

Python 3を使用したIOS-XRアップグレード用GISOファイルの作成

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[関連製品](#)

[XRアップグレード用ゴールデンISO](#)

[gisobuild Pythonツール](#)

[準備](#)

[GISOの作成](#)

[確認](#)

はじめに

このドキュメントでは、eXRバージョンのCisco IOS® XRソフトウェアを実行するルータのアップグレードでのゴールデンISO(GISO)の作成について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Cisco IOS XR ソフトウェア
- Cisco IOS XR ソフトウェアのインストールおよびアップグレード手順
- Linuxの基本コマンドとコマンドラインのナビゲーション

使用するコンポーネント

このドキュメントは、特定のハードウェアバージョンに限定されるものではありません。IOS XR 64ビットを実行するすべてのルータに適用されます。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな(デフォルト)設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

関連製品

このドキュメントは、次のバージョンのハードウェアにも使用できます。

- Cisco 9000 シリーズ ルータ
- Cisco 8000 シリーズ ルータ
- Cisco NCS 5700 シリーズ ルータ
- Cisco NCS 5500 シリーズ ルータ
- NCS 540 & 560ルータ

XRアップグレード用ゴールデンISO

通常、シスコはミニ/ベースISOとしてIOS-XRソフトウェアをリリースします。これには、特定のプラットフォーム用の必須のIOS-XRパッケージ、追加機能を実行するためのオプションパッケージのセット、およびSoftware Maintenance Upgrade(SMU)の形式でのバグ修正用のソフトウェアパッチが含まれます。オプションパッケージとSMUはRPMパッケージ形式です。

ゴールデンISOツールは、ミニ/ベースISOの完全な内容と、ユーザが選択したオプションパッケージおよびSMUを含むISOを作成します。ゴールデンISOが作成されると、これをiPXEブートに使用するか、または現在実行中のバージョンから新しいバージョンのIOS-XRへのSU (システムアップグレード) に使用できます。



注：ゴールデンISOの作成にはミニベースISOが必須です

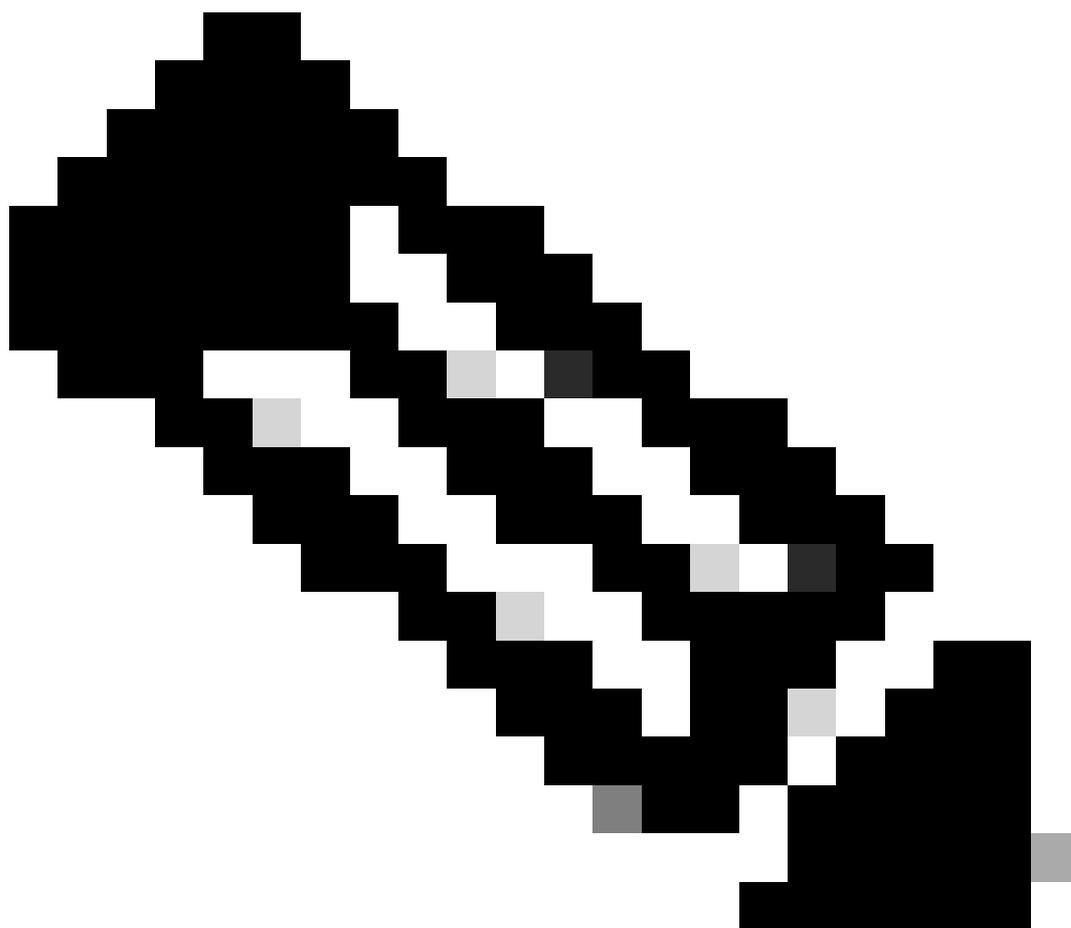
gisobuild Pythonツール

このツールは、Linuxホスト上でネイティブに実行できます。また、このツールはDockerを有効にして、公開されている「cisco-xr-gisobuild」イメージをDocker Hubからプルする機能を持つLinuxシステムでも実行できます。この例はDebian 11.8で動作することを推奨します

このツールには、次の実行可能要件があります。

- python3 >= 3.6
- rpm >= 4.14
- cpio >= 2.10
- gzip >= 1.9
- 作成リポジトリ(_c)
- ファイル
- 等値情報

- mkisofs
 - mksquashfs
 - openssl
 - unsquashfs
 - 7z (オプション – ただし、機能は省略できます)
 - iso読み取り (オプション。ただし、機能は省略できます)
 - zip (オプション。ただし、使用しなくても機能を縮小可能)
 - unzip (オプション。ただし、使用せずに機能を縮小できます)
-



注：ネイティブのLinuxシステムでは、すべての依存関係が満たされていません。次のコマンド (sudoを使用する場合もあります) を実行することで、サポートされているディストリビューションにツールの依存関係をインストールできます。

```
./setup/dep_dependency.sh
```

また、次のPython (>= 3.6)モジュールも必要です。

- データクラス

- defusedxml
- distutils
- 包装
- rpm
- ヤマル

Linuxホスト上でネイティブに動作させるために、次のディストリビューション、特にこのシナリオ用のDebianがテストされています。

- アルマLinux 8
- フェドラ34
- Debian 11.2

準備

必要なオプションパッケージとその理由を最初に確認する必要があります。不要なパッケージや過剰なパッケージをインストールすると、ディスク領域使用率の問題やインストール中の障害が発生する可能性があります。GISOを作成する前に、各プラットフォームの前提条件とディスク容量の要件を確認します。

必要なソフトウェアをダウンロードするには、ソフトウェアダウンロードの公式サイト([シスコソフトウェアダウンロード](#))を参照してください。

スクリプトは、iso、ブリッジ修正、SMUなど、さまざまなファイルを統合するのに役立ちます。

gisobuild.pyスクリプトをサーバー上の特定の場所にコピーするために必要です。スクリプトは[gisobuild Git Site](#)にあります。

GISOの作成

このスクリプトの使用に使用できる引数の概要を次に示します。

```
usage: gisobuild.py [-h] [--iso ISO] [--repo REPO [REPO ...]]
                  [--bridging-fixes BRIDGE_FIXES [BRIDGE_FIXES ...]]
                  [--xrconfig XRCONFIG] [--ztp-ini ZTP_INI] [--label LABEL]
                  [--no-label] [--out-directory OUT_DIRECTORY]
                  [--create-checksum] [--yamlfile CLI_YAML] [--clean]
                  [--pkglist PKGLIST [PKGLIST ...]] [--script SCRIPT]
                  [--docker] [--x86-only] [--migration] [--optimize]
                  [--full-iso]
                  [--remove-packages REMOVE_PACKAGES [REMOVE_PACKAGES ...]]
                  [--skip-usb-image] [--copy-dir COPY_DIRECTORY]
                  [--clear-bridging-fixes] [--verbose-dep-check] [--debug]
                  [--isoinfo ISOINFO] [--image-script IMAGE_SCRIPT]
                  [--version]
```

この例では、ASR 9901用のGISOが作成され、簡略化のためにISISおよびOSPFパッケージが使

用され、7.9.21バージョン用のミニとコンフィギュレーションファイルもGISOに追加されます。

次のコマンドで確認できるように、パッケージ、mini、およびconfigファイルは/srcディレクトリにコピーされ、スクリプト実行コマンドのクリアなバージョンを作成します。

```
root@debian:/gisobuild-master/src# ls
asr9k-9000v-nV-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-mcast-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-bng-ipoe-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-mgbl-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-bng-pppoe-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-mini-x64-7.9.21.iso
asr9k-bng-suppl-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-mp1s-te-rsvp-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-bng-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-mp1s-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-optic-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-eigrp-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-ospf-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-isis-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-services-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-li-x64-1.0.0.0-r7921.x86_64.rpm
asr9k-m2m-x64-1.0.0.0-r7921.x86_64.rpm
```

```
1nt
1ntmod
output_gisobuild
utils
validate
wrappers
exrmod
running-config-ASR9K
gisobuild.py
```

-----GISO CREATION-----

```
root@debian:/gisobuild-master/src# ./gisobuild.py --iso asr9k-mini-x64-7.9.21.iso --pkglist asr9k-isis-
System requirements check [PASS]
```

Platform: asr9k Version: 7.9.21

XR-Config file (/gisobuild-master/src/running-config-ASR9K) will be encapsulated in Golden ISO.
Warning: No RPMS or Optional Matching 7.9.21 packages found in repository

Building Golden ISO...

Summary

XR Config file:
router.cfg

...Golden ISO creation SUCCESS.

Golden ISO Image Location: /gisobuild-master/src/output_gisobuild/asr9k-golden-x64-7.9.21-firstGiso.iso

Creating USB Boot zip...

Skipping USB Boot Zip creation: Not supported for platform: asr9k

USB BOOT ZIP NEEDED?: Contact asr9k team to add support.

```
root@debian:/gisobuild-master/src#
```

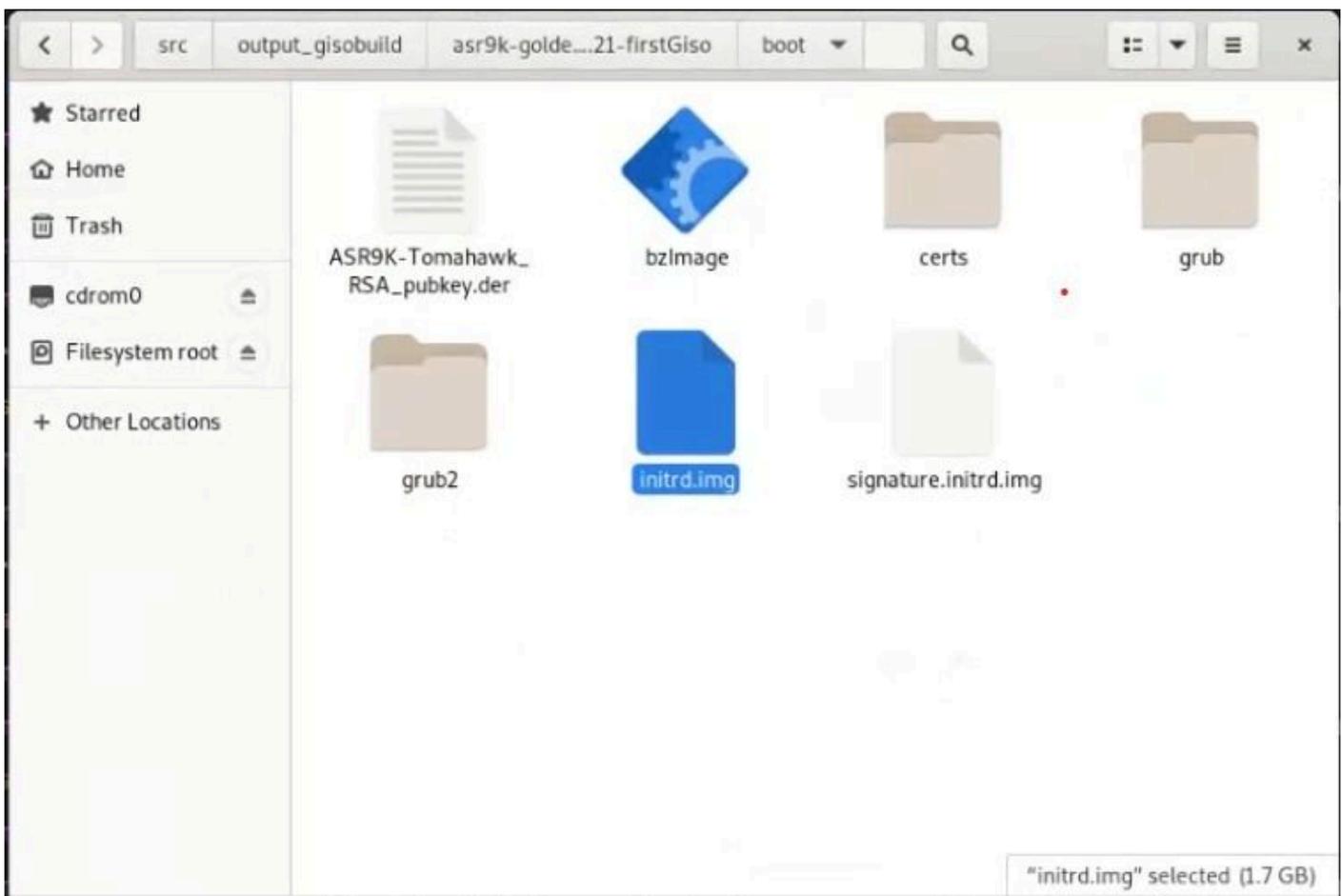
このGISO作成に使用されるパラメータの簡単な説明を次に示します。

- アイソメ図	Mini.iso/Full.isoファイルへのパス
-xrconfig XRCONFIG	XRコンフィギュレーションファイルへのパス
- ラベルLABEL、-iラベル	ゴールデンISOラベル
- クリーン	続行する前に出力ディレクトリを削除してください
- イメージをスキップする	USBイメージを構築しない (ASR9Kプラットフォームではサポートされない)

確認

インストールログに記載されているように、この例の場所を確認することで、GISOの作成を確認できます。ゴールデンISOイメージの場所： /gisobuild-master/src/output_gisobuild/asr9k-golden-x64-7.9.21-firstGiso.iso

次のイメージに示すように、initrd.img は1.7 GBで、ここに含めたインストールパッケージが含まれています。



翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。