

Executar o ISSU nos switches Catalyst 3850 e Catalyst 9000 Series

Contents

[Introdução](#)

[O que é ISSU](#)

[Plataformas suportadas e matriz de suporte da versão](#)

[Pré-requisitos antes do ISSU](#)

[1. Verificar Versão Atual do Código](#)

[2. Verifique o modo de inicialização](#)

[3. Verifique se Há Memória Disponível Suficiente no Flash](#)

[4. Verifique se os Switches estão no Modo SSO](#)

[5. Verifique se a Inicialização Automática está Habilitada](#)

[6. Verifique o ISSU atual e os estados de instalação](#)

[7. Copie a Nova Imagem a Ser Atualizada / Rebaixada](#)

[Fluxos de trabalho do ISSU - Atualização real](#)

[Fluxo de Trabalho de Uma Etapa](#)

[Fluxo de trabalho em três etapas](#)

[Lista de verificação pós-ISSU](#)

[Ação em caso de falha de ISSU](#)

[Abortar ISSU](#)

[Limpar estado de ISSU](#)

Introdução

Este documento descreve as etapas envolvidas na execução do In-Service Software Upgrade (ISSU) nos switches Catalyst 3850 e Catalyst 9000 Series.

O que é ISSU

O In-Service Software Upgrade é um processo que atualiza uma imagem para outra imagem em um dispositivo enquanto a rede continua a encaminhar pacotes. O ISSU ajuda os administradores de rede a evitarem uma interrupção de rede quando executam uma atualização de software. As imagens são atualizadas no modo de instalação, em que cada pacote é atualizado individualmente.

O ISSU é suportado em todas as séries Catalyst 3850 e Catalyst 9000 que suportam Stackwise-Virtual e também em chassis autônomos Catalyst 9400 / 9600 com supervisores duplos.


O StackWise Virtual (SVL) compreende dois switches que são conectados juntos para formar um switch virtual. O SVL suporta atualizações de software em serviço.

Plataformas suportadas e matriz de suporte da versão

Antes de continuar com o ISSU, verifique se a plataforma realmente suporta o ISSU. Além disso, verifique se o ISSU é suportado entre o código atual e o código de destino. Os detalhes do modo nas plataformas suportadas e na matriz de compatibilidade do ISSU podem ser encontrados aqui:

[Suporte ISSU entre versões](#)

Pré-requisitos antes do ISSU

 Note: O exemplo neste documento é baseado nos switches Cisco Catalyst 9500 configurados como Stackwise-Virtual. As etapas são aplicáveis ao chassi independente Cat9400 / Cat9600 com mecanismos de supervisão duplos e também para dispositivos Catalyst 3850 / Catalyst 9000 configurados como Stackwise-Virtual. Para evitar qualquer perda de pacotes durante o processo, o ISSU requer redundância nos nós para os dispositivos ativo e standby. Como ocorre uma recarga e um switchover entre eles é necessário para garantir uma alta disponibilidade na rede.

1. Verificar Versão Atual do Código

```
<#root>
```

```
C9500#show version | in IOS XE
```

```
Cisco IOS XE Software, Version 16.09.02
```

2. Verifique o modo de inicialização

O ISSU é suportado apenas se ambos os switches no Stackwise Virtual forem inicializados no modo de instalação.

```
<#root>
```

```
C9500#show ver | in INSTALL
```

```
* 1 50 C9500-40X 16.9.2 CAT9K_IOSXE INSTALL
  2 50 C9500-40X 16.9.2 CAT9K_IOSXE INSTALL
```

On Catalyst 9400, the above output is not available. Check if the switch booted from packages.conf file

```
C9400#show version | in System image
System image file is "flash:packages.conf"
```

Se o chassi for inicializado no modo Bundle, o ISSU não é suportado. Você pode ver erros como esse se tentar executar o ISSU quando os switches estiverem sendo executados no modo de pacote.

```
<#root>
```

```
*Nov 13 14:55:57.338: %INSTALL-5-INSTALL_START_INFO: Chassis 1 R1/0: install_engine: Started install on  
ERROR: install_add_activate_commit: One-Shot ISSU operation is
```

```
not supported in bundle boot mode
```

```
FAILED: install_add_activate_commit exit(1) Tue Nov 13 14:56:03 UTC 2018
```

3. Verifique se Há Memória Disponível Suficiente no Flash

```
<#root>
```

```
C9500#dir flash: | in free
```

```
11353194496 bytes total (8565174272 bytes free)
```

```
C9500#dir stby-flash: | in free
```

```
11353980928 bytes total (8566865920 bytes free)
```

Verifique se há pelo menos 1 GB de espaço na memória flash para expandir uma nova imagem. Limpe arquivos de instalação antigos em caso de espaço insuficiente e use o comando `install remove inative`.

4. Verifique se os Switches estão no Modo SSO

```
<#root>
```

```
C9500#show redundancy
```

```
Redundant System Information :
```

```
-----
```

```
    Available system uptime = 4 minutes
```

```
Switchovers system experienced = 0
```

```
    Standby failures = 0
```

```
    Last switchover reason = none
```

```
    Hardware Mode = Duplex
```

```
Configured Redundancy Mode = sso
```

Operating Redundancy Mode = sso

Maintenance Mode = Disabled
Communications = Up

Current Processor Information :

Active Location = slot 1
Current Software state = ACTIVE
Uptime in current state = 30 minutes
Image Version = Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_IOSXE),
Technical Support: <http://www.cisco.com/techsupport>
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Mon 05-Nov-18 19:32 by mcpre

BOOT = flash:packages.conf;

CONFIG_FILE =
Configuration register = 0x102

Peer Processor Information :

Standby Location = slot 2
Current Software state = STANDBY HOT
Uptime in current state = 26 minutes
Image Version = Cisco IOS Software [Fuji], Catalyst L3 Switch Software (CAT9K_IOSXE),
Technical Support: <http://www.cisco.com/techsupport>
Copyright (c) 1986-2018 by Cisco Systems, Inc.
Compiled Mon 05-Nov-18 19:32 by mcpre

BOOT = flash:packages.conf;

CONFIG_FILE =
Configuration register = 0x102

5. Verifique se a Inicialização Automática está Habilitada

<#root>

C9500#show boot system

switch 1

Current Boot Variables:
BOOT variable = flash:packages.conf;

Boot Variables on next reload:

```
BOOT variable = flash:packages.conf;
```

```
Manual Boot = no
```

```
<<<<< Manual Boot should be set to "no"
```

```
Enable Break = no
```

```
Boot Mode = DEVICE
```

```
iPXE Timeout = 0
```

```
-----
```

```
Switch 2
```

```
-----
```

```
Current Boot Variables:
```

```
BOOT variable = flash:packages.conf;
```

```
Boot Variables on next reload:
```

```
BOOT variable = flash:packages.conf;
```

```
Manual Boot = no
```

```
Enable Break = no
```

```
Boot Mode = DEVICE
```

```
iPXE Timeout = 0
```

Se a Inicialização automática não estiver habilitada, isso poderá ser alterado conforme mostrado:

```
<#root>
```

```
c9500(config)#no boot manual
```

6. Verifique o ISSU atual e os estados de instalação

Note: Este passo é muito importante!

<#root>

```
C9500#show issu state detail
```

```
--- Starting local lock acquisition on switch 1 ---  
Finished local lock acquisition on switch 1
```

```
No ISSU operation is in progress      <<<<<<<< If see anything else, abort ISSU before proceeding.
```

Check on how to manually abort ISSU.

```
C9500#show install summary
```

[Switch 1 2] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
 C - Activated & Committed, D - Deactivated & Uncommitted

Type St Filename/Version

IMG C 16.9.2.0.2433 <<<<<<<< State should be Activated & Committed for current version alone.

If not clear install state before proceeding. Check on how to clear install state.

Auto abort timer: inactive

7. Copie a Nova Imagem a Ser Atualizada / Rebaixada

É suficiente copiar a imagem somente para o chassi Ativo (no caso de Stackwise) ou para o supervisor Ativo (no caso de SUP duplo Cat9400).

<#root>

C9500#copy tftp: bootflash:

Address or name of remote host []? X.X.X.X
Source filename []? cat9k_iosxe.16.09.02.SPA.bin
Destination filename [cat9k_iosxe.16.09.02.SPA.bin]?

Fluxos de trabalho do ISSU - Atualização real

Depois que todos os pré-requisitos forem verificados, você poderá prosseguir com a atualização real e usar um dos métodos listados:

- Fluxo de Trabalho de Uma Etapa (apenas uma etapa e não suporta reversão)
- Fluxo de trabalho de três etapas (envolve três etapas e suporta reversão em caso de falha)

Fluxo de Trabalho de Uma Etapa

Esse fluxo de trabalho envolve apenas uma etapa e ajuda na otimização.

Note: Não é possível reverter, pois a atualização é confirmada automaticamente. Vá para o Fluxo de trabalho de 3 etapas se desejar reverter.

<#root>

```
// This example has SW-2 as Active and Sw-1 as Standby before starting ISSU
```

```
C9500#install add file flash:cat9k_iosxe.16.09.02.SPA.bin activate issu commit
```

```
install_add_activate_commit: START Fri Feb 8 10:07:51 jst 2019
```

```
*Feb 8 10:07:52.456 jst: %INSTALL-5-INSTALL_START_INFO: Switch 2 R0/0: install_engine: Started install
```

```
--- Starting initial file syncing ---
```

```
[2]: Copying flash:cat9k_iosxe.16.09.02.SPA.bin from switch 2 to switch 1
```

```
[1]: Finished copying to switch 1
```

```
Info: Finished copying flash:cat9k_iosxe.16.09.02.SPA.bin to the selected switch(es)
```

```
Finished initial file syncing
```



```
--- Starting Add ---
Performing Add on all members
  [1] Add package(s) on switch 1
  [1] Finished Add on switch 1
  [2] Add package(s) on switch 2
  [2] Finished Add on switch 2
Checking status of Add on [1 2]
Add: Passed on [1 2]
Finished Add

install_add_activate_commit: Activating ISSU
```

Going to start Oneshot ISSU install process

STAGE 0: Initial System Level Sanity Check before starting ISSU

```
=====
--- Verifying install_issu supported ---
--- Verifying standby is in Standby Hot state ---
--- Verifying booted from the valid media ---
--- Verifying AutoBoot mode is enabled ---
Finished Initial System Level Sanity Check
```

STAGE 1: Installing software on Standby

```
=====
--- Starting install_remote ---
Performing install_remote on Chassis remote
  [1] install_remote package(s) on switch 1
  [1] Finished install_remote on switch 1
install_remote: Passed on [1]
Finished install_remote
```

STAGE 2: Restarting Standby

```
=====
--- Starting standby reload ---
Finished standby reload

--- Starting wait for Standby to reach terminal redundancy state ---

<<<<< Standby (Sw-1) reloads here!!!
```

<<<<<<< After Standby (Sw-1) comes up >>>>>>>>>>

```
*Feb  8 10:19:10.223 jst: %REDUNDANCY-3-IPC: IOS versions do not match.
*Feb  8 10:19:48.421 jst: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Feb  8 10:19:49.422 jst: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO) <<<<< Standby (Sw
*Feb  8 10:21:02.975 jst: %PLATFORM-6-HASTATUS_DETAIL: RP switchover, received chassis event became act
```

```
<<<<<<< After new Standby (Sw-2) comes up >>>>>>>
```

```
*Feb 8 10:27:09.715 jst: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
```

```
*Feb 8 10:27:10.717 jst: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SS0).
```

```
<<<< ISSU commit starts after this automatically..
```

```
*Feb 8 10:28:27.302 jst: %INSTALL-5-INSTALL_START_INFO: Switch 2 R0/0: install_engine: Started install
```

```
%IOSXEBOOT-4-ISSU_ONE_SHOT: (rp/0): ISSU finished successfully
```

```
*Feb 8 10:29:32.127 jst: %INSTALL-5-INSTALL_COMPLETED_INFO: Switch 2 R0/0: install_engine: Completed i
```

Prossiga para depois que o ISSU for concluído.

Fluxo de trabalho em três etapas

- Esse fluxo de trabalho envolve três etapas: adicionar, ativar e confirmar. Após a ativação, todos os switches são atualizados para a nova versão do software, exceto que o software não é confirmado automaticamente, mas deve ser executado manualmente através do comando `install commit`.
- A vantagem dessa abordagem é que o sistema pode ser revertido para uma versão anterior do software.
- O sistema reverte automaticamente se o temporizador de reversão não for parado e usa o comando `install autabort-timer stop` ou `install commit`. Se o temporizador de reversão for interrompido, a nova versão do software poderá ser executada no dispositivo por qualquer duração e revertida para a versão anterior.

Etapa 1. Instale o add.

Esse comando faz o download da imagem no flash de inicialização e a expande em ambos os switches.

```
<#root>
```

```
// This example has SW-1 as Active and Sw-2 as Standby before starting ISSU
```

```
C9500#install add file flash:cat9k-universalk9.SPA.16.09.03.BETA.E1.SSA.bin.bin  
install_add: START Fri Feb 8 09:22:00 jst 2019
```

```
*Feb 8 09:22:02.055 jst: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine: Started install
```

```

--- Starting initial file syncing ---
[1]: Copying flash:cat9k-universalk9.SPA.16.09.03.BETA.E1.SSA.bin.bin from switch 1 to switch 2
[2]: Finished copying to switch 2
Info: Finished copying flash:cat9k-universalk9.SPA.16.09.03.BETA.E1.SSA.bin.bin to the selected switch(
Finished initial file syncing

--- Starting Add ---
Performing Add on all members
  [1] Add package(s) on switch 1
  [1] Finished Add on switch 1
  [2] Add package(s) on switch 2
  [2] Finished Add on switch 2
Checking status of Add on [1 2]
Add: Passed on [1 2]
Finished Add

SUCCESS: install_add Fri Feb  8 09:26:26 jst 2019 <<<< Wait until install_add says SUCCESS. If fails,

```

Etapa 2. Instale o ativate.

- Quando você executa este comando, esta sequência de eventos ocorre:
 - (i) Um temporizador de reversão é iniciado. Se o temporizador de reversão expirar, o sistema voltará ao mesmo estado antes do início do ISSU. O temporizador de reversão pode ser parado quando você usa o comando `install autosabort-timer stop`. O ISSU pode ser revertido quando você usa o comando `install abort issu`.
 - (ii) O switch em standby é provisionado com o novo software e é recarregado com a nova versão do software. Em seguida, o switch ativo é provisionado com o novo software e é recarregado. O switch em standby com a nova imagem agora se torna o switch ativo e o switch ativo antigo se torna o switch em standby.

Ao final desse procedimento, ambos os switches são executados com a nova imagem de software.

<#root>

```
C9500#install activate issu
```

```
install_activate: START Fri Feb  8 09:28:27 jst 2019
install_activate: Activating ISSU
```

```
*Feb  8 09:28:28.905 jst: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine: Started install
Going to start Activate ISSU install process
```

```
STAGE 0: Initial System Level Sanity Check before starting ISSU=====
--- Verifying install_issu supported ---
--- Verifying standby is in Standby Hot state ---
--- Verifying booted from the valid media ---
--- Verifying AutoBoot mode is enabled ---
Finished Initial System Level Sanity Check
```

```
STAGE 1: Installing software on Standby
```

=====

--- Starting install_remote ---
Performing install_remote on Chassis remote

*Feb 8 09:28:31.880 jst: %INSTALL-5-INSTALL_AUTO_ABORT_TIMER_PROGRESS: Switch 1 R0/0: rollback_timer: 0

[2] install_remote package(s) on switch 2
[2] Finished install_remote on switch 2
install_remote: Passed on [2]
Finished install_remote

STAGE 2: Restarting Standby

=====

--- Starting standby reload ---
Finished standby reload--- Starting wait for Standby to reach terminal redundancy state ---

<<<<<<< Standby (Sw-2) reloads here!!!*Feb 8 09:35:16.489 jst: %REDUNDANCY-3-IPC: IOS versions do not match

*Feb 8 09:36:00.238 jst: %HA_CONFIG_SYNC-6-BULK_CFGSYNC_SUCCEED: Bulk Sync succeeded
*Feb 8 09:36:01.240 jst: %RF-5-RF_TERMINAL_STATE: Terminal state reached for (SSO)

<<<< At this point, Standby (Sw-2) comes up with new code and joins as Hot Standby
Finished wait for Standby to reach terminal redundancy state

STAGE 3: Installing software on Active

=====

--- Starting install_active ---

Performing install_active on Chassis 1] install_active package(s) on switch 1
[1] Finished install_active on switch 1
install_active: Passed on [1]
Finished install_active
Chassis 1 reloading, reason - Non participant detected

STAGE 4: Restarting Active (switchover to standby) <<<<<<< At this point, there is a switchover and standby is active

=====

--- Starting active reload ---
New software can load after reboot process is completed
SUCCESS: install_activate Fri Feb 8 09:37:14 jst 2019

No final do estado Ativar, verifique os estados de ISSU.

<#root>

C9500#show issu state detail

--- Starting local lock acquisition on switch 2 ---
Finished local lock acquisition on switch 2

Operation type: Step-by-step ISSU
Install type : Image installation using ISSU
Current state : Activated state
Last operation: Switchover

Completed operations:

Operation	Start time
-----	-----
Activate location standby Chassis 2	2019-02-08:09:28:32
Activate location active Chassis 1	2019-02-08:09:36:03
Switchover	2019-02-08:09:37:16

State transition: Added -> Standby activated -> Active switched-over

Auto abort timer: automatic, remaining time before rollback: 01:43:55

Running image: flash:packages.conf

Operating mode: sso, terminal state reached

<<<<< Wait until SSO terminal state before proceeding to commit.

Etapa 3. Instale o commit.

O comando commit executa a limpeza necessária, ativa o novo software como permanente (remove a versão mais antiga do software) e interrompe o temporizador de reversão. Qualquer reinicialização após a confirmação pode ser inicializada com o novo software.

<#root>

C9500#install commit

install_commit: START Fri Feb 8 09:45:22 jst 2019

install_commit: Committing ISSU

*Feb 8 09:45:23.533 jst: %INSTALL-5-INSTALL_START_INFO: Switch 2 R0/0: install_engine: Started install

Going to start Commit ISSU install process

STAGE 0: Initial System Level Sanity Check before starting ISSU

=====

--- Verifying install_issu supported ---

--- Verifying standby is in Standby Hot state ---

--- Verifying booted from the valid media ---

--- Verifying AutoBoot mode is enabled ---

Finished Initial System Level Sanity Check

--- Starting install_commit_2 ---

Performing install_commit_2 on Chassis 2

[2] install_commit_2 package(s) on switch 2

[2] Finished install_commit_2 on switch 2

install_commit_2: Passed on [2]

Finished install_commit_2

STAGE 1: Dispatching the commit command to remote

=====

--- Starting install_commit_remote ---

Performing install_commit_remote on Chassis 1

Feb 8 09:48:33.364: %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started install commit

*Feb 8 09:48:33.352 jst: %INSTALL-5-INSTALL_START_INFO: Switch 1 R0/0: install_engine: Started install

Feb 8 09:51:27.505: %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine: Completed install commit

[1] install_commit_remote package(s) on switch 1

[1] Finished install_commit_remote on switch 1

install_commit_remote: Passed on [1]

Finished install_commit_remote

SUCCESS: install_commit Fri Feb 8 09:51:27 jst 2019

<<<<< ISSU is completed here!!!!

Lista de verificação pós-ISSU

Quando o ISSU for concluído com êxito,

- Verifique se ambos os switches são executados no novo software.
- Marque a saída show issu state detail para ser limpa e não mostrar nenhum ISSU em andamento.
- Verifique a saída show install issue history para garantir o êxito da operação do ISSU (Comando disponível somente na versão 16.10.1 e posterior).
- É recomendável dar tempo de imersão suficiente no novo software antes de habilitar qualquer novo recurso.

Ação em caso de falha de ISSU

- Se o ISSU falhar, espera-se que a anulação automática possa recuperar o estado inicial do sistema (imagem mais antiga). No entanto, se isso também falhar, é esperada a recuperação manual do chassi.
- Durante a recuperação manual, verifique se os modos ativo e standby executam a imagem mais antiga (caso contrário, recupere o chassi individual).
- Depois de garantir que ambos os chassis executam a imagem antiga, execute install remove inactive para remover qualquer pacote de imagem não utilizado.
- Quando os dois chassis executarem o software antigo, limpe manualmente todos os estados internos de operação do ISSU. (Consulte aqui sobre como limpar os estados internos da ISSU).

Abortar ISSU

- No fluxo de trabalho de 3 etapas, durante o processo de ativação do ISSU, se o temporizador de interrupção expirar, o sistema poderá cancelar automaticamente para uma imagem mais antiga.



Note: Se o standby não alcançar o SSO durante o abortamento, o abortamento manual será necessário. Além disso, se por qualquer motivo você quiser abortar o ISSU no meio, o abortamento manual é necessário.

<#root>

EXAMPLE : During install add, we notice these erro

rs:

```
C9400#install add file flash:cat9k_iosxe.16.09.02.SPA.bin
install_add: START Tue Nov 13 20:47:53 UTC 2018
```

```
*Nov 13 20:47:54.787: %INSTALL-5-INSTALL_START_INFO: Chassis 1 R1/0: install_engine: Started install ad
```

```
--- Starting initial file syncing ---
```

```
[1]: Copying flash:cat9k_iosxe.16.09.02.SPA.bin from chassis 1 to chassis 2
```

```
[2]: Finished copying to chassis 2
```

```
Info: Finished copying flash:cat9k_iosxe.16.09.02.SPA.bin to the selected chassis
```

```
Finished initial file syncing
```

```
--- Starting Add ---
Performing Add on all members
  [1] Add package(s) on chassis 1
  [1] Finished Add on chassis 1
  [2] Add package(s) on chassis 2
      cp: cannot stat '/tmp/packages.conf': No such file or directory
  [2] Finished Add on chassis 2
Checking status of Add on [1 2]
Add: Passed on [1]. Failed on [2]
Finished Add
```

```
FAILED: install_add exit(1) Tue Nov 13 20:51:58 UTC 2018 <<<<< install_add failed. If see any such e
```

```
C9400#install abort issu
```

```
install_abort: START Tue Nov 13 20:57:40 UTC 2018
install_abort: Abort type ISSU subtype NONE smutype NONE
```

```
*Nov 13 20:57:41.759: %INSTALL-5-INSTALL_START_INFO: Chassis 1 R1/0: install_engine: Started install ab
```

```
NOTE: Going to start Abort ISSU install process
```

```
STAGE 0: Initial System Level Sanity Check before starting ISSU
```

```
=====
```

```
--- Verifying install_issu supported ---
--- Verifying booted from the valid media ---
--- Verifying AutoBoot mode is enabled ---
Finished Initial System Level Sanity Check
```

```
FAILED: ABORT operation is not allowed in ADDED state
ERROR: install_abort exit(2 ) Tue Nov 13 20:57:49 UTC 2018
```

```
*Nov 13 20:57:49.756: %INSTALL-5-INSTALL_COMPLETED_INFO: Chassis 1 R1/0: install_engine:
```

```
Completed install abort ISSU
```

Limpar estado de ISSU

Se o upgrade/downgrade/abort/auto-abort do ISSU não for bem-sucedido, a limpeza manual dos estados internos do ISSU será necessária.

```
<#root>
```

```
C9400#sh issu state detail
```

```
--- Starting local lock acquisition on chassis 1 ---
Finished local lock acquisition on chassis 1
```

```
Operation type: One-shot ISSU
Install type : Image installation using ISSU
```

```
Current state : Added state
```

```
Last operation: Activate location standby Chassis 2 <<<< Previous Add is still pending. This needs to
```

```
Completed operations:
```


Operation	Start time

Activate location standby Chassis 2	2018-11-13:16:26:34

State transition: Added

Auto abort timer: inactive
Running image: flash:packages.conf
Operating mode: sso, terminal state not reached

Enable Service Internal before you run this command

C9400#clear install state

clear_install_state: START Tue Nov 13 17:05:47 UTC 2018
--- Starting clear_install_state ---
Performing clear_install_state on all members
 [1] clear_install_state package(s) on chassis 1
 [1] Finished clear_install_state on chassis 1
Checking status of clear_install_state on [1]
clear_install_state: Passed on [1]

Finished clear_install_state

C9400#sh issu state detail

--- Starting local lock acquisition on chassis 1 ---
Finished local lock acquisition on chassis 1

No ISSU operation is in progress

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.