

Crea file GISO per aggiornamento IOS-XR con Python 3

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Prodotti correlati](#)

[Golden ISO per aggiornamenti XR](#)

[Strumento Python gisobuild](#)

[Preparazione](#)

[Creazione GISO](#)

[Verifica](#)

Introduzione

Questo documento descrive la creazione di Golden ISO (GISO) per gli aggiornamenti nei router con software Cisco IOS® XR nelle versioni eXR.

Prerequisiti

Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Software Cisco IOS XR
- Procedure di installazione e aggiornamento del software Cisco IOS XR
- Comandi di base di Linux e navigazione della riga di comando

Componenti usati

Il documento può essere consultato per tutte le versioni hardware; è valido per tutti i router che eseguono IOS XR a 64 bit.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Prodotti correlati

Questo documento può essere utilizzato anche con queste versioni hardware:

- Cisco serie 9000 Router
- Cisco serie 8000 Router
- Cisco NCS serie 5700 Router
- Cisco NCS serie 5500 Router
- NCS 540 e 560 Router

Golden ISO per aggiornamenti XR

In genere, Cisco rilascia il software IOS-XR come ISO mini/base che contiene i pacchetti IOS-XR obbligatori per una determinata piattaforma, un set di pacchetti opzionali per eseguire funzionalità aggiuntive e patch software per qualsiasi correzione di bug sotto forma di Software Maintenance Upgrade (SMU). Pacchetto opzionale e SMU sono in formato RPM.

Lo strumento Golden ISO crea un ISO contenente l'intero contenuto dell'ISO mini/base insieme a pacchetti opzionali e SMU a scelta dell'utente. Una volta creato, il Golden ISO può essere utilizzato sia per l'avvio iPXE sia per l'SU (aggiornamento del sistema) dalla versione corrente a una nuova versione di IOS-XR.



Nota: Mini/base ISO è obbligatorio per qualsiasi creazione di Golden ISO

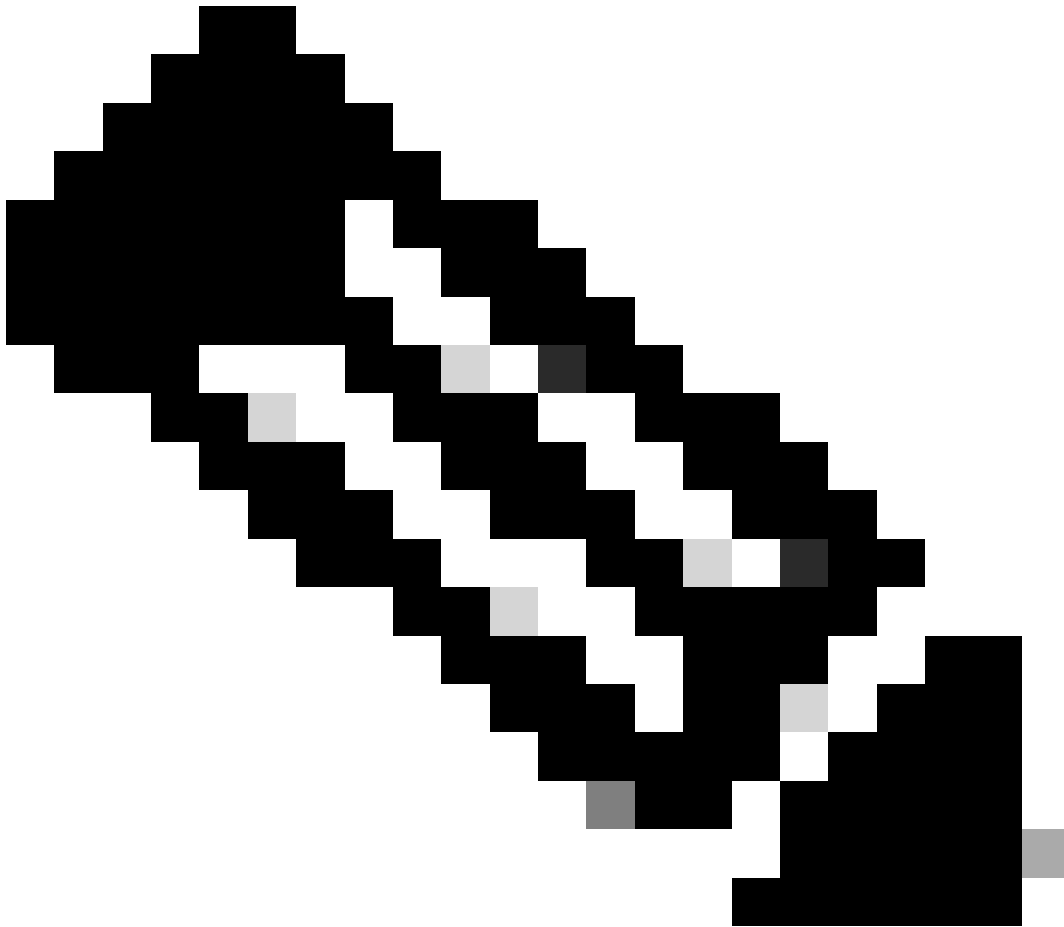
Strumento Python gisobuild

Questo strumento può essere eseguito in modo nativo su un host Linux. In alternativa, lo strumento può anche essere eseguito su un sistema Linux con Docker abilitato e la capacità di estrarre l'immagine 'cisco-xr-gisobuild' pubblicata da Docker Hub, per questo esempio si consiglia di lavorare in Debian 11.8

Questo strumento ha i requisiti di eseguibilità successivi:

- python3 >= 3,6
- rpm >= 4,14
- cpio >= 2,10
- gzip >= 1,9
- createrepo_c
- file

- isoinfo
 - mkisofs
 - mksquashfs
 - openssl
 - unsquashfs
 - 7z (opzionale, ma la funzionalità può essere ridotta senza)
 - lettura iso (opzionale, ma senza funzionalità ridotte)
 - zip (opzionale, ma la funzionalità può essere ridotta senza)
 - unzip (opzionale, ma la funzionalità può essere ridotta senza)
-



Nota: su un sistema Linux nativo, per il quale non sono state soddisfatte tutte le dipendenze, è possibile installare le dipendenze dello strumento nelle distribuzioni supportate eseguendo il comando successivo (possibilmente tramite sudo):

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

Richiede anche i successivi moduli Python (≥ 3.6):

- classi di dati
- defusedxml
- distutils
- imballaggio
- rpm
- yaml

Per l'esecuzione nativa su un host Linux, sono state testate le successive distribuzioni, in particolare per questo scenario, Debian.

- Alma Linux 8
- Fedora 34
- Debian 11.2

Preparazione

È essenziale prima confermare quali pacchetti opzionali sono necessari e perché. L'installazione di pacchetti inutili o eccessivi può causare problemi di utilizzo dello spazio su disco e guasti durante l'installazione. Confermare i prerequisiti e i requisiti di ciascuna piattaforma per lo spazio su disco prima di creare la GISO.

Per scaricare il software necessario, consultare il sito ufficiale per il download del software: [Cisco Software Download](#)

Gli script possono aiutarci a unificare una vasta gamma di file, ad esempio iso, correzioni di bridge, SMU e così via.

È necessario per copiare lo script gisobuild.py in una posizione specifica sul server. Lo script si trova nel [sito Git di gisobuild](#)

Creazione GISO

Di seguito è riportato un riepilogo degli argomenti che è possibile utilizzare per questo utilizzo dello 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]
```

Per questo esempio, viene creato un GISO per ASR 9901, i pacchetti ISIS e OSPF vengono utilizzati per brevità, i pacchetti mini per la versione 7.9.21 e il file di configurazione aggiunto anche al GISO.

Come illustrato nel comando successivo, i pacchetti, i file mini e di configurazione vengono copiati nella directory /src per creare una versione cancellata del comando di esecuzione dello script.

```
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#
```

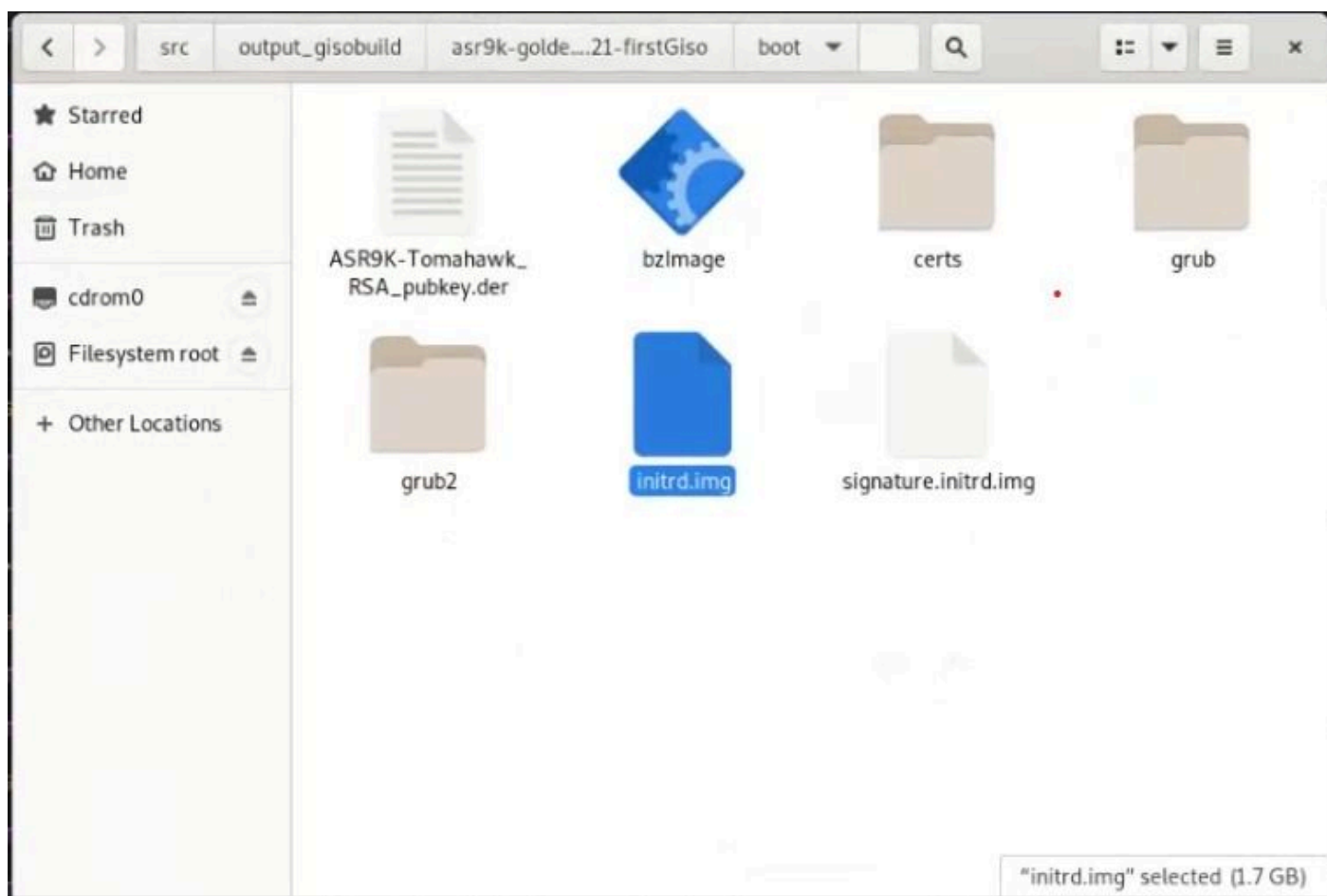
Di seguito è riportata una breve descrizione dei parametri utilizzati per la creazione di questo oggetto GISO:

—iso ISO	Percorso del file Mini.iso/Full.iso
—xrconfig XRCONFIG	Percorso del file di configurazione XR
—label ETICHETTA, -I ETICHETTA	Etichetta Golden ISO
--clean	Elimina directory di output prima di continuare
—ignora immagine	Non creare l'immagine USB (non supportata per la piattaforma ASR9K)

Verifica

Come accennato nei log di installazione, è possibile confermare la creazione del GISO esaminando il percorso, per questo esempio. Posizione immagine ISO dorata: /gisobuild-master/src/output_gisobuild/asr9k-golden-x64-7.9.21-firstGiso.iso

Come per l'immagine successiva, `initrd.img` è 1,7 GB contenente i pacchetti di installazione che abbiamo incluso.



Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).