



ゴールデン ISO ワークフロー

次の図は、ゴールデン ISO を構築してインストールするためのワークフローを示しています。

- [ゴールデン ISO の構築 \(1 ページ\)](#)
- [ゴールデン ISO のインストール \(4 ページ\)](#)
- [ゴールデン ISO への置換のインストール \(5 ページ\)](#)

ゴールデン ISO の構築

カスタマイズした ISO は、github の場所 (<https://github.com/ios-xr/gisobuild>) で利用可能なシスコ ゴールデン ISO (GISO) 作成スクリプト `gisobuild.py` を使用して構築します。

GISO 作成スクリプトは、自動依存関係管理をサポートし、次の機能を提供します。

- パッケージリポジトリ内に存在するすべてのパッケージの RPM データベースを構築します。
- `mini-x.iso` バージョンと一致しない Cisco RPM をスキップおよび削除します。
- `mini-x.iso` 内にすでに存在するサードパーティ製の基本パッケージの SMU ではないサードパーティの RPM をスキップおよび削除します。
- 同じリリースで異なるバージョンの基本 RPM が複数ある場合、エラーを表示し、作成プロセスを終了します。
- すべての RPM の互換性チェックと依存関係チェックを実行します。たとえば、子 RPM は親 RPM に依存します。子 RPM のみが含まれる場合、ゴールデン ISO の作成は失敗します。

GISO を作成するには、スクリプトに次の入力パラメータを指定します。

- 基本 `mini-x.iso` (必須)
- XR コンフィギュレーション ファイル (任意)
- ホスト、XR、およびシステム管理用の 1 つまたは複数のシスコ固有の SMU (必須)
- ホスト、XR、およびシステム管理用の 1 つまたは複数のサードパーティ SMU (必須)

- ゴールデン ISO のラベル (任意)



(注) ゴールデン ISO はミニ ISO からのみ作成できます。full または fullk9 バンドル ISO はサポートされていません。

GISO を作成する場合は、次の命名規則を使用します。

GISO ビルド	書式	例
k9sec RPM を使用しない GISO	$\langle \text{platform-name} \rangle \text{-golden-x.iso-}\langle \text{version} \rangle .\langle \text{label} \rangle$ $\langle \text{platform-name} \rangle \text{-golden-x-}\langle \text{version} \rangle .\text{iso}.\langle \text{label} \rangle$	<プラットフォーム名 >-golden-x64.iso-<バージョン>.v1 <プラットフォーム名 >-golden-x64-<バージョン>.iso.v1
k9sec RPM を使用した GISO	$\langle \text{platform-name} \rangle \text{-goldenk9-x.iso-}\langle \text{version} \rangle .\langle \text{label} \rangle$ $\langle \text{platform-name} \rangle \text{-goldenk9-x-}\langle \text{version} \rangle .\text{iso}.\langle \text{label} \rangle$	<プラットフォーム名 >-goldenk9-x64.iso-<バージョン>.v1 <プラットフォーム名 >-goldenk9-x64-<バージョン>.iso.v1



(注) k9sec RPM を GISO に適切に追加するには、**chmod** コマンドを使用してファイルの権限を 644 に変更します。

```
chmod 644 [k9 sec rpm]
```

GISO を作成するには、次の手順を実行します。

始める前に

- 非 GISO から GISO バージョンにアップグレードするには、最初に GISO サポートを使用してミニ ISO にアップグレードする必要があります。
- GISO が構築されているシステムは、次の要件を満たしている必要があります。
 - システムには Python バージョン 2.7 以降が必要です。
 - システムには、最低 3 ~ 4 GB の空きディスク領域が必要です。
 - システムに Linux ユーティリティ mount、rm、cp、umount、zcat、chroot、mkisofs があることを確認します。これらのユーティリティはスクリプトによって使用されます。これらすべての Linux コマンドを実行する権限があることを確認します。
 - システムのカーネルバージョンは、Cisco ISO のカーネルバージョンより後の 3.16 以降である必要があります。

- Linux カーネルでサポートされている `libyaml rpm` が、ツールで `yaml` を正常に実行できることを確認します。
- ユーザは `rpm` リポジトリのセキュリティ `rpm (k9sec-rpm)` に対する適切な権限を持っている必要があります。それ以外の場合は、ゴールデン ISO の作成でセキュリティ `rpm` が無視されます。
- `gisobuild` スクリプトが実行されるシステムには、`root` クレデンシャルを使用する必要があります。

手順

- ステップ 1** GISO を作成するオフライン システムまたは外部サーバに `github` の場所 (<https://github.com/ios-xr/gisobuild>) からスクリプト `gisobuild.py` をコピーします。このシステムが上記の「はじめる前に」セクションに記載された前提条件を満たしていることを確認します。
- ステップ 2** スクリプト `gisobuild.py` を実行し、ルータからゴールデン ISO を作成するためのパラメータを指定します。すべての RPM と SMU が同じディレクトリ内に存在することを確認します。ゴールデン ISO の作成に使用できる RPM と SMU の数は 128 です。

(注) `-i` オプションは必須で、`-r` と `-c` のいずれかまたはその両方を指定する必要があります。

```
[directory-path]$ gisobuild.py [-h] [-i <mini-x.iso>] [-r <rpm repository>]
[-c <config-file>] [-l <giso label>] [-m] [-v]
```

次に、スクリプトの出力例を示します。

値は次のとおりです。

- `-i` は `mini-x.iso` へのパスです
- `-r` は RPM リポジトリへのパスです
- `-c` は XR config ファイルへのパスです
- `-l` はゴールデン ISO ラベルです
- `-h` はヘルプ メッセージを表示します
- `-v` は、作成ツール `gisobuild.py` のバージョンです
- `-m` は、IOS XR から IOS XR 64 ビットに移行するための移行 tar を構築します

GISO は、指定されたディレクトリ内の各フォルダに配置された RPM を使用して作成され、ログファイル `giso_summary.txt` および `gisobuild.log-<タイムスタンプ>` も含まれています。XR コンフィギュレーション ファイルはディレクトリ内に `router.cfg` として格納されます。



(注) GISO スクリプトは XR 設定の検証をサポートしていません。

次のタスク

ゴールデン ISO をルータにインストールします。

ゴールデン ISO のインストール

ゴールデン ISO (GISO) は、次のアクションを自動的に実行します。

- ホストおよびシステム管理 RPM をインストールします。
- RP でリポジトリと TFTP ブートをパーティションに分割します。
- システム管理モードおよび XR モードでソフトウェア プロファイルを作成します。
- XR RPM をインストールします。 **show instal active** コマンドを使用して RPM のリストを表示します。
- XR 設定を適用します。 XR モードで **show running-config** コマンドを使用して確認します。

手順

ステップ 1 次のいずれかのオプションを使用して、ルータに GISO イメージをダウンロードします。

- **PXE ブート** : ルータが起動すると、ブートモードが識別されます。 PXE をブートモードとして検出すると、利用可能なすべてのイーサネットインターフェイスが起動し、各インターフェイスで DHCPclient が実行されます。 DHCPclient スクリプトは HTTP または TFTP プロトコルを解析し、GISO がボックスにダウンロードされます。
- **USB ブートまたはディスク ブート** : ブート中に USB モードが検出され、GISO が識別されると、追加の RPM および XR 設定ファイルが抽出されてインストールされます。
- システムのアップグレード時の **システムアップグレード** では、 **install add**、 **install activate**、または **install update** コマンドを使用して GISO をインストールできます。
 - **非 GISO (GISO をサポートしていないイメージ) から GISO イメージへのシステムアップグレード** : システムが GISO をサポートしていないイメージを使用してバージョン 1 を実行している場合、システムは GISO をサポートするイメージのバージョン 2 に直接アップグレードすることはできません。その代わりに、バージョン 1 をバージョン 2 ミニ ISO にアップグレードし、次にバージョン 2 GISO にアップグレードする必要があります。
 - **バージョン 1 GISO からバージョン 2 GISO へのリリースでのシステムアップグレード** : 両方の GISO イメージの基本バージョンは同じでラベルが異なる場合、 **install add** および **install activate** コマンドは同じバージョンの 2 つのイメージをサポートしませ

ん。その代わりに、**install update** コマンドを使用してデルタ RPM のみをインストールします。システムのリロードはデルタ RPM の再起動タイプに基づいています。

- **バージョン1 GISO からバージョン2 GISO へのリリース間でのシステムアップグレード**：両方の GISO イメージの基本バージョンが異なります。**install add** および **install activate** コマンド、または **install update** コマンドを使用して、システムアップグレードを実行します。ルータは、バージョン2 GISO イメージを使用したアップグレード後にリロードされます。

ステップ2 システム管理モードで **show install repository all** コマンドを実行し、ホスト、システム管理、および XR の RPM と基本 ISO を表示します。

ステップ3 **show install package <golden-iso>** コマンドを実行し、RPM のリストおよび GISO に組み込まれているパッケージを表示します。

GISO 内の ISO、SMU、およびパッケージがルータにインストールされます。

ゴールデン ISO への置換のインストール

ゴールデン ISO (GISO) は、単一の操作でソフトウェアメンテナンスアップデート (SMU) の事前定義されたリストを持つバージョンにルータをアップグレードします。ただし、異なる SMU セットを使用した同じバージョンに更新するには、2段階のプロセスが必要です。このプロセスでは、GISO をアップグレードしてデルタ SMU を追加し、使用されていない SMU を手動で非アクティブ化する必要があります。

この2段階のプロセスを回避するには、コマンドを使用して、現在アクティブなバージョンを、新しく追加した GISO のイメージを含む完全なパッケージに置き換えます。



(注) **install updatereplace** キーワードは GISO でのみサポートされています。**.mini** および **.rpm** パッケージでは直接サポートされていません。

手順

ステップ1

例：

```
RP/0/RP0/CPU0:ios#install harddisk:/misc/disk1/<giso-image>.iso
noprmt
+++++
Install operation 11 started by root:
exec-timeout is suspended.
No install operation in progress at this moment
Label = More_Pkgs
ISO <giso-iso-image>.iso in input package list. Going to upgrade the system to
```

```

version <new-giso-image>.
System is in committed state
Current full-label: <giso-image>_R_Commit
Current only-label: R_Commit
Current label: R_Commit
Updating contents of golden ISO
Scheme : localdisk
Hostname : localhost
Username : None
SourceDir : /ws
Collecting software state..
Getting platform
Getting supported architecture
Getting active packages from XR
Getting inactive packages from XR
Getting list of RPMs in local repo
Getting list of provides of all active packages
Getting provides of each rpm in repo
Getting requires of each rpm in repo
Fetching .... <giso-image>.iso
Label within GISO: More_Pkgs
Skipping <platform>-mgb1-3.0.0.0-<release>.x86_64.rpm from GISO as it's active
Adding packages
    <platform>-golden-x-<release>-<Label>.iso
RP/0/RP0/CPU0:Jun 20 14:43:59.349 UTC: sdr_instmgr[1164]:
%INSTALL-INSTMGR-2-OPERATION_SUCCESS :

Install operation 12 finished successfully
Install add operation successful
Activating <platform>-golden-x-<release>-<Label>
Jun 20 14:44:05 Install operation 13 started by root:
    install activate pkg <platform>-golden-x-<release>-<Label> replace noprompt
Jun 20 14:44:05 Package list:
Jun 20 14:44:05     <platform>-golden-x-<release>-<Label>.iso
Jun 20 14:44:29 Install operation will continue in the background
exec-timeout is resumed.
RP/0/RP0/CPU0:ios#Jun 20 14:51:01 Install operation 13 finished successfully
RP/0/RP0/CPU0:Jun 20 14:51:01.416 UTC: sdr_instmgr[1164]:
%INSTALL-INSTMGR-2-OPERATION_SUCCESS :

Install operation 13 finished successfully
RP/0/RP0/CPU0:Jun 20 14:51:01.417 UTC: sdr_instmgr[1164]:
%INSTALL-INSTMGR-2-SYSTEM_RELOAD_INFO :
```

新しく追加された GISO のバージョンおよびラベルは、現在アクティブなバージョンのバージョンおよびラベルと比較されます。不一致が特定されると、新しいパーティションが作成され、完全なパッケージがインストールされます。インストール後、システムは新しく追加された GISO からイメージおよびパッケージをリロードします。

- (注) 有効なラベルを持つシステムでアクティブ化または非アクティブ化すると、ラベルが無効になります。この操作は元に戻せません。たとえば、システムで **show version** コマンドを実行すると、ラベル 6.3.3.15I_633rev1005 が表示されます。システムで SMU がアクティブ化または非アクティブ化になると、ラベル 633rev1005 は無効になり、show version コマンドはラベルとして 6.3.3.15I のみを表示します。

ステップ 2 show version

例：

```
RP/0/RP0/CPU0:ios#show version
Wed Jun 20 15:06:37.915 UTC
Cisco IOS XR Software, Version <new-giso-image>
Copyright (c) 2013-2018 by Cisco Systems, Inc.
```

```
Build Information:
Build By      : <user>
Build On     : <date>
Build Host   : <host-name>
Workspace    : <workspace-name>
Version      : <version>
Location     : <path>
Label       : <label-name>
```

```
cisco <platform> () processor
System uptime is 3 hours 51 minutes
```

システムは新しく追加された GISO からイメージおよびパッケージをリロードします。

