Effectuer une mise à niveau pendant que l’appareil fonctionne sur les commutateurs Catalyst des séries 3850 et 9000

Table des matières

- Introduction
- Qu'est-ce que ISSU
- Plates-formes prises en charge et matrice de support des versions
- Prérequis avant ISSU
  - 1. Vérifier la version actuelle du code
  - 2. Vérifiez le mode de démarrage
  - 3. Vérifiez si la mémoire flash est suffisante
  - 4. Vérifiez si les commutateurs sont en mode SSO
  - 5. Vérifiez si le démarrage automatique est activé
  - 6. Vérifiez l'état actuel de l'ISSU et de l'installation
  - 7. Copier la nouvelle image à mettre à niveau / rétrograder
- Workflows ISSU - Mise à niveau réelle
  - Flux de travail en une étape
  - Processus en trois étapes
- Liste de contrôle Post ISSU
- Action en cas d'échec ISSU
  - Abandonner ISSU
  - État ISSU propre

Introduction

Ce document décrit les étapes nécessaires pour effectuer la mise à niveau pendant que l’appareil fonctionne (ISSU) sur les commutateurs Catalyst des séries 3850 et 9000.

Qu'est-ce que ISSU

La mise à niveau logicielle en service est un processus qui met à niveau une image vers une autre image sur un périphérique pendant que le réseau continue à transférer des paquets. ISSU aide les administrateurs réseau à éviter une panne réseau lorsqu'ils effectuent une mise à niveau logicielle. Les images sont mises à niveau en mode d'installation, chaque package étant mis à niveau individuellement.

ISSU est pris en charge sur toutes les gammes Catalyst 3850 et Catalyst 9000 qui prennent en charge Stackwise-Virtual, ainsi que sur les châssis autonomes Catalyst 9400 / 9600 avec deux superviseurs.

StackWise Virtual (SVL) comprend deux commutateurs qui sont connectés ensemble pour former un commutateur virtuel. SVL prend en charge les mises à niveau logicielles en service.

Plates-formes prises en charge et matrice de support des versions

Avant de continuer avec ISSU, vérifiez si la plate-forme prend réellement en charge ISSU. Vérifiez également si ISSU est pris en charge entre le code actuel et le code cible. Vous trouverez ici des informations détaillées sur les plates-formes prises en charge et la matrice de compatibilité ISSU :

Prise en charge ISSU entre versions

Prérequis avant ISSU

Remarque : L'exemple de ce document est basé sur les commutateurs Cisco Catalyst 9500 configurés comme Stackwise-Virtual. Les étapes s'appliquent aux châssis autonomes Cat9400 / Cat9600 avec deux moteurs de supervision, ainsi qu'aux périphériques Catalyst 3850 / Catalyst 9000 configurés en tant que Stackwise-Virtual. Pour éviter toute perte de paquets pendant le processus, ISSU exige une redondance dans vos noeuds vers les périphériques actifs et en veille. Étant donné qu'un rechargement se produit et qu'une commutation entre eux est nécessaire pour garantir une haute disponibilité dans votre réseau.

---

1. Vérifier la version actuelle du code

<#root>

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

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

2. Vérifiez le mode de démarrage

ISSU n'est pris en charge que si les deux commutateurs dans Stackwise Virtual sont démarrés en mode Install.

<#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"
```

Si le châssis est amorcé en mode bundle, l'ISSU n'est pas pris en charge. Vous pouvez voir des erreurs comme celle-ci si vous essayez d'exécuter ISSU lorsque les commutateurs s'exécutent en mode Bundle.

<#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. Vérifiez si la mémoire flash est suffisante

<#root>

C9500#dir flash: | in free

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

C9500#dir stby-flash: | in free

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

Assurez-vous que vous disposez d'au moins 1 Go d'espace dans la mémoire flash pour développer une nouvelle image. Nettoyez les anciens fichiers d'installation en cas d'espace insuffisant et utilisez la commande install remove inactive.

4. Vérifiez si les commutateurs sont en mode 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), V
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), V
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. Vérifiez si le démarrage automatique est activé

<#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
```

Si l'amorçage automatique n'est pas activé, vous pouvez le modifier comme suit :

<#root>

C9500(config)#no boot manual

6. Vérifiez l'état actuel de l'ISSU et de l'installation

Remarque : Cette étape est très 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. Copier la nouvelle image à mettre à niveau / rétrograder

Il suffit de copier l'image uniquement sur le châssis actif (dans le cas de Stackwise) ou sur le superviseur actif (dans le cas d'un double SUP 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]?
```

Workflows ISSU - Mise à niveau réelle

Une fois toutes les conditions préalables vérifiées, vous pouvez procéder à la mise à niveau réelle et utiliser l'une des méthodes répertoriées :

- Processus en une étape (une seule étape et ne prend pas en charge la restauration)
- Processus en trois étapes (comprend trois étapes et prend en charge la restauration en cas de défaillance)

Flux de travail en une étape

Ce workflow ne comporte qu'une seule étape et facilite l'optimisation.



Remarque : Vous ne pouvez pas revenir en arrière car la mise à niveau est validée automatiquement. Passez au processus en trois étapes si vous souhaitez revenir en arrière.

<#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 (SSO).
```

<<<< 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
```

Passez à la section après avoir terminé l'ISSU.

Processus en trois étapes

- Ce workflow comprend trois étapes : ajouter, activer et valider. Après activation, tous les commutateurs sont mis à niveau vers la nouvelle version du logiciel, à l'exception du fait que le logiciel n'est pas validé automatiquement, mais qu'il doit être exécuté manuellement via la commande install commit.
- L'avantage de cette approche est que le système peut être restauré à une version logicielle précédente.
- Le système revient automatiquement en arrière si le temporisateur de retour arrière n'est pas arrêté et utilise la commande install auto-abort-timer stop ou la commande install commit. Si le minuteur de restauration est arrêté, la nouvelle version du logiciel peut être exécutée sur le périphérique pendant une durée quelconque, puis restaurée à la version précédente.

Étape 1 : installation et ajout

Cette commande télécharge l'image dans le bootflash et l'étend sur les deux commutateurs.

<#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,
```

Étape 2 : installation et activation

- Lorsque vous exécutez cette commande, cette séquence d'événements se produit :

  (i) Un minuteur de restauration est démarré. Si le temporisateur de restauration expire, le système revient au même état avant le début de l'ISSU. Le temporisateur de restauration peut être arrêté lorsque vous utilisez la commande install auto-abort-timer stop. L'ISSU peut être restauré lorsque vous utilisez la commande install abort issu.

  (ii) Le commutateur de secours est mis en service avec le nouveau logiciel et il se recharge avec la nouvelle version du logiciel. Ensuite, le commutateur actif est provisionné avec le nouveau logiciel et il se recharge. Le commutateur de secours avec la nouvelle image devient alors le commutateur actif et l'ancien commutateur actif devient le commutateur de secours.

À la fin de cette procédure, les deux commutateurs s'exécutent avec la nouvelle image logicielle.

<#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:


[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

```
*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 11] 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 ar

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

À la fin de l'état Activate, vérifiez les états 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 ISSUCurrent 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.

Étape 3 : installation de la validation

La commande commit effectue le nettoyage nécessaire, active le nouveau logiciel comme permanent (supprime l'ancienne version du logiciel) et arrête le minuteur de restauration. Tout redémarrage après la validation peut démarrer avec un nouveau logiciel.

<#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!!!!!
```

Liste de contrôle Post ISSU

Une fois l'ISSU terminée,

- Vérifiez si les deux commutateurs fonctionnent sur le nouveau logiciel.
- Vérifiez que le résultat de la commande show issue state detail est propre et qu'aucun ISSU n'est en cours.

- Vérifiez le résultat de la commande show install issue history pour vous assurer que l'ISSU fonctionne correctement (commande disponible uniquement avec la version 16.10.1 et ultérieure).
- Il est recommandé de prévoir suffisamment de temps d'imprégnation dans le nouveau logiciel avant d'activer une nouvelle fonctionnalité.

Action en cas d'échec ISSU

- En cas d'échec de l'ISSU, l'abandon automatique est supposé rétablir l'état initial du système (ancienne image). Cependant, si cela échoue également, une récupération manuelle du châssis est attendue.
- Lors d'une récupération manuelle, vérifiez si l'image plus ancienne est exécutée à la fois en mode actif et en mode veille (sinon, récupérez le châssis individuel).
- Après vous être assuré que les deux châssis exécutent l'ancienne image, exécutez install remove inactive pour supprimer tous les packages d'image inutilisés.
- Une fois que les deux châssis exécutent l'ancien logiciel, nettoyez manuellement tous les états internes du fonctionnement de l'ISSU. (Reportez-vous ici à la procédure de nettoyage des états ISSU internes).

Abandonner ISSU

- Dans le processus en 3 étapes, lors de l'activation de l'ISSU, si le délai d'abandon expire, le système peut abandonner automatiquement l'image plus ancienne.



Remarque : Si la mise en veille n'atteint pas l'authentification unique lors de l'abandon, un abandon manuel est requis. En outre, si, pour une raison quelconque, vous souhaitez interrompre l'ISSU entre les deux, une interruption manuelle est requise.

<#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
```

État ISSU propre

Si la mise à niveau/rétrogradation/abandon/abandon automatique de l'ISSU échoue, un nettoyage manuel des états internes de l'ISSU est requis.

<#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
```

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.