

NXOS - 디스크의 내용을 안전하게 지우기

목차

[소개](#)

[배경 정보](#)

[자신에게 적합한 절차를 결정하는 방법](#)

[준비](#)

[SSD가 있는 스위치에서 Init-System 절차 사용](#)

[eUSB를 사용하는 스위치/수퍼바이저/시스템 컨트롤러에서 추가 절차 사용](#)

[dd를 사용하여 I/O 모듈의 관련 파티션에 0바이트 쓰기](#)

[스위치 복구 및 OS 재설치](#)

소개

이 문서에서는 표준 Linux 유틸리티를 사용하는 Cisco Nexus 스위치의 디스크를 안전하게 삭제하는 방법에 대해 설명합니다. 이는 특정 군사 및 정부 고객이 보안 지역에서 비보안 영역으로 장비를 이동하거나 규정 준수 요구 사항을 가진 다른 고객이 장비를 구상에서 벗어나는 데 필요합니다.

배경 정보

스위치에 SSD 또는 eUSB 드라이브가 있는지 여부에 따라 두 가지 옵션이 있습니다.

- Init-System은 SSD가 있는 최신 모델 스위치에서 사용됩니다. Init-System은 ATA Secure Erase를 사용하여 드라이브의 모든 섹터에 이진 0을 씁니다.
- eUSB 드라이브가 있는 이전 모델 스위치의 경우 제로 바이트 지우기 방법을 사용하여 드라이브의 모든 섹터에 0을 쓸 수도 있습니다.

문서화된 절차에 사용된 표준 유틸리티는 스토리지 디스크의 데이터를 안전하게 제거하는 일련의 명령을 사용하며, 대부분의 경우 데이터를 복구하는 것이 어렵거나 불가능하게 만듭니다.

이 설명서에서는 Cisco Nexus 3000 Series 스위치, Cisco Nexus 5000 Series 스위치, Cisco Nexus 9000 Series 스위치, Cisco Nexus 700 Series 스위치 및 Cisco MDS Series 스위치를 염두에 두고 두 프로세스를 모두 설명하지만, 초기화 시스템 또는 Access Bash를 보유한 경우 대부분의 다른 Cisco Nexus 스위치에서 작동합니다. 실행 중인 스위치나 소프트웨어 릴리스에서 기능 bash가 Bash 셸에 액세스할 수 있도록 설정할 수 없는 경우 Cisco TAC에서 서비스 요청을 열어 이 절차에 대한 디버그 플러그인을 사용하는 데 도움을 받으십시오.

자신에게 적합한 절차를 결정하는 방법

PID가 0 값을 반환하면 시스템은 SSD를 사용하고 Init-System 메서드를 사용하여 드라이브를 지울 수 있습니다.

PID가 1 값을 반환하면 시스템에서 eUSB 드라이브를 사용하고 있으므로 0바이트 지우기 방법을 사용해야 합니다.

```
F340.23.13-C3064PQ-1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
F340.23.13-C3064PQ-1(config)# feature bash-shell
F340.23.13-C3064PQ-1(config)#
F340.23.13-C3064PQ-1(config)# exit
F340.23.13-C3064PQ-1# run bash bash-4.2$ cat /sys/block/sda/queue/rotational 1
bash-4.2$
```

위 절차를 수행한 후 시스템에 있는 드라이브 유형과 디스크 내용을 안전하게 삭제하기 위해 사용해야 하는 절차를 아직 알 수 없으면 Cisco TAC에서 서비스 요청을 엽니다.

준비

드라이브를 초기화하기 전에 다음 항목이 있어야 합니다.

1. 스위치에 대한 콘솔 액세스.
2. management0 인터페이스를 통해 TFTP 서버에 액세스 - 현재 컨피그레이션을 백업한 다음 OS를 복원하는 데 필요합니다.
3. 이 프로세스에서 제거되므로 시스템에서 오프라인으로 저장할 실행 중인 구성 및 기타 파일의 백업!

참고: 더 이상 프로덕션 상태가 아니거나 프로덕션 새시에 설치된 제품에 대해 이 절차를 수행하는 것이 좋습니다. 의도하지 않은 네트워크 중단을 방지하기 위해 이 절차를 수행하기 전에 디바이스 또는 부품을 비생산 환경으로 이동해야 합니다.

SSD가 있는 스위치에서 Init-System 절차 사용

참고: 모듈형 기반 스위치 내부의 수퍼바이저에서 이 절차를 수행할 때는 시스템에 절차를 수행할 계획인 수퍼바이저만 설치하는 것이 좋습니다.

1. 콘솔을 통해 연결된 상태에서 스위치를 다시 로드하거나 전원을 껐다가 켜십시오.
2. 스위치가 부팅되는 동안 CTRL-C를 사용하여 스위치를 loader> 프롬프트로 전환합니다.
3. loader> 프롬프트에서 cmdline recoverymode=1을 입력합니다. **switch(boot)#** 프롬프트에서 스위치 부팅을 중지합니다.

```
loader > cmdline recoverymode=1
```

4. **boot bootflash:<nxos_filename.bin>**으로 부팅 절차를 시작합니다.

```
loader > boot bootflash:nxos.7.0.3.I7.8.bin
```

5. 스위치가 **switch(boot)#** 프롬프트로 부팅됩니다. 이 프롬프트에서 초기화 시스템 CLI와 **clear nvram** CLI를 사용하여 라이선스 블록을 제외하고 nvram의 모든 블록에 0을 기록합니다. **참고:** 이 테스트는 Intel Core i3-CPU @ 2.50GHz 및 110G SSD가 있는 N9K-C9372TX-E에서 수행되었습니다. 초기화 시스템에 대한 총 시간(8초):

```
switch(boot)# clear nvram
```

```
switch(boot)# init system This command is going to erase your startup-config, licenses as well as the contents of your bootflash:. Do you want to continue? (y/n) [n] y
```

6. 5단계가 완료되면 스위치를 다시 로드합니다.

```
switch(boot)# reload
This command will reboot this supervisor module. (y/n) ? y
```

eUSB를 사용하는 스위치/수퍼바이저/시스템 컨트롤러에서 추가 절차 사용

1. 콘솔 포트를 통해 스위치의 관리자 계정에 로그인합니다.

참고:모듈형 기반 스위치 내부의 Supervisor에서 이 절차를 수행할 경우 시스템에 절차를 수행할 Supervisor만 설치하는 것이 좋습니다.

2. 구성 모드에서 기능 **bash-shell**을 활성화하고 **실행 bash**를 사용하여 Bash-프롬프트를 입력합니다(N3K/9K만 해당). 다른 Cisco Nexus 스위치에 액세스하려면 디버그 플러그인이 필요합니다.)

```
F340.23.13-C3064PQ-1# config terminal
F340.23.13-C3064PQ-1(config)# feature bash-shell F340.23.13-C3064PQ-1(config)# exit
F340.23.13-C3064PQ-1# run bash
bash-4.2$
```

```
N7K-1# load n7000-s2-debug-sh.7.2.1.D1.1.gbin Loading plugin version 7.2(1)D1(1)
##### Warning: debug-plugin is for engineering internal use only! For security reason, plugin image has been deleted.
##### Successfully loaded debug-plugin!!! Linux(debug)#
```

3. **sudo su**로 루트 액세스 획득

참고:이 절차에서는 디버그 플러그인을 사용하는 Cisco Nexus 7000 Series 스위치에 대해 이 단계를 건너뛸 수 있습니다.

```
bash-4.2$ sudo su -
root@F340#
```

4. Nexus 9000 Series 스위치에 설치된 시스템 컨트롤러에서 이 절차를 수행하려면 이 절차를 수행할 슬롯 번호에 원격으로 로그인해야 합니다. 예를 들어, 슬롯 29의 시스템 컨트롤러에 대해 다음 작업을 수행합니다.

```
N9K-EOR# run bash bash-4.2$ sudo su - root@N9K-EOR#rlogin lc29 root@sc29:~#
```

5. **fdisk -l**로 각 디스크의 블록 크기를 확인합니다. N3K-C3064PQ-10X에는 512바이트 블록 크기에 /dev/sda만 있습니다. 다음을 참조하십시오.

참고:일부 Cisco Nexus 스위치에는 단일 디스크 이상이 있을 수 있습니다.추가 작업을 수행할 때 고려해야 합니다.예를 들어 N7K-SUP2에는 /dev/sda, /dev/sdb, /dev/sdc//dev/md2, /dev/md3, /dev/md4, /dev/md5 및/dev/md6이 있습니다. 보안 지우기 절차를 올바르게 완료하려면 각 작업에 대해 dd 작업을 수행해야 합니다.

참고:Cisco Nexus 9000 Series 스위치에서 시스템 컨트롤러는 /dev/mtdblock0, /dev/mtdblock1, /dev/mtdblock2, /dev/mtdblock3, /dev/mtdblock4, /dev/mtdblock5 및 /dev/mtdblock6을 가집니다. 보안 지우기 절차를 올바르게 완료하려면 각 작업에 대해 추가 작업을 수행해야 합니다.

```
root@F340# fdisk -l
```

```
Disk /dev/sda: 2055 MB, 2055208960 bytes
64 heads, 62 sectors/track, 1011 cylinders
Units = cylinders of 3968 * 512 = 2031616 bytes
Disk identifier: 0x8491e758
```

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1		1	5	9889	83	Linux
/dev/sda2		6	45	79360	5	Extended
/dev/sda3		67	1011	1874880	83	Linux
/dev/sda4		46	66	41664	83	Linux
/dev/sda5		6	26	41633	83	Linux
/dev/sda6		27	45	37665	83	Linux

6. 디스크의 모든 섹터에 0바이트를 씁니다.

참고:이 테스트는 Intel Celeron CPU P4505 @1.87GHz 및 13G eUSB가 있는 N3K-C3064PQ-10X에서 수행되었으며, 0바이트 프로세스는 ~501초가 걸렸습니다.

```
root@F340# dd if=/dev/zero of=/dev/sda bs=512
```

참고:일부 부분에서는 이 단계에서 생성된 커널 메시지를 볼 수 있습니다.

7. 5단계가 완료되면 스위치, 슈퍼바이저 또는 시스템 컨트롤러를 다시 로드합니다.

참고:Cisco Nexus 9000 Series 모듈형 스위치에서 시스템 컨트롤러를 다시 로드하려면 `reload module <slot_number>` CLI를 입력합니다.

```
bash-4.2$ exit
F340.23.13-C3064PQ-1# exit
F340.23.13-C3064PQ-1# reload
WARNING: There is unsaved configuration!!!
WARNING: This command will reboot the system
Do you want to continue? (y/n) [n] y
```

dd를 사용하여 I/O 모듈의 관련 파티션에 0바이트 쓰기

1. 콘솔 포트를 통해 스위치의 관리자 계정에 로그인합니다.

2. 구성 모드에서 기능 `bash-shell`을 활성화하고 실행 `bash`를 사용하여 Bash-프롬프트를 입력합니다(N3K/N9K에만 해당). 다른 Cisco Nexus 스위치에 액세스하려면 디버그 플러그인이 필요합니다

) 디버그 플러그인이 필요한 경우 Cisco TAC에 문의하고 2단계 대신 3단계를 수행하십시오.

참고:Bash-prompt에서 LC/FM에 액세스하려면 루트 액세스 권한을 얻은 후 `rlogin lc# CLI`를 입력합니다.이제 CLI의 #을 작업을 수행하려는 슬롯 번호로 교체합니다.

```
N7K-1# config terminal
Enter configuration commands, one per line. End with CNTL/Z.
N7K-1(config)# feature bash-shell
N7K-1(config)# exit
N7K-1# run bash
bash-4.3$
```

```
N9K-EOR# run bash bash-4.2$ sudo su - root@N9K-EOR#rlogin lc22 root@fm22:~#
```

3. 디버그 플러그인을 사용하는 Cisco Nexus 스위치의 경우 실행 중인 소프트웨어 버전의 디버그 플러그인이 bootflash에 복사되었는지 확인하고 보안 지우기 절차를 실행할 모듈에 디버그 플러그인을 로드합니다.

참고:Supervisor 모듈에서 사용할 수 있는 디버그 플러그인 이미지와 달리 Nexus 7000 Series 스위치 I/O 모듈에 사용할 별도의 디버그 플러그인 이미지가 있습니다.스위치에서 실행되는 소프트웨어 릴리스에 LC 이미지를 사용합니다.

```
switch# attach module 3 Attaching to module 3 ... To exit type 'exit', to abort type '$.'
module-3# load bootflash:dplug-lc_p476-bin.7.2.1.D1.1.bin Name of debug-plugin from SUP:
'/bootflash/dplug-lc_p476-bin.7.2.1.D1.1.bin' Downloaded debug-plugin to LC: '/tmp/dplug-
lc_p476-bin.7.2.1.D1.1.bin' Loading plugin version 7.2(1)D1(1)
##### Warning: debug-plugin is for
engineering internal use only! #####
Warning: /debug-plugin/.autorun is using deprecated /bin/bash. Please change to /bin/sh
Successfully loaded debug-plugin!!! Linux(debug)#
```

4. 그 다음, Cisco Nexus 7000 Series Line Card의 경우, 파일 시스템에 마운트된 위치와 `mnt/pss`를 결정합니다. 이렇게 하려면 `mount` 명령을 사용하여 `/mnt/plog(logflash)` 및 `/mnt/pss`가 상주하는 위치를 찾습니다.

참고:Cisco Nexus 9000 Series Line Card의 경우, `/dev/mmcblk0`에서 `dd` 작업을 수행합니다.

참고:Cisco Nexus 9000 Series Fabric Module의 경우 `/tmpfs`,`/dev/root`, `/dev/zram0`,`/dev/loop0`,`/dev/loop1` 및 `/onfs`에서 `dd` 작업을 수행합니다.

```
Linux(debug)# mount | grep plog /dev/mtdblock2 on /mnt/plog type jffs2 (rw,noatime)
Linux(debug)# Linux(debug)# mount | grep pss tmpfs on /mnt/pss type tmpfs
(rw,size=409600k,mode=777) Linux(debug)#
```

5. 이제 `/mnt/plog`가 `/dev/mtdblock2`에 있고 `/mnt/pss`가 `/tmpfs`에 있는 것으로 알려져, `dd` 명령을 사용하여 모두 0바이트를 쓰고, 디버그 플러그인을 종료하고, 모듈을 다시 로드합니다.

```
Linux(debug)# dd if=/dev/zero of=/dev/mtdblock2 bs=1024 dd: writing '/dev/mtdblock2': No space
left on device 15361+0 records in 15360+0 records out Linux(debug)# Linux(debug)# dd if=/dev/zero
of=/tmpfs bs=1024 dd: writing '/tmpfs': No space left on device 23781+0 records in 23780+0
records out Linux(debug)# Linux(debug)# exit
```

```
##### Warning: for security reason, please delete plugin image on sup.##### module-3# exit rlogin: connection closed. switch# switch# reload module 3 This command will reload module 3. Proceed[y/n]? [n] y reloading module 3 ... switch#
```

스위치 복구 및 OS 재설치

스위치를 껐다가 다시 켜면 로더 프롬프트에 부팅됩니다.

loader> 프롬프트에서 복구하려면 다음 단계에 따라 스위치를 TFTP로 부팅해야 합니다.

1. 스위치의 mgmt0 인터페이스에 IP 주소를 설정(또는 할당):

```
loader > set ip <IP_address> <Subnet_Mask>
```

2. 부팅하는 TFTP 서버가 다른 서브넷에 있는 경우 스위치에 기본 게이트웨이를 할당합니다.

```
loader > set gw <GW_IP_Address>
```

3. 부팅 프로세스를 수행합니다. 스위치가 switch(boot) 프롬프트로 부팅됩니다.

참고:Cisco Nexus 5000 Series 스위치, Cisco Nexus 6000 Series 스위치, Cisco Nexus 7000 Series 스위치 등 별도의 시스템/kickstart 이미지를 사용하는 스위치의 경우 이 단계에서 kickstart 이미지를 부팅해야 합니다. Cisco Nexus 9000 Series 스위치 및 Cisco Nexus 3000 Series 스위치와 같은 단일 NXOS 이미지를 사용하는 스위치의 경우 이 단계에서 단일 이미지를 부팅해야 합니다.

```
loader > boot tftp://
```

4. clear nvram, Init system 및 format bootflash를 수행합니다.

참고:Cisco Nexus 5000 Series 스위치 및 Cisco Nexus 6000 Series 스위치의 경우 **switch(boot)#** 프롬프트에서 nvram 지우기를 사용할 수 없습니다.

```
switch(boot)# clear nvram
switch(boot)# init system
This command is going to erase your startup-config, licenses as well as the contents of your bootflash:.
Do you want to continue? (y/n) [n] y
Initializing the system ...
```

<snip>

```
switch(boot)# format bootflash:
This command is going to erase the contents of your bootflash:.
Do you want to continue? (y/n) [n] y
get_sup_active_slot failed with -1
Unknown card
Formatting bootflash:
```

<snip>

5. 스위치를 다시 로드합니다.

```
switch(boot)# reload This command will reboot this supervisor module. (y/n) ? y (c) Copyright
2011, Cisco Systems. N3000 BIOS v.5.0.0, Tue 06/05/2018, 05:24 PM
```

6. 스위치의 mgmt0 인터페이스에 IP 주소를 설정(또는 할당):

```
loader > set ip <IP_address> <Subnet_Mask>
```

7. 부팅하는 TFTP 서버가 다른 서브넷에 있는 경우 스위치에 기본 게이트웨이를 할당합니다.

```
loader > set gw <GW_IP_Address>
```

8. 스위치를 다시 로드합니다.

참고:이 단계(8)는 Cisco Nexus 5000 Series 스위치, Cisco Nexus 6000 Series 스위치, Cisco Nexus 7000 Series 스위치 수퍼바이저 모듈 또는 Cisco Nexus 9000 Series 스위치 수퍼바이저 모듈에서 이 절차를 수행할 때 필요하지 **않습니다**. Cisco Nexus 5000 Series 스위치, Cisco Nexus 6000 Series 스위치, Cisco Nexus 7000 Series 스위치 수퍼바이저 모듈 또는 Cisco Nexus 9000 Series 스위치 수퍼바이저 모듈에서 이 절차를 수행하려면 9단계로 건너뛵니다.

```
loader> reboot
```

9. 부팅 프로세스를 수행합니다. 스위치가 **switch(boot)** 프롬프트로 부팅됩니다.

참고:Cisco Nexus 7000 Series 스위치와 같이 별도의 시스템/kickstart 이미지를 사용하는 스위치의 경우 이 단계에서 kickstart 이미지를 부팅해야 합니다. Cisco Nexus 9000 Series 스위치 및 Cisco Nexus 3000 Series 스위치와 같은 단일 NXOS 이미지를 사용하는 스위치의 경우 이 단계에서 단일 이미지를 부팅해야 합니다.

```
loader > boot tftp://<server_IP>/<nxos_image_name>
```

10. Cisco Nexus 5000 Series 스위치, Cisco Nexus 6000 Series 스위치, Cisco Nexus 7000 Series 스위치 등 별도의 시스템/kickstart 이미지를 사용하는 스위치의 경우 이 단계에서 스위치를 부팅하기 위한 몇 가지 추가 단계를 수행해야 합니다. 관리 0 IP 주소 및 서브넷 마스크를 구성하고 기본 게이트웨이를 정의해야 합니다. 이 작업이 완료되면 kickstart 및 시스템 이미지를 스위치에 복사하고 로드할 수 있습니다.

```
switch(boot)# config terminal Enter configuration commands, one per line. End with CNTL/Z.
switch(boot)(config)# interface mgmt 0 switch(boot)(config-if)# ip address 10.122.160.55
255.255.255.128 switch(boot)(config-if)# no shutdown switch(boot)(config-if)# exit
switch(boot)(config)# switch(boot)(config)# ip default-gateway 10.122.160.1
switch(boot)(config)# switch(boot)(config)# exit switch(boot)# switch(boot)# switch(boot)# copy
ftp: bootflash: Enter source filename:
```

11. Cisco Nexus 5000 Series 스위치, Cisco Nexus 6000 Series 스위치 및 Cisco Nexus 7000 Series 스위치 수퍼바이저 모듈의 경우 **switch(boot)#** 프롬프트에서 **bootflash:<system_image>**를 입력합니다. 이렇게 하면 스위치의 부팅 프로세스가 완료됩니다.

```
switch(boot)# load bootflash:<system_image>
```

12. 시스템 이미지가 성공적으로 로드되면 설정 프롬프트를 통해 원하는 사양에 맞게 디바이스 구성을 시작해야 합니다.