SMU はどのような順序でもインストールまたはアンインストールできます。

Cisco IOS XE Everest 16.6.1 以降、SMU は拡張メンテナンス リリースでのみ、基盤となるソフトウェア リリースのライフサイクルにわたってサポートされます。

次に、SMU をインストールする 3 つの基本ステップを示します。

- ファイルシステムへの SMU の追加
- システムでの SMU のアクティブ化
- リロード後も保持するための SMU の変更のコミット

## サポートされるプラットフォーム

Cisco IOS XE Everest 16.6.1 以降、ソフトウェアメンテナンスアップグレードについて次のプラットフォームがサポートされています。

- Cisco ASR 1000 シリーズ アグリゲーション サービス ルータ (ASR1001-X、ASR1002-X、ASR1001-HX、ASR1002-HX、ASR1000-RP2、ASR1000-RP3)
- Cisco ISR 4000 シリーズ サービス統合型ルータ (ISR4351、ISR4331、ISR4431、ISR4321、ISR4451)
- Cisco CSR 1000v シリーズ クラウド サービス ルータ
- Cisco Catalyst 9300 シリーズ スイッチ
- Cisco Catalyst 9400 シリーズ スイッチ

## ソフトウェアメンテナンスアップグレードパッケージ

SMU パッケージには、SMU が要求されている報告済みの問題のメタデータと修正が含まれています。

## ソフトウェアメンテナンスアップグレードのワークフロー

SMU プロセスは、SMU Committee への要求によって開始されます。カスタマーサポートに連絡し、SMU 要求を行います。

リリース時に、SMU パッケージは次の情報とともにシスコのソフトウェアダウンロードページに公開されます。

- 対処済みの不具合
- 不具合の種類 - PSIRT など

## SMU のリロード

SMU のタイプは、SMU のインストール後のシステムへの影響を説明します。SMU はトラフィックに影響を与えない場合もありますが、デバイスのリロードやスイッチオーバーを引き起こす可能性もあります。

システムのコールドリロードでは、オペレーティングシステムの完全なリロードが必要です。このアクションは、リロードの間（現在は最大5分間）、トラフィックフローに影響を与えません。リロードにより、SMUの一部としてインストールされている正しいライブラリとファイルですべてのプロセスが起動します。

## ソフトウェアメンテナンスアップグレードの制約事項

- In Service Software Upgrade (ISSU) のスイッチオーバーはサポートされていません。

## ソフトウェアメンテナンスアップグレードの設定方法

### SMU の追加、アクティブ化、コミット

SMU パッケージをインストールするには、ダウンロードした SMU パッケージをデバイス上の該当するディレクトリにコピーします。次のコマンドを使用して、SMU を追加、アクティブ化、およびコミットします。

- **install add** : ファイルで基本的な互換性チェックを実行し、SMU パッケージがプラットフォームでサポートされていることを確認します。また、パッケージ/SMU の .sta ファイル内にエントリを追加することで、それ以降ステータスを監視または維持できるようにします。install add コマンドは、パッケージファイルの場所とダウンロード方法 (tftp、ftp など) を入力として受け取ります。
- **install active** : 互換性チェックを実行し、パッケージをインストールして、パッケージのステータスの詳細を更新します。再起動可能なパッケージの場合は、適切なポストインストールスクリプトをトリガーして必要なプロセスを再起動します。また、再起動できないパッケージの場合は、リロードをトリガーします。
- **install commit** リロードが繰り返されても保持されるようにアクティブ化の変更をコミットします。アクティブ化の後で、システムがアップしている間、または最初のリロード後にコミットできます。パッケージがアクティブになってもコミットされなかった場合は、最初のリロード後はアクティブの状態を保ちますが、2 回目のリロード後はアクティブ状態を保ちません。

次の設定を実行して、SMU を追加、アクティブ化、およびコミットします。

```
enable
install add file bootflash:isr4300-universalk9.BLD_
SMU_LATEST_20170128_040557.1.CSCxxx.SSA.smu.bin

show install summary // Shows the installed SMU package as inactive package in the
command output

install activate file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxx.SSA.smu.bin

show version // Shows the image version tagged with the "SMU Patched" phrase
```

```
show install summary // Shows the installed SMU package as an active package in the
command output

install commit

show install summary // Shows the installed SMU package as a committed package in the
command output.
```

## SMU のロールバック、非アクティブ化、または削除

次のコマンドを使用して、SMU をロールバック、非アクティブ化、および削除します。

- **install rollback** : デバイスを以前のインストール状態に戻します。このロールバックにはリロードが必要です。
- **install deactivate** : アクティブなパッケージを非アクティブ化し、パッケージステータスを更新し、再起動またはリロードするプロセスをトリガーします。
- **install remove** : すべての、または指定した非アクティブな SMU パッケージをファイルシステムから削除します。

次のタスクを実行して、SMU をロールバック、非アクティブ化、および削除します。

```
enable
install rollback to committed

install deactivate file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxXXXX.SSA.smu.bin

install remove file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxx.SSA.smu.bin
```

## ソフトウェアメンテナンスアップグレードの設定例

### 例 : SMU の追加、アクティブ化、コミット

#### SMU の追加、アクティブ化、コミット

次の例は、SMU の追加、アクティブ化、コミットのワークフローを示しています。

```
Device# install add file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxXXXX.SSA.smu.bin
install_add: START Tue Aug 1 04:22:48 UTC 2017
install_add: Adding SMU

*Aug 1 04:22:54.492: %IOSXE-5-PLATFORM: SIP2: Aug 1 04:22:54 install_engine.sh:
%INSTALL-5-INSTALL_START_INFO: Started install add
bootflash:isr4300-universalk9.16.06.01.CSCxxXXXX.SPA.smu.bin--- Starting SMU Add operation
---
Performing SMU_ADD on Active/Standby
```



```

[R0] SMU_ADD package(s) on R0
[R0] Finished SMU_ADD on R0
Checking status of SMU_ADD on [R0]
SMU_ADD: Passed on [R0]
Finished SMU Add operation

SUCCESS: install_add Tue Aug 1 04:23:10 UTC 2017

Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   I    bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C    16.6.1.0

Device# install activate file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxx.SSA.smu.bin

install_activate: START Tue Aug 1 04:24:42 UTC 2017
install_activate: Activating SMU

*Aug 1 04:24:48.682: %IOSXE-5-PLATFORM: SIP2: Aug 1 04:24:48 install_engine.sh:
%INSTALL-5-INSTALL_START_INFO: Started install activate
bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
This operation requires a reload of the system. Do you want to proceed? [y/n]y

--- Starting SMU Activate operation ---
Performing SMU_ACTIVATE on Active/Standby
[R0] SMU_ACTIVATE package(s) on R0
     DMP package.
[R0] Finished SMU_ACTIVATE on R0
Checking status of SMU_ACTIVATE on [R0]
SMU_ACTIVATE: Passed on [R0]
Finished SMU Activate operation

install_activate: Reloading the box to complete activation of the SMU...
install_activate will reload the system now!
Aug 1 04:25:36
Aug 1 04:25:45.742 RP0/0: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install activate
SMU bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin

<after reload>

Device# show version
Cisco IOS XE Software, Version 16.06.01 - SMU-PATCHED
Cisco IOS Software [Everest], ISR Software (X86_64_LINUX_IOSD-UNIVERSALK9-M), Version
16.6.1, RELEASE SOFTWARE (fc2)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2017 by Cisco Systems, Inc.
Compiled Sat 22-Jul-17 05:55 by mcpre

...

Active SMU Information:
State (St): C - Committed, U - Uncommitted
-----
Type  Defect_ID  Version  St  Filename
-----
SMU   CSCxxXXXXX  16.6.1.0  U   isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu
-----

```

## 例: SMU の追加、アクティブ化、コミット

```
cisco ISR4351/K9 (2RU) processor with 7941107K/6147K bytes of memory.
Processor board ID FLM2007WOMJ
3 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
16777216K bytes of physical memory.
14659583K bytes of flash memory at bootflash:.
0K bytes of WebUI ODM Files at webui:.
```

```
Configuration register is 0x0
```

```
Device# show install summary
```

```
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type  St  Filename/Version
-----
```

```
SMU   U   bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C   16.6.1.0
```

```
Device# show install active
```

```
[ R0 ] Active Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type  St  Filename/Version
-----
```

```
SMU   U   bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C   16.6.1.0
```

```
Device# install commit
```

```
install_commit: START Tue Aug 1 04:48:03 UTC 2017
install_commit: Committing SMU
```

```
*Aug 1 04:48:10.042: %IOSXE-5-PLATFORM: SIP2: Aug 1 04:48:10 install_engine.sh:
%INSTALL-5-INSTALL_START_INFO: Started install commit--- Starting SMU Commit operation
---
```

```
Performing SMU_COMMIT on Active/Standby
```

```
[R0] SMU_COMMIT package(s) on R0
```

```
[R0] Finished SMU_COMMIT on R0
```

```
Checking status of SMU_COMMIT on [R0]
```

```
SMU_COMMIT: Passed on [R0]
```

```
Finished SMU Commit operation
```

```
SUCCESS: install_commit Tue Aug 1 04:48:33 UTC 2017
```

```
Device# show install summary
```

```
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type  St  Filename/Version
-----
```

```
SMU   C   bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C   16.6.1.0
```

## 例：SMUのロールバック、非アクティブ化、または削除

### 例：SMUのロールバック、非アクティブ化、または削除

```

Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
SMU   C    bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C    16.6.1.0

Device# show install rollback
ID      Label      Description
-----
4       No Label    No Description

Device# install rollback to committed

install_rollback: START Tue Aug  1 05:00:37 UTC 2017

*Aug  1 05:00:44.038: %IOSXE-5-PLATFORM: SIP2: Aug  1 05:00:44 install_engine.sh:
%INSTALL-5-INSTALL_START_INFO: Started install rollbackinstall_rollback: Rolling back
SMU

This operation requires a reload of the system. Do you want to proceed? [y/n]y

--- Starting SMU Rollback operation ---
Performing SMU_ROLLBACK on Active/Standby
  [R0] SMU_ROLLBACK package(s) on R0
  [R0] Finished SMU_ROLLBACK on R0
Checking status of SMU_ROLLBACK on [R0]
SMU_ROLLBACK: Passed on [R0]
Finished SMU Rollback operation

install_rollback will reload the system now!
Aug  1 05:01:40.43
Aug  1 05:01:53.558 RP0/0: %INSTALL-5-INSTALL_COMPLETED_INFO: Completed install rollback
SMU

<after reload>

Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
           C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St   Filename/Version
-----
IMG   C    16.6.1.0

//install deactivate: Deactivates an active package and triggers a process restart or a
reload.

Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,

```

例: SMU のロールバック、非アクティブ化、または削除

```

C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
SMU   C   bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C   16.6.1.0

Device# install deactivate file
bootflash:isr4300-universalk9.BLD_SMU_LATEST_20170128_040557.1.CSCxxXXXX.SSA.smu.bin
install_deactivate: START Tue Aug  1 05:28:47 UTC 2017
install_deactivate: Deactivating SMU

This operation requires a reload of the system. Do you want to proceed? [y/n]y

--- Starting SMU Deactivate operation ---
Performing SMU_DEACTIVATE on Active/Standby
  [R0] SMU_DEACTIVATE package(s) on R0
      DMP package.
  [R0] Finished SMU_DEACTIVATE on R0
Checking status of SMU_DEACTIVATE on [R0]
SMU_DEACTIVATE: Passed on [R0]
Finished SMU Deactivate operation

install_deactivate: Reloading the box to complete activation of the SMU...
install_deactivate will reload the system now!

<after reload>

Device# show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----
SMU   D   bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG   C   16.6.1.0

Device#install commit
install_commit: START Tue Aug  1 05:39:29 UTC 2017
install_commit: Committing SMU

*Aug  1 05:39:35.222: %IOSXE-5-PLATFORM: SIP2: Aug  1 05:39:35 install_engine.sh:
%INSTALL-5-INSTALL_START_INFO: Started install commit--- Starting SMU Commit operation
---
Performing SMU_COMMIT on Active/Standby
  [R0] SMU_COMMIT package(s) on R0
  [R0] Finished SMU_COMMIT on R0
Checking status of SMU_COMMIT on [R0]
SMU_COMMIT: Passed on [R0]
Finished SMU Commit operation

SUCCESS: install_commit  Tue Aug  1 05:39:58 UTC 2017
Completed install commit SMU

Device#show install summary
[ R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
            C - Activated & Committed, D - Deactivated & Uncommitted
-----
Type  St  Filename/Version
-----

```

```
SMU I bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
IMG C 16.6.1.0
```

```
//install remove: Deletes the inactive SMU file from the file system.
```

```
Device# show install summary
```

```
[ R0 ] Installed Package(s) Information:
```

```
State (St): I - Inactive, U - Activated & Uncommitted,
```

```
          C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type St  Filename/Version
-----
```

```
SMU I bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
```

```
IMG C 16.6.1.0
```

```
Device#install remove file bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
```

```
install_remove: START Tue Aug 1 05:43:22 UTC 2017
```

```
install_remove: Removing SMU
```

```
--- Starting SMU Remove operation ---
```

```
Performing SMU_REMOVE on Active/Standby
```

```
  [R0] SMU_REMOVE package(s) on R0
```

```
  [R0] Finished SMU_REMOVE on R0
```

```
Checking status of SMU_REMOVE on [R0]
```

```
SMU_REMOVE: Passed on [R0]
```

```
Finished SMU Remove operation
```

```
SUCCESS: install_remove Tue Aug 1 05:43:43 UTC 2017
```

```
//Remove inactive: Deletes all inactive packages from the file system
```

```
Device#show install summary
```

```
[ R0 ] Installed Package(s) Information:
```

```
State (St): I - Inactive, U - Activated & Uncommitted,
```

```
          C - Activated & Committed, D - Deactivated & Uncommitted
```

```
-----
Type St  Filename/Version
-----
```

```
SMU I bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
```

```
IMG C 16.6.1.0
```

```
Device#install remove inactive
```

```
install_remove: START Tue Aug 1 05:52:31 UTC 2017
```

```
Cleaning up unnecessary package files
```

```
  Scanning boot directory for packages ... done.
```

```
  Preparing packages list to delete ...
```

```
    isr4300-universalk9.16.06.01.SPA.bin
```

```
      File is in use, will not delete.
```

```
    packages.conf
```

```
      File is in use, will not delete.
```

```
  done.
```

```
The following files will be deleted:
```

```
[R0]:
```

```
/bootflash/isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
```

```
Do you want to remove the above files? [y/n]y
```

```
[R0]:
```

```
Deleting file bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin ... done.
```

```
SUCCESS: Files deleted.
```

```
--- Starting Post_Remove_Cleanup ---
```

```

Performing Post_Remove_Cleanup on Active/Standby
  [R0] Post_Remove_Cleanup package(s) on R0
  [R0] Finished Post_Remove_Cleanup on R0
Checking status of Post_Remove_Cleanup on [R0]
Post_Remove_Cleanup: Passed on [R0]
Finished Post_Remove_Cleanup

SUCCESS: install_remove Tue Aug  1 05:53:19 UTC 2017

///Show install package

Device#show install package bootflash:isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
Name: isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin
Version: 16.6.1.0.202.1500742946..Everest
Platform: ISR4300
Package Type: SMU
Defect ID: CSCxxXXXXX
Package State: Not Installed
Supersedes List: {}
SMU ID: 0
SMU Type: reload
SMU Compatible with Version: 16.6.1.0.202

//Show install log
Device#show install log
[0|install_op_boot]: START Tue Aug  1 05:34:59 Universal 2017
[0|install_op_boot(INFO, )]: SMU
/bootflash/isr4300-universalk9.16.06.01.CSCxxXXXXX.SPA.smu.bin will be activated upon
reload.
[0|install_op_boot]: END SUCCESS Tue Aug  1 05:35:06 Universal 2017

```

## ソフトウェアメンテナンスアップグレードのその他の参考資料

### 関連資料

関連項目	マニュアルタイトル
Cisco IOS コマンド	<a href="#">『Cisco IOS Master Commands List, All Releases』</a>

## ソフトウェアメンテナンスアップグレードの機能情報

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

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

表 12: ソフトウェアメンテナンスアップグレードの機能情報

機能名	リリース	機能情報
ソフトウェアメンテナンスアップグレード	Cisco IOS XE Everest 16.6.1	ソフトウェアメンテナンスアップグレード (SMU) は、システムにインストールしてパッチ修正やセキュリティ解決をリリースされたイメージに提供できるパッケージです。  次のコマンドが導入または変更されました。 <b>install</b> 、 <b>show install</b>

