

# Catalyst 3850 및 Catalyst 9000 시리즈 스위치에서 ISSU 수행

## 목차

---

[소개](#)

[ISSU란?](#)

[지원되는 플랫폼 및 릴리스 지원 매트릭스](#)

[ISSU 이전의 전제 조건](#)

- [1. 현재 코드 버전 확인](#)
- [2. 부팅 모드 확인](#)
- [3. 플래시에 사용 가능한 메모리가 충분하지 확인합니다](#)
- [4. 스위치가 SSO 모드인지 확인합니다.](#)
- [5. 자동 부팅이 활성화되었는지 확인합니다.](#)
- [6. 현재 ISSU 및 설치 상태 확인](#)
- [7. 업그레이드/다운그레이드할 새 이미지를 복사합니다.](#)

[ISSU 워크플로 - 실제 업그레이드](#)

[1단계 워크플로](#)

[3단계 워크플로](#)

[ISSU 체크리스트 게시](#)

[ISSU 장애 조치](#)

[ISSU 중단](#)

[Clean ISSU 상태](#)

---

## 소개

이 문서에서는 Catalyst 3850 및 Catalyst 9000 시리즈 스위치에서 ISSU(In-Service Software Upgrade)를 수행하는 단계를 설명합니다.

## ISSU란?

In-Service Software Upgrade는 네트워크에서 패킷을 계속 전달하는 동안 이미지를 디바이스의 다른 이미지로 업그레이드하는 프로세스입니다. ISSU는 네트워크 관리자가 소프트웨어 업그레이드를 수행할 때 네트워크 중단을 방지할 수 있도록 도와줍니다. 이미지는 설치 모드에서 업그레이드되며, 각 패키지는 개별적으로 업그레이드됩니다.

ISSU는 Stackwise-Virtual을 지원하는 모든 Catalyst 3850 및 Catalyst 9000 시리즈와 듀얼 슈퍼바이저가 포함된 Catalyst 9400/9600 독립형 새시에서 지원됩니다.


SVL(StackWise Virtual)은 서로 연결되어 하나의 가상 스위치를 구성하는 두 개의 스위치로 구성됩니다. SVL은 서비스 중 소프트웨어 업그레이드를 지원합니다.

# 지원되는 플랫폼 및 릴리스 지원 매트릭스

ISSU를 진행하기 전에, 플랫폼이 실제로 ISSU를 지원하는지 확인하십시오. 또한 현재 코드와 대상 코드 간에 ISSU가 지원되는지 확인합니다. 지원되는 플랫폼 및 ISSU 호환성 매트릭스에 대한 모든 세부 정보는 다음 사이트에서 확인할 수 있습니다.

[릴리스 간 ISSU 지원](#)

## ISSU 이전의 전제 조건

 참고: 이 문서의 예는 Stackwise-Virtual로 구성된 Cisco Catalyst 9500 스위치를 기반으로 합니다. 이 단계는 듀얼 슈퍼바이저 엔진이 포함된 Cat9400/Cat9600 독립형 새시와 Stackwise-Virtual로 구성된 Catalyst 3850/Catalyst 9000 디바이스에도 적용됩니다. 프로세스 중에 패킷 손실을 방지하려면 ISSU에서 액티브 및 스탠바이 디바이스 둘 다에 대해 노드에 이중화를 구현해야 합니다. 다시 로드가 발생하므로 네트워크에서 고가용성을 보장하려면 두 항목 간의 전환이 필요합니다.

### 1. 현재 코드 버전 확인

<#root>

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

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

### 2. 부팅 모드 확인

ISSU는 Stackwise Virtual의 두 스위치가 모두 설치 모드에서 부팅된 경우에만 지원됩니다.

<#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"
```

새시가 번들 모드로 부팅되는 경우 ISSU는 지원되지 않습니다. 스위치가 번들 모드로 실행 중일 때 ISSU를 실행하려고 하면 이와 같은 오류가 표시됩니다.

```
<#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. 플래시에 사용 가능한 메모리가 충분한지 확인합니다

```
<#root>
```

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

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

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

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

새 이미지를 확장하려면 플래시에 최소 1GB의 공간이 있어야 합니다. 공간이 부족한 경우 오래된 설치 파일을 정리하고 `install remove inactive` 명령을 사용합니다.

### 4. 스위치가 SSO 모드인지 확인합니다.

```
<#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. 자동 부팅이 활성화되었는지 확인합니다.

```
<#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

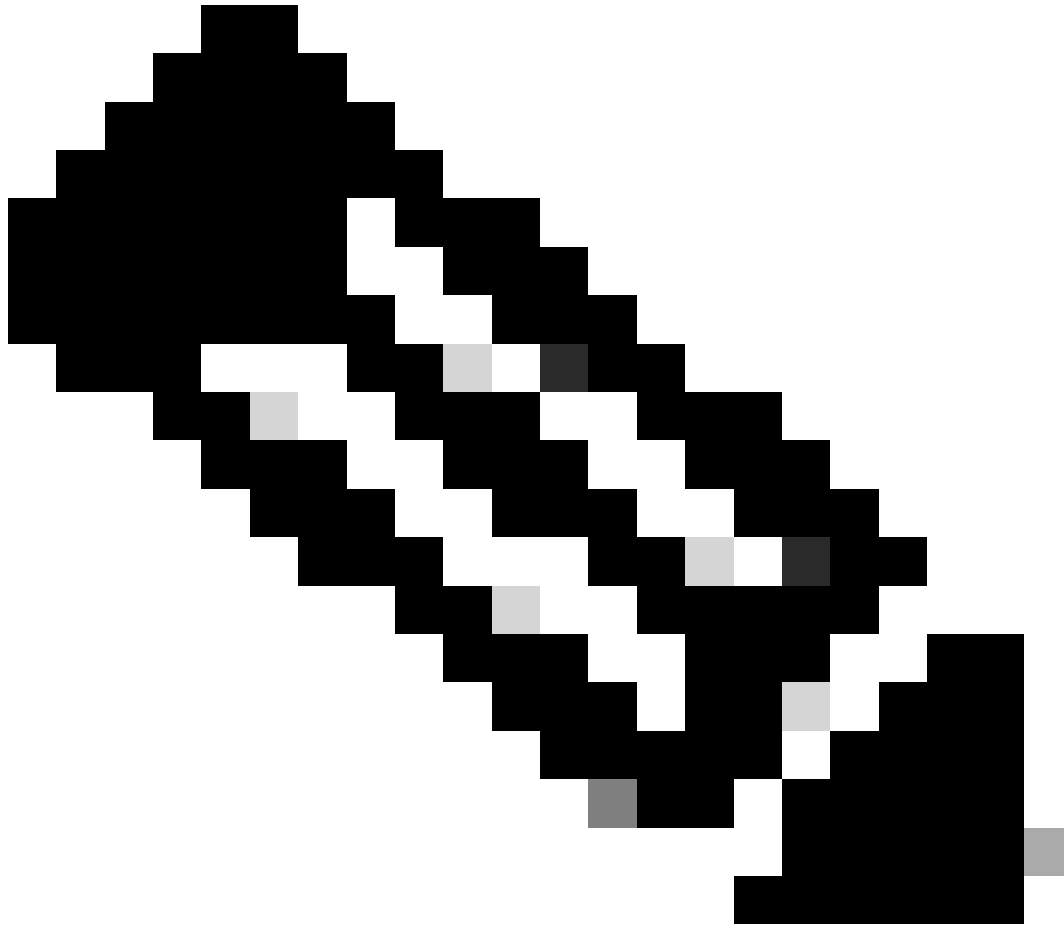
Enable Break = no  
Boot Mode = DEVICE  
iPXE Timeout = 0

자동 부팅이 활성화되지 않은 경우 다음과 같이 변경할 수 있습니다.

<#root>

c9500(config)#no boot manual

## 6. 현재 ISSU 및 설치 상태 확인



참고: 이 단계는 매우 중요합니다!

---

```
<#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. 업그레이드/다운그레이드할 새 이미지를 복사합니다.

이미지는 활성 새시(Stackwise의 경우) 또는 활성 슈퍼바이저(Cat9400 이종 SUP의 경우)에만 복사하면 됩니다.

<#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]?

## ISSU 워크플로 - 실제 업그레이드

모든 전제 조건이 확인되면 실제 업그레이드를 진행하고 다음 방법 중 하나를 사용할 수 있습니다.

- 1단계 워크플로(1단계만 포함되며 롤백을 지원하지 않음)
- 3단계 워크플로(3단계 포함, 장애 시 롤백 지원)

### 1단계 워크플로

이 워크플로는 한 단계만 포함되며 최적화에 도움이 됩니다.

---

참고: 업그레이드가 자동으로 커밋되므로 롤백할 수 없습니다. 롤백하려면 3단계 워크플로로 진행합니다.

---

<#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-1) reloads here!!!>>>>>>>>>>  
\*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

ISSU가 완료되면 를 진행합니다.

### 3단계 워크플로

- 이 워크플로는 세 단계로 구성됩니다. 추가, 활성화 및 커밋합니다. 활성화 후에는 소프트웨어가 자동으로 커밋되지 않는다는 점을 제외하고 모든 스위치가 새 소프트웨어 버전으로 업그레이드되지만 install commit 명령을 통해 수동으로 수행해야 합니다.
- 이 접근 방식의 이점은 시스템을 이전 소프트웨어 버전으로 롤백할 수 있다는 것입니다.
- 롤백 타이머가 중지되지 않은 경우 시스템은 자동으로 롤백하고 install auto-abort-timer stop 또는 install commit 명령을 사용합니다. 롤백 타이머가 중지된 경우, 새 소프트웨어 버전은 모든 기간 동안 디바이스에서 실행된 다음 이전 버전으로 롤백할 수 있습니다.

1단계. add를 설치합니다.

이 명령은 이미지를 bootflash로 다운로드하고 두 스위치에서 확장합니다.

<#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,

## 2단계. activate 설치

- 이 명령을 실행하면 다음과 같은 일련의 이벤트가 발생합니다.

(i) 롤백 타이머가 시작됩니다. 롤백 타이머가 만료되면 시스템은 ISSU를 시작하기 전에 동일한 상태로 롤백합니다. rollback timer는 install auto-abort-timer stop 명령을 사용할 때 중지할 수 있습니다. install abort issu 명령을 사용할 때 ISSU를 롤백할 수 있습니다.

(ii) 대기 스위치는 새 소프트웨어로 프로비저닝되며 새 소프트웨어 버전으로 다시 로드됩니다. 다음으로, 활성 스위치가 새 소프트웨어로 프로비저닝되고 다시 로드됩니다. 이제 새 이미지가 있는 스탠바이 스위치가 액티브 스위치가 되고 이전 액티브 스위치가 스탠바이가 됩니다

이 절차가 끝나면 두 스위치 모두 새 소프트웨어 이미지로 실행됩니다.

<#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 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 ar**

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

Activate(활성화) 상태가 끝나면 ISSU 상태를 확인합니다.

<#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

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.

3단계. Commit을 설치합니다.

commit 명령은 필요한 정리를 수행하고, 새 소프트웨어를 영구적으로 활성화하고(이전 버전의 소프트웨어 제거), 롤백 타이머를 중지합니다. 커밋 후 재부팅하면 새 소프트웨어로 부팅할 수 있습니다.

<#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!!!!
```

## ISSU 체크리스트 게시

ISSU가 성공적으로 완료되면

- 두 스위치가 모두 새 소프트웨어에서 실행되는지 확인합니다.
- show issu state detail output to be clean and not showing any ISSU in progress(issu 상태 세 부사항 출력이 깨끗하고 진행 중인 ISSU가 표시되지 않음)를 선택합니다.
- show install issu history output(설치 issu 기록 출력 표시)을 선택하여 성공적인 ISSU 작업을 확인합니다(16.10.1 릴리스 이상에서만 명령 사용 가능).
- 새 기능을 사용하기 전에 새 소프트웨어에 충분한 시간을 투자하는 것이 좋습니다.

## ISSU 장애 조치

- ISSU가 실패하면 자동 중단으로 시스템을 초기 상태(이전 이미지)로 복구할 수 있습니다. 그러나 이 역시 실패할 경우 새시의 수동 복구가 필요합니다.
- 수동 복구 중에 액티브 및 스탠바이 모두 이전 이미지를 실행하는지 확인합니다(그렇지 않은 경우 개별 새시를 복구함).
- 두 새시에서 모두 기존 이미지를 실행하도록 한 후 install remove inactive(비활성 설치 제거)를 실행하여 사용하지 않는 이미지 패키지를 제거합니다.
- 두 새시가 모두 이전 소프트웨어를 실행하면 ISSU 작업의 모든 내부 상태를 수동으로 정리합니다. (내부 ISSU 상태를 정리하는 방법은 여기를 참조하십시오.)

## ISSU 중단

- 3단계 워크플로에서 ISSU 활성화 프로세스 중에 중단 타이머가 만료되면 시스템이 이전 이미지로 자동 중단할 수 있습니다.

---

참고: 중단하는 동안 스탠바이가 SSO에 도달하지 않으면 수동 중단해야 합니다. 또한 어떤 이유로든 ISSU를 중단하려는 경우 수동으로 중단해야 합니다.

---

<#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
```

## Clean ISSU 상태

ISSU 업그레이드/다운그레이드/중단/자동 중단이 성공하지 못하면 ISSU 내부 상태를 수동으로 정리해야 합니다.

```
<#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

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.