

Crear archivo GISO para actualización de IOS-XR con Python 3

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Productos Relacionados](#)

[ISO dorado para actualizaciones de XR](#)

[Herramienta Gisobuild Python](#)

[Preparación](#)

[Creación de GISO](#)

[Verificación](#)

Introducción

Este documento describe la creación de Golden ISO (GISO) para actualizaciones en routers que ejecutan el software Cisco IOS® XR en versiones eXR.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Software Cisco IOS XR
- Procedimientos de instalación y actualización del software Cisco IOS XR
- Comandos básicos de Linux y navegación de la línea de comandos

Componentes Utilizados

Este documento no está restringido a versiones específicas de hardware, este documento se aplica a todos los routers que ejecutan IOS XR de 64 bits.

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Productos Relacionados

Este documento también se puede utilizar con estas versiones de hardware:

- Cisco 9000 Series Routers
- Cisco 8000 Series Routers
- Routers de la serie Cisco NCS 5700
- Routers de la serie Cisco NCS 5500
- Routers NCS 540 y 560

ISO dorado para actualizaciones de XR

Normalmente, Cisco lanza el software IOS-XR como un mini/base ISO que contiene paquetes IOS-XR obligatorios para una plataforma determinada, un conjunto de paquetes opcionales para ejecutar funciones adicionales y parches de software para cualquier corrección de errores en forma de actualización de mantenimiento de software (SMU). El paquete opcional y SMU están en formato de empaquetado RPM.

La herramienta Golden ISO crea una ISO que contiene todo el contenido de la mini/base ISO junto con paquetes opcionales y SMU de la elección del usuario. Una vez que se crea el ISO dorado, se puede utilizar para el arranque iPXE o para su (actualización del sistema) de la versión actual en ejecución a una nueva versión de IOS-XR.



Nota: La ISO mini/base es obligatoria para cualquier creación de ISO dorado

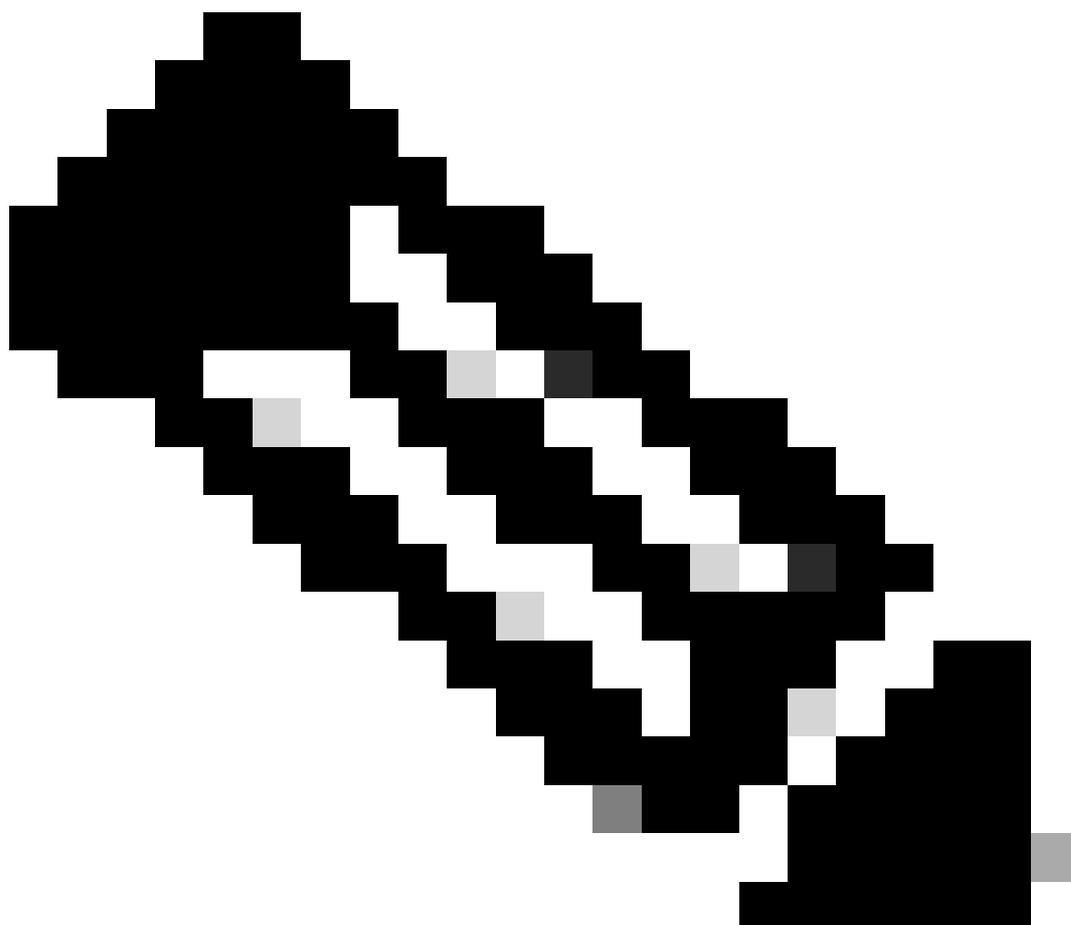
Herramienta Gisobuild Python

Esta herramienta se puede ejecutar de forma nativa en un host Linux. Alternativamente, la herramienta también se puede ejecutar en un sistema Linux con Docker habilitado y la capacidad de extraer la imagen publicada 'cisco-xr-gisobuild' de Docker Hub, para este ejemplo se sugiere trabajar en Debian 11.8

Esta herramienta tiene los siguientes requisitos ejecutables:

- Python3 >= 3,6
- rpm >= 4,14
- cpio >= 2.10
- gzip >= 1,9
- createrepo_c
- archivo

- isoinfo
 - mkisofs
 - mksquashfs
 - openssl
 - unsquashfs
 - 7z (opcional, pero la funcionalidad se puede reducir sin él)
 - iso-read (opcional, pero la funcionalidad se puede reducir sin él)
 - zip (opcional, pero la funcionalidad se puede reducir sin)
 - unzip (opcional, pero la funcionalidad se puede reducir sin)
-



Nota: En un sistema Linux nativo, que no tiene todas las dependencias cumplidas, las dependencias de la herramienta se pueden instalar en las distribuciones soportadas ejecutando el siguiente comando (posiblemente a través de sudo):

```
./setup/prep_dependency.sh
```

También requiere los siguientes módulos de Python (>= 3.6):

- clases de datos
- defusedxml
- distutils
- embalaje
- rpm
- yaml

Para ejecutarse de forma nativa en un host Linux, se han probado las siguientes distribuciones, específicamente para este escenario, Debian.

- Alma Linux 8
- Fedora 34
- Debian 1.2

Preparación

Es imperativo confirmar primero qué paquetes opcionales son necesarios y por qué. La instalación de paquetes innecesarios o excesivos puede producir problemas de utilización del espacio en disco y fallas durante la instalación. Confirme los requisitos previos de cada plataforma y los requisitos de espacio en disco antes de crear el GISO.

Para descargar el software necesario, consulte el sitio oficial de descarga de software: [Descarga de software de Cisco](#)

La secuencia de comandos puede ayudarnos a unificar una gran variedad de archivos, por ejemplo, iso, correcciones de puente, SMU, etc.

Es necesario copiar el script gisobuild.py a una ubicación particular en el servidor. El script se encuentra en [gisobuild Git Site](#)

Creación de GISO

A continuación se muestra un resumen de los argumentos que podemos utilizar para este uso del script:

```
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]
```

Para este ejemplo, se crea un GISO para ASR 9901, se utilizan los paquetes ISIS y OSPF para la brevedad, mini para la versión 7.9.21 y el archivo de configuración también se agregan al GISO.

Como se ve en el siguiente comando, los paquetes, el archivo mini y el archivo config se copian en el directorio /src, esto para crear una versión borrada del comando script run.

```
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
Tnt
Tntmod
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#
```

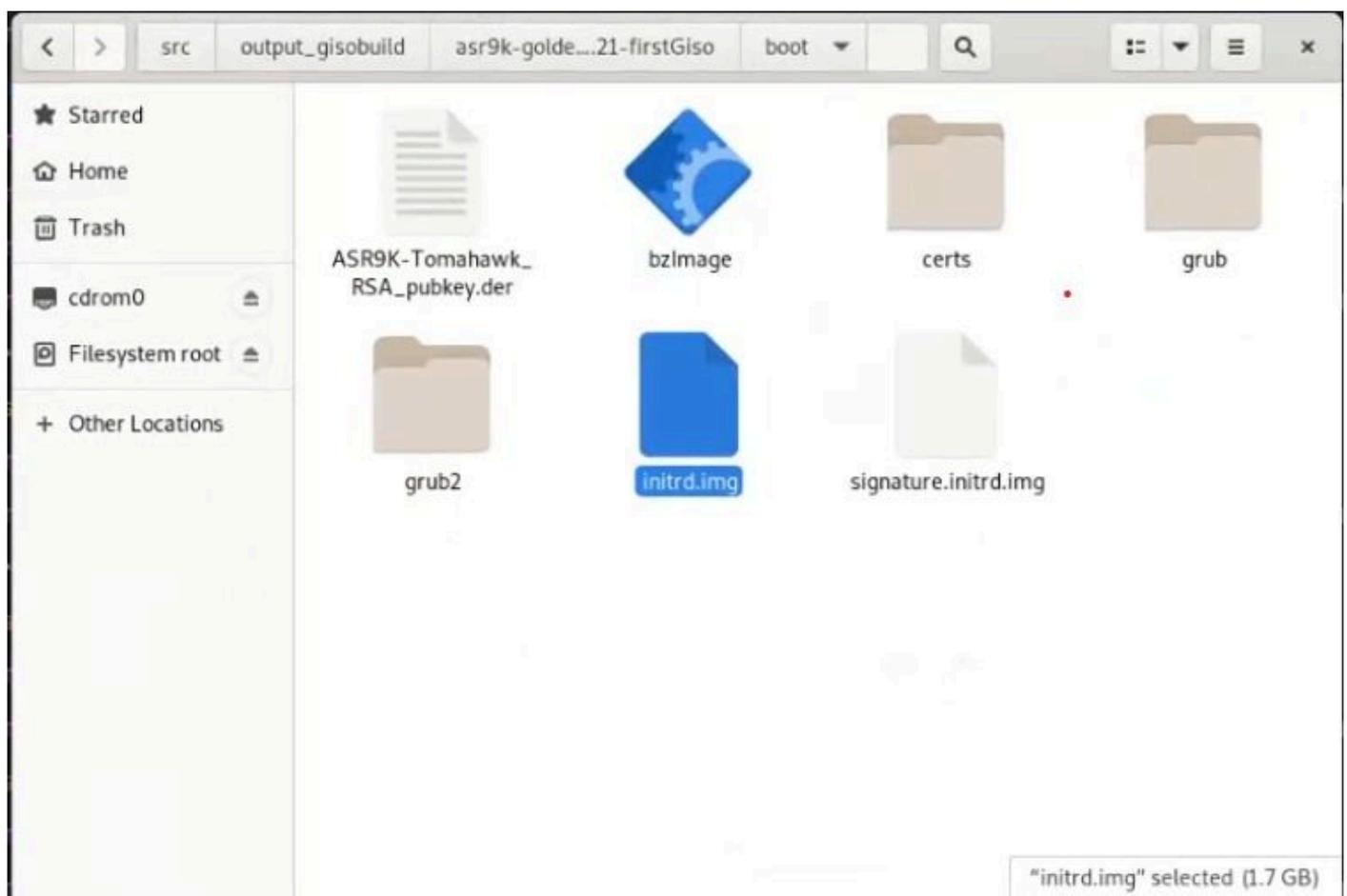
Aquí hay una breve descripción de los parámetros utilizados para esta creación de GISO:

: ISO ISO ISO	Ruta al archivo Mini.iso/Full.iso
—xrconfig XRCONFIG	Ruta al archivo de configuración XR
—etiqueta ETIQUETA, -I ETIQUETA	Etiqueta ISO dorada
--clean	Eliminar dir de salida antes de continuar
—skip-image	No cree la imagen USB (no compatible con la plataforma ASR9K)

Verificación

Como mencionan los registros de instalación, podemos confirmar nuestra creación de GISO revisando la ubicación, para este ejemplo. Ubicación de la imagen ISO dorada: /gisobuild-master/src/output_gisobuild/asr9k-golden-x64-7.9.21-firstGiso.iso

Según la siguiente imagen, initrd.img es 1.7GB que contiene los paquetes de instalación que incluimos.



Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).