

Catalyst 9000 스위치의 앱 호스팅 이해

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[애플리케이션 호스팅 프레임워크 제한 사항](#)

[애플리케이션 호스팅 아키텍처](#)

[앱 호스팅을 위해 지원되는 Catalyst 모델](#)

[애플리케이션을 위한 하드웨어 리소스](#)

[컨테이너 네트워킹](#)

[Docker 런타임 옵션](#)

[지원되지 않는 Docker 런타임 옵션 목록](#)

[설정](#)

[문제 해결](#)

[관련 정보](#)

소개

이 문서에서는 Catalyst 9000 Series 플랫폼에서 애플리케이션 호스팅을 구현하고 문제를 해결하는 방법에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- LAN 스위칭 기본 사항
- 에 익숙함 Cisco IOS® XE 및 라이선싱.
- 스위칭 아키텍처 이해
- 기본 Linux 명령에 익숙합니다.

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- Catalyst 9300
- Catalyst 9400
- Catalyst 9500

- Cisco IOS® XE 및 16.12.X 또는 17.X 소프트웨어
- 자세한 내용은 앱 호스팅을 위해 지원되는 Catalyst 모델을 참조하십시오

 참고: 다른 Cisco 플랫폼에서 이러한 기능을 활성화하는 데 사용되는 명령에 대해서는 해당 구성 가이드를 참조하십시오.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 정보

애플리케이션은 프로토콜 분석기와 같은 관리 툴, 침입 탐지 서비스와 같은 보안 솔루션을 비롯하여 다양한 비즈니스 크리티컬 활용 사례를 처리하기 위해 엔터프라이즈 네트워크에서 필수적입니다. 전통적으로 이러한 애플리케이션은 외부 물리적 또는 클라우드 기반 가상 서버에서 작동했지만, 현대의 요구로 인해 이러한 접근 방식의 한계가 부각되었습니다.

Cisco Catalyst 9000 Series 스위치는 현대 디지털 시대를 위해 설계된 고급 기능을 통합하여 엔터프라이즈 네트워킹에 혁신을 가져왔습니다. Cisco IOS® XE 운영 체제 및 x86 CPU를 기반으로 하는 이 스위치는 기존 네트워크 장치 이상의 기능을 제공합니다. 애플리케이션 호스팅을 위한 플랫폼 역할을 하므로, 기업은 인프라를 통합하고 에지 컴퓨팅의 새로운 가능성을 활용할 수 있습니다. 따라서 보안 에이전트, IoT 센서, 트래픽 모니터링 툴과 같은 애플리케이션이 스위치에서 직접 실행되므로 외부 컴퓨팅 하드웨어가 필요하지 않습니다.

애플리케이션 호스팅 프레임워크 제한 사항

Catalyst 9000에서 Application Hosting Framework를 사용하려면 다음 요구 사항이 필요합니다.

- 스위치는 릴리스 버전 16.12를 실행 중이어야 합니다. 이 버전은 기본 Docker 엔진을 지원하므로 Docker 앱은 릴리스 16.12에서만 지원됩니다.
- Catalyst 9000에서 애플리케이션 호스팅을 활성화하려면 먼저 Cisco 인증 USB3.0 플래시 드라이브를 디바이스 후면 패널 USB3.0 포트에 설치해야 합니다. 앱 호스팅은 후면 패널 USB3.0에서만 작동합니다.
- Cisco Smart Licensing은 Catalyst 9000 플랫폼에 필요합니다. 앱 호스팅을 활성화하려면 Cisco DNA-Advantage 라이선스가 필요합니다.
- Docker 컨테이너만 지원됩니다. Docker 앱은 Cisco 지원 SSD 스토리지에 설치할 수 있습니다.
- 타사 애플리케이션에는 내부 플래시가 지원되지 않으며 ThousandEyes Agent와 같은 Cisco 서명 애플리케이션만 지원됩니다.
- 타사 컨테이너는 Catalyst 9000 스위치에서 제공하는 외부 SSD(Solid State Disk) 스토리지 옵션 중 하나에 상주해야 합니다. SSD 스토리지는 Catalyst 9300 Series 스위치에서 지원됩니다. SATA 스토리지는 Catalyst 9400 Series 및 Catalyst 9500 Series 고성능 모델과 Catalyst 9600 Series 스위치에서 지원됩니다. SSD 스토리지 또는 SATA 스토리지는 스위치의 전면 패널 USB 포트에서 사용할 수 없습니다.

- Cisco는 사전 패키징된 세 번째 제품을 제공하지 않습니다. 파티의 지원되지 않는 앱입니다. 고객이 직접 포장해야 합니다. 삼분의 일 cisco와 협업한 당사자 공급업체는 Cisco 설치와 관련된 링크를 공유할 수 있습니다.
- 애플리케이션 호스팅을 위한 전면 패널 USB 포트는 Catalyst 9300LM 모델에서만 지원됩니다

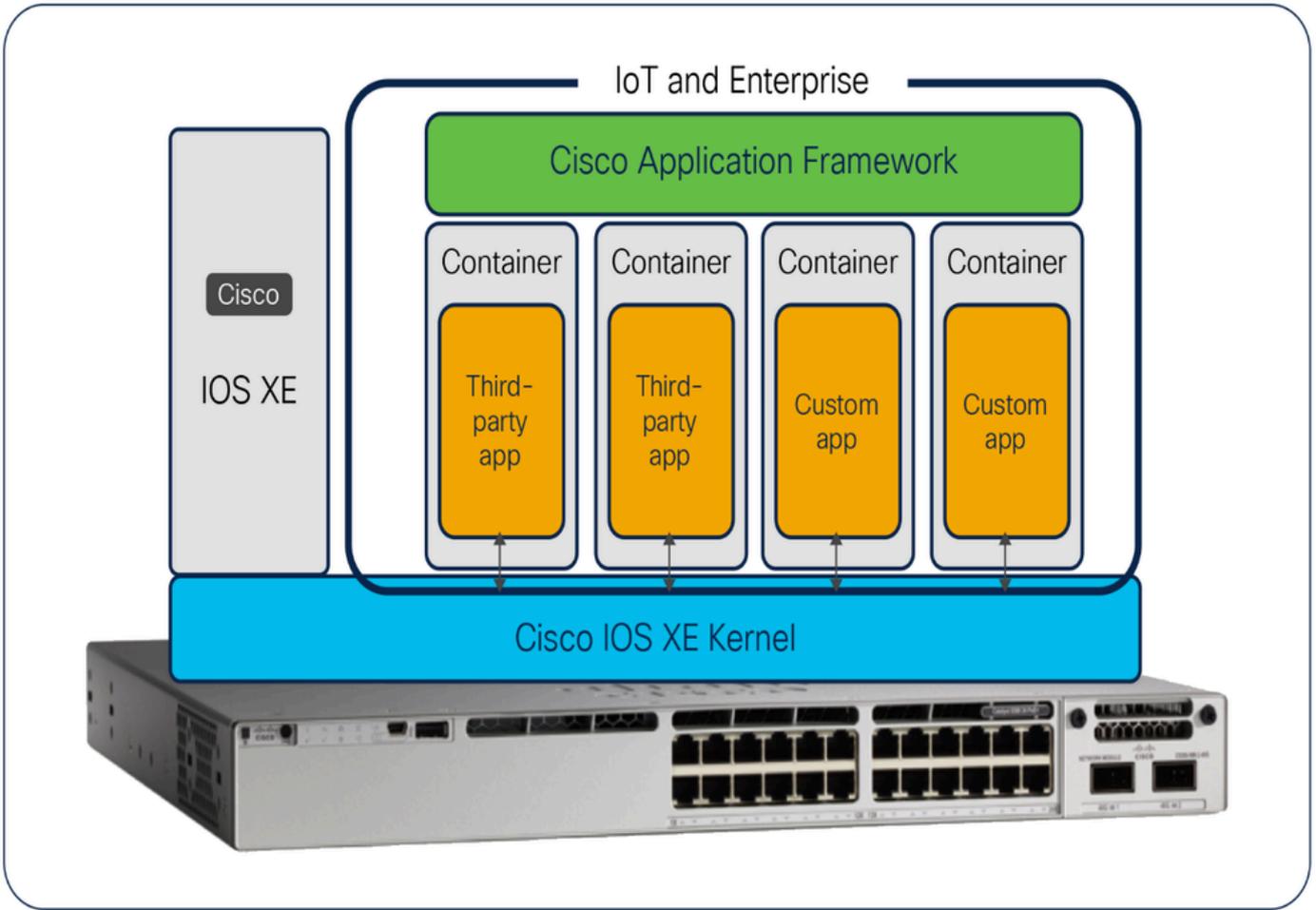
애플리케이션 호스팅 아키텍처

Cisco Catalyst 9000 스위치에서 애플리케이션 호스팅 기능을 지원하기 위해 이 스위치는 애플리케이션이 상주하고 실행될 수 있는 하드웨어 리소스를 제공합니다. Cisco Catalyst 9000 스위치에서 실행되는 Cisco IOS XE는 애플리케이션 호스팅을 위한 전용 메모리 및 CPU 리소스를 예약하여 스위치의 무결성과 성능을 손상시키지 않으면서 사용자 애플리케이션을 위한 별도의 실행 공간을 제공합니다.

또한 애플리케이션은 특정 Cisco Catalyst 9000 플랫폼에 따라 외부 SSD(Solid-State Drive) 스토리지 옵션(USB 또는 M2 SATA) 중 하나에 상주해야 합니다. 애플리케이션은 Cisco IOS XE에서 무결성을 보호하도록 예약된 내부 디바이스 플래시 스토리지에 액세스할 수 없습니다.

Cisco Catalyst 9000 스위치의 Cisco IOS XE 커널 및 Cisco Application Framework는 유연성을 극대화하고 기본 운영 체제로부터 완벽하게 격리하기 위해 Cgroups(제어 그룹) 및 사용자 네임스페이스를 활용하여 컨테이너화된 애플리케이션을 지원합니다. Cgroup은 애플리케이션의 CPU 및 메모리와 같은 물리적 리소스에 대한 액세스를 제한합니다. Cisco Application Framework는 애플리케이션을 활성화하고 설치할 수 있는 리소스가 충분한지 확인합니다. 하드웨어 리소스를 응용 프로그램에 사용할 수 없는 경우 해당 하드웨어 리소스는 응용 프로그램을 활성화할 수 없으며 관리자에게 관련 메시지가 제공됩니다.

그림 1. Cisco Catalyst 9000 플랫폼의 Cisco Application Framework를 시각적으로 표시합니다



또한 SSD 스토리지는 SSD 스토리지에 대한 AES-256 하드웨어 암호화 및 SSD 스토리지와 스위치 모두에 대한 암호 인증을 제공하여 동급 최고의 보안을 제공합니다.

AES-256 암호화는 하드웨어에서 완전히 수행됩니다. 패스코드 인증을 사용할 경우 패스코드는 SSD와 스위치 모두에 설정되어야 합니다. 패스코드 인증이 미리 구성된 SSD를 일치하는 패스코드 구성이 없는 Catalyst 9000 스위치에 삽입하면, 스위치에 올바른 패스코드가 구성되지 않아 인증이 실패합니다. 그림 2에 표시된 것처럼 성공적인 인증 해제를 위해서는 패스코드가 SSD 스토리지와 스위치 모두에서 일치해야 합니다.

패스코드 구성된 SSD 스토리지를 Catalyst 9000 스위치에서 제거하고 비 Catalyst 스위치에 삽입하면 내용이 보호되고 액세스할 수 없습니다. 모든 민감한 데이터는 올바른 패스코드가 있는 Catalyst 스위치에서 잠금을 해제한 경우에만 액세스할 수 있습니다.

그림 2. SSD 스토리지의 패스코드 인증을 보여줍니다.



- 예를 들어 다음 명령을 사용하여 보안을 활성화 또는 비활성화하고 비밀번호를 설정할 수 있습니다.

```
<#root>
```

```
Cat9k#
```

```
hw-module switch 1 usbflash1 security ?
```

```
disable  disable security on USB3.0
enable   Enable security on USB3.0
unlock   Unlock USB3.0
```

```
Cat9k(config)#hw-module switch 1 usbflash1-password
```

앱 호스팅을 위해 지원되는 Catalyst 모델

애플리케이션 호스팅 기능은 다음과 같이 지원됩니다.

지원되는 플랫폼	Cisco IOS XE 릴리스
Catalyst 9300/L	16.12.1 릴리스 이상
Catalyst 9300X	17.5.1 릴리스 이상
Catalyst 9300LM	17.9.1 릴리스 이상
Catalyst 9404 및 9407	17.1.1 릴리스 이상
Catalyst 9410	17.5.1 릴리스 이상
Catalyst 9400X	17.8.1 릴리스 이상
Catalyst 9500 고성능 및 9600 Series	17.5.1 릴리스 이상
Catalyst 9500X 및 9600X Series	17.8.1 릴리스 이상



참고: Cisco Catalyst 9200 Series 및 Catalyst 9500(UADP 2.0 기반: C9500-40X, C9500-16X, C9500-24Q, C9500-12Q) 모델은 애플리케이션 호스팅을 지원하지 않습니다.

애플리케이션을 위한 하드웨어 리소스

표 1. 애플리케이션용 Cisco Catalyst 9000 플랫폼 하드웨어 리소스가 포함되어 있습니다.

Cisco Models	Networking (AppGig Port)	Memory (RAM)	CPU	Storage
Catalyst 9300X	2x10G	Up to 8 GB	50% of total CPU (2 CPU Cores)	120/240 GB (USB 3.0 SSD)
Catalyst 9300/L/LM	1x1G	2 GB	25% of total CPU (1 CPU Core)	120/240 GB (USB 3.0 SSD)
Catalyst 9400X	2x10G	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)
Catalyst 9400	1x1G	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)
Catalyst 9500X	2x10G	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)
Catalyst 9500 High performance	Management Port*	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)
Catalyst 9600X	Management Port*	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)
Catalyst 9600	Management Port*	Up to 8 GB	25% of total CPU (1 CPU Core)	240 - 960 GB (SATA)

기본 옵션이 충분하지 않으면 리소스, CPU, 메모리 및 vCPU를 사용자 지정 리소스 프로필로 예약할 수 있습니다. 디바이스 리소스 제한은 다음 정보를 참조하십시오.

Platform	Memory (GB)	vCPUs	CPU Units	USB Back Storage (GB)	M2 SATA Storage (GB)
Catalyst 9300	2	2	7400	120	NA
Catalyst 9400	4-10	2	7400	NA	960
Catalyst 9500	8	2	7400	120	NA
Catalyst 9500 High-performance	8	2	7400	NA	960

- vCPU: 이는 단일 애플리케이션에서 동시에 활용할 수 있는 가상 CPU의 최대 수를 나타냅니다.
- CPU 장치: 애플리케이션 호스팅에 할당된 총 CPU 로드 리소스를 나타냅니다. 각 애플리케이션은 안정적인 작동을 보장하는 데 필요한 보장된 최소 CPU 로드를 지정합니다.
- USB 백 스토리지: 후면 패널 오목한 USB 3.0 슬롯을 나타냅니다. 애플리케이션 호스팅은 이 후면 패널 USB 3.0 슬롯에서만 지원됩니다.
- M.2 SATA 스토리지: 애플리케이션 데이터 스토리지에 사용되는 내부 SSD(Solid-State Drive)를 가리킵니다.
- 해당 없음(해당 없음): 기능 또는 리소스를 지정된 구성에 사용할 수 없음을 나타냅니다.
- 이 예에서는 사용자 지정 프로필을 생성하는 방법을 설명합니다.

```
<#root>
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-resource profile custom
```

```
Cat9k(config-app-resource-profile-custom)#
```

```
cpu 7400
```

```
Cat9k(config-app-resource-profile-custom)#
```

```
memory 2048
```

```
Cat9k(config-app-resource-profile-custom)#
```

```
vcpu 2
```

애플리케이션 고가용성

Catalyst 9000 스위치는 시스템 전환 또는 재시작 시 마지막으로 구성된 앱 작동 상태를 유지할 수 있는 애플리케이션 자동 재시작 기능을 지원합니다. 이 기능은 기본적으로 활성화되어 있으며 액티브 스위치와 스탠바이 스위치 모두에서 필요한 스토리지 유형이 동일합니다.

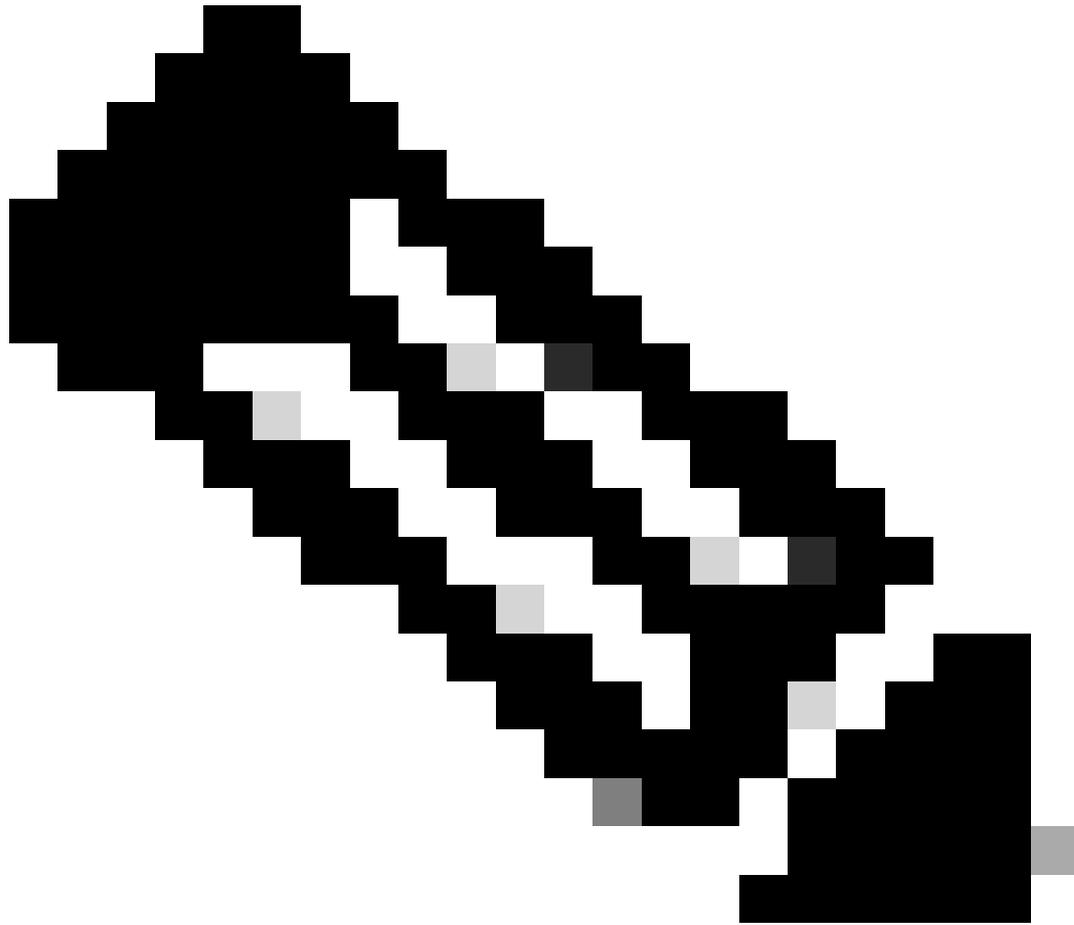
표 2. Cisco Catalyst 9000 플랫폼 앱 자동 재시작 기능

Supported Platforms	IOS XE Release
Catalyst 9300/L StackWise 480/360 (1+1 mode only)	17.2.1
Catalyst 9300LM StackWise 360 (1+1 mode only)	17.9.1
Catalyst 9300X StackWise 1T (1+1 mode only)	17.6.1
Catalyst 9400 Dual Sup (Single Chassis and StackWise Virtual)	17.5.1
Catalyst 9400X Dual Sup (Single Chassis and StackWise Virtual)	17.9.1
Catalyst 9500 High Performance StackWise Virtual	17.5.1
Catalyst 9600 Dual Sup (Single Chassis and StackWise Virtual)	17.5.1
Catalyst 9600X Dual Sup (Single Chassis)	17.9.1

애플리케이션 호스팅 프레임워크

응용 프로그램 호스팅 프레임워크는 기본적으로 활성화되어 있지 않습니다. Cisco CAF(Application Framework)는 모든 네트워크 디바이스에서 실행되는 컨테이너화된 애플리케이션을 관리하기 위해 Cisco에 의해 구축됩니다. CAF는 IOx라고도 합니다. Cisco IOx는 네트워크 에지에서 IoT 애플리케이션을 실행하는 동시에 Cisco IOS 소프트웨어와의 안전한 연결을 보장합니다(포그 컴퓨팅). 애플리케이션 호스팅을 위해 4GB의 디스크 공간을 예약합니다. IOx가 CLI를 통해 비활성화되거나 IOx 인프라가 SSD를 활용할 경우 파티션 및 애플리케이션 데이터가 지워집니다.

- 앱 호스팅 인프라를 활성화하는 Cisco IOS XE CLI 컨피그레이션입니다.



참고: Cisco IOS XE 16.12.1부터는 IoX 패키징이 더 이상 필수 사항이 아닙니다. Docker 앱은 그대로 설치할 수 있습니다. 이를 통해 사용자는 추가 패키징 없이 자체 애플리케이션을 구축하고 가져올 수 있습니다. 애플리케이션 개발자는 [Cisco DevNet](#) 사이트에서 애플리케이션 호스팅에 대한 자세한 정보를 찾을 수 [있습니다](#)

컨테이너 네트워킹

이 그림에서는 컨테이너의 연결 아키텍처를 보여 줍니다. 앱 호스팅을 위한 Cat9k의 모든 가능한 네트워크 모드를 강조 표시합니다. 여기에는 다음이 포함됩니다.

- 내부 전용 AppGigabitEthernet 인터페이스를 사용하는 전면 패널 데이터 포트 스위칭
- 관리 인터페이스(GigabitEthernet 0/0)

AppGigabitEthernet은 하드웨어가 전면 패널 데이터 포트에 스위칭되는 내부 하드웨어 데이터 포트입니다. 컨테이너는 관리 인터페이스와 전면 패널 데이터 포트를 모두 사용하여 연결됩니다. 컨테이너 내에서 vNIC(virtual network interface card)는 eth0, eth1 등으로 레이블이 지정된 표준 이더넷

참고: C9300X, C9400X-Sup-2/2XL 및 C9500X 모델에는 10G AppGigabitEthernet 포트가 2개 있습니다.

애플리케이션 호스팅을 위한 네트워크

- 트렁크 인터페이스의 경우 포트에서 수신하는 모든 트래픽을 앱에서 사용할 수 있습니다

```
<#root>
```

```
Cat9k(config)#
```

```
interface AppGigabitEthernet 1/0/1
```

```
Cat9k(config)#
```

```
switchport mode trunk
```

```
Cat9k(config-if)#exit
```

```
Cat9k(config)#  
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#  
app-vnic AppGigabitEthernet trunk
```

```
Cat9k(config-config-app-hosting-trunk)#  
guest-interface
```

```
Cat9k(config-config-app-hosting-trunk-mode-guest)#  
end
```

- VLAN 인터페이스의 경우 애플리케이션이 특정 VLAN에 연결됨

```
<#root>
```

```
Cat9k(config)#  
interface AppGigabitEthernet 1/0/1
```

```
Cat9k(config)#  
switchport trunk allowed vlan
```

```
Cat9k(config-if)#  
exit
```

```
Cat9k(config)#  
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#  
app-vnic AppGigabitEthernet trunk
```

```
Cat9k(config-config-app-hosting-trunk)#
```

```
vlan
```

```
guest-interface
```

```
Cat9k(config-config-app-hosting-trunk-mode-guest)#  
end
```

- 관리 인터페이스의 경우 애플리케이션이 관리 포트(GigabitEthernet0/0)에 연결됩니다.

```
<#root>
```

```
Cat9k(config)#  
interface gigabitEthernet 0/0
```

```
Cat9k(config-if)#  
ip address
```

```
Cat9k(config-if)#  
exit
```

```
Cat9k(config)#  
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#(config-app-hosting)#  
app-vnic management guest-interface
```

```
Cat9k((config-app-hosting-mgmt-gateway)#  
end
```

앱 컨테이너에 대한 IP 주소 할당: 컨테이너 인터페이스의 IP 주소는 스위치 CLI를 통해 명시적으로 할당되거나 DHCP를 통해 동적으로 얻을 수 있습니다.

- AppGigabitEthernet을 통해 앱의 고정 IP 주소를 구성합니다.

```
<#root>
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-vnic AppGigabitEthernet trunk
```

```
Cat9k(config-config-app-hosting-trunk)#
```

```
vlan
```

```
guest-interface
```

```
Cat9k(config-config-app-hosting-vlan-access-ip)#
```

```
guest-ipaddress
```

netmask

<-- Container IP Address

Cat9k(config-config-app-hosting-vlan-access-ip)#

exit

Cat9k(config-config-app-hosting-trunk)#

exit

Cat9k(config-app-hosting)#

app-default-gateway

guest-interface

Cat9k(config-app-hosting)#

exit

- GigabitEthernet 0/0을 통해 앱의 고정 IP 주소 구성

<#root>

Cat9k(config)#

app-hosting appid MYAPP

Cat9k(config-app-hosting)#

app-vnic management guest-interface

```
Cat9k(config-app-hosting-mgmt-gateway)#
```

```
guest-ipaddress
```

```
netmask
```

```
Cat9k(config-app-hosting-mgmt-gateway)#
```

```
exit
```

```
Cat9k(config-app-hosting)#
```

```
app-default-gateway
```

```
guest-interface
```

```
Cat9k(config-app-hosting)#
```

```
exit
```

- 앱의 동적 IP 주소 (DHCP) 구성

```
<#root>
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-vnic AppGigabitEthernet trunk
```

```
Cat9k(config-config-app-hosting-trunk)#
```

```
vlan
```

```
guest-interface
```

```
Cat9k(config-config-app-hosting-vlan-access-ip)#
```

```
end
```

```
Cat9k#
```

```
Or through Interface
```

```
GigabitEthernet 0/0
```

```
Cat9k(config)#
```

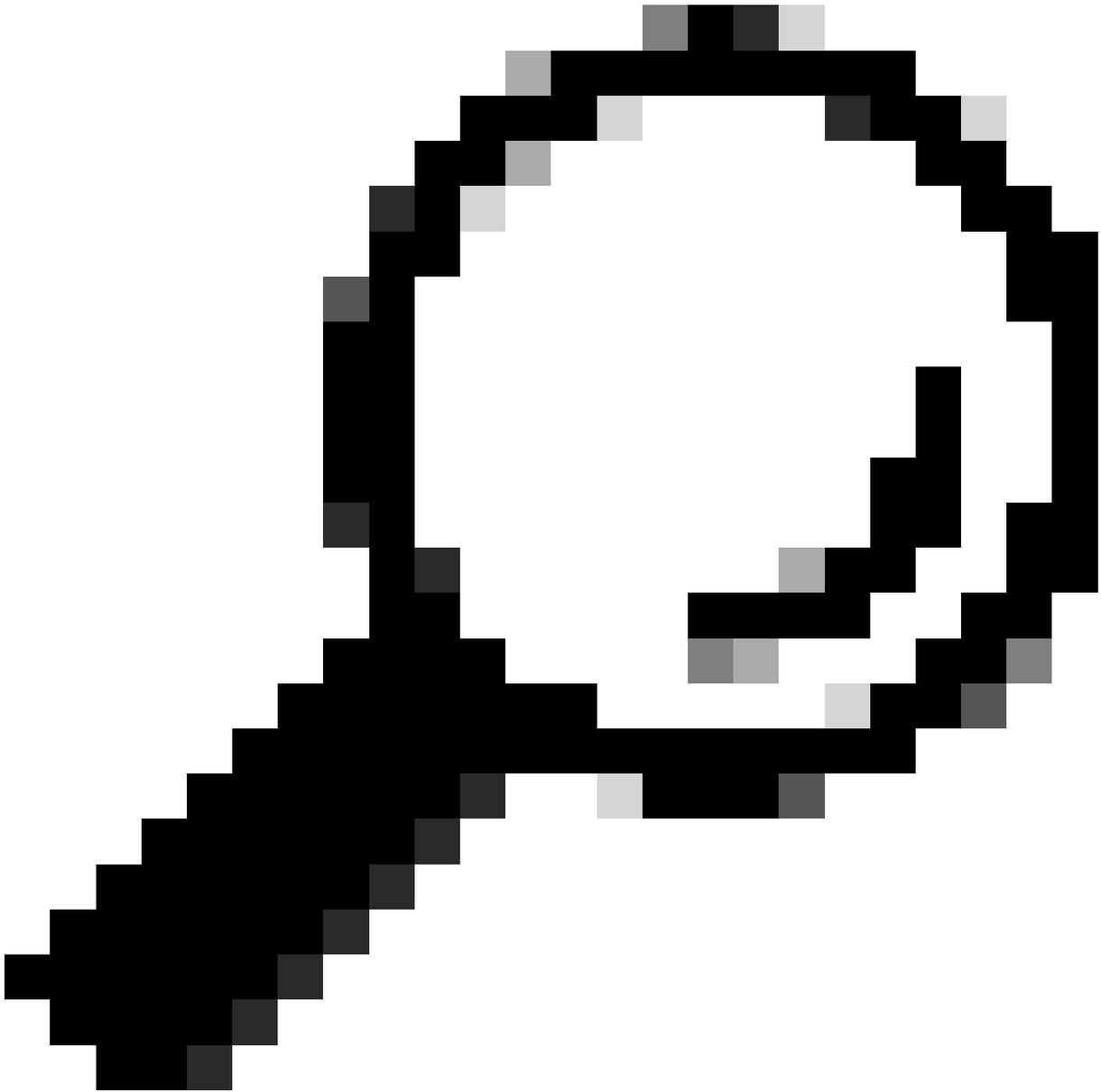
```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-vnic management guest-interface
```

```
Cat9k(config-app-hosting-mgmt-gateway)#
```

```
end
```



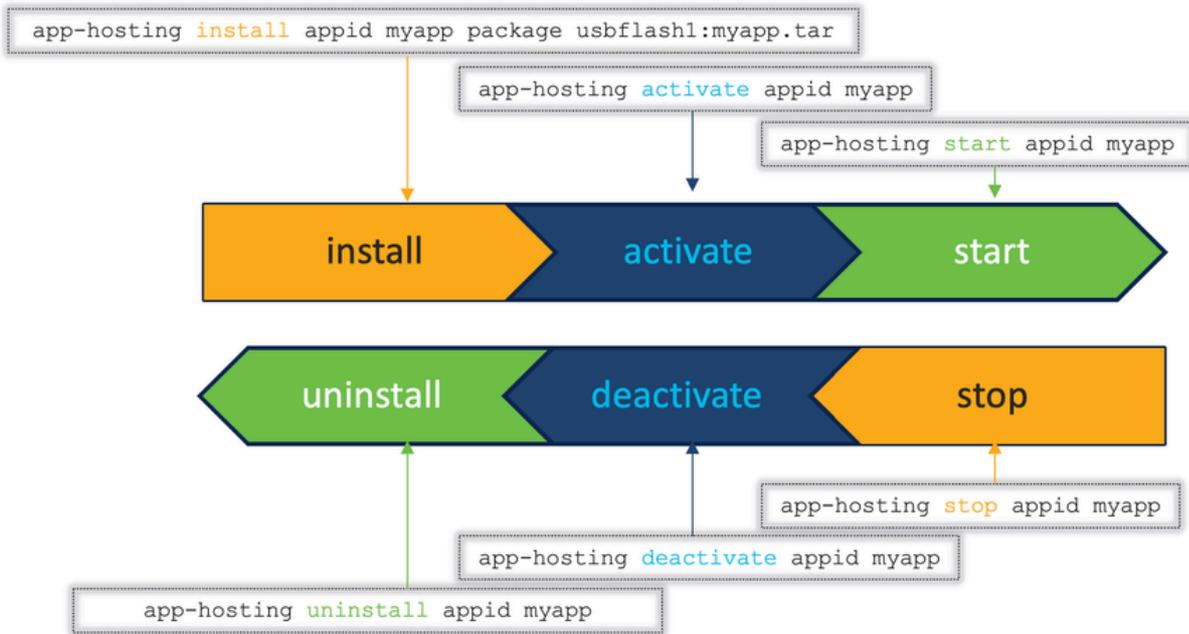
팁: 앱을 설치하기 전에 네트워킹이 올바르게 구성되었는지 확인하십시오. 변경이 필요한 경우 중지, 비활성화 및 제거 후 다시 시작

및 애플리케이션의 라이프사이클

Cisco Catalyst 9000 스위치의 애플리케이션 라이프사이클은 그림 4와 같이 3단계로 구성됩니다.

- 설치: 이 단계에서 애플리케이션이 디바이스에 설치됩니다. 그러나 애플리케이션에 필요한 리소스가 아직 할당되지 않았습니다.
- 활성화: 응용 프로그램에 필요한 하드웨어 리소스는 기능을 보장하기 위해 커밋됩니다.
- 시작하기: 애플리케이션이 작동 상태로 전환되어 완전히 활성화되고 작동합니다.

그림 4. 애플리케이션 수명주기 관리



Docker 런타임 옵션

컨테이너 애플리케이션에 명령줄 옵션(예: 컨트롤러 IP, 데이터 디렉토리 및 기타)으로 전달되는 런타임 옵션이 필요한 경우 "app-resource docker" 명령을 사용하여 이러한 옵션을 구성할 수 있습니다.

시스템은 여러 행의 실행 옵션 문자열 컨피그레이션을 지원합니다. 다음은 사용자를 위한 주요 고려 사항입니다.

- 사용자는 최대 30줄의 docker 실행 옵션을 입력/복사-붙여넣기할 수 있습니다.
- 시스템은 라인 1~30의 문자열에서 연결된 문자열을 해당 순서로 생성합니다.
- 각 run-option 행의 각 문자열은 최대 235자를 가질 수 있습니다.
- 각 문자열은 235자에 모두 포함되는 한 둘 이상의 "완전한" docker 실행 옵션을 가질 수 있습니다.
- 어떤 행에서도 실행 옵션 문자열을 다음 행으로 분할할 수 없습니다.
- 공백으로 각 행의 문자열을 종료할 필요가 없습니다. 시스템은 기존의 모든 실행 옵션 문자열을 단일 문자열로 연결할 때 각 줄에 대한 공백을 자동으로 생성합니다.
- 사용자가 실행 옵션을 변경한 경우 새 실행 옵션을 적용하려면 응용 프로그램을 중지, 비활성화, 활성화 및 다시 시작해야 합니다.

다음은 Cat9k 스위치에서 가능한 Docker 옵션 및 이와 동등한 구성의 예입니다.

```
docker run -v $(APP_DATA):/data --entrypoint startup.sh
```

Catalyst 9000 스위치의 구성:

```
<#root>
```

```
Cat9k(config)#  
app-hosting appid
```

MYAPP

```
Cat9k(config-app-hosting)#  
app-resource docker
```

```
Cat9k(config-app-hosting-docker)#  
run-opts 1 "-v $(APP_DATA):/data"
```

```
Cat9k(config-app-hosting-docker)#  
run-opts 2 "--entrypoint startup.sh"
```

외부 영구 데이터 스토리지의 경우 다음 명령을 사용합니다. run-opts 1 "-v /vol/usb1/iox_host_data_share:/(APP_DATA)"

<#root>

```
Cat9k(config)#  
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#  
app-resource docker
```

```
Cat9k(config-app-hosting-docker)#  
run-opts 1 "-v /vol/usb1/iox_host_data_share:/(APP_DATA)"
```

컨피그레이션에서 실행 옵션 행을 제거하려면 이 예와 유사한 "no" 명령을 실행합니다.

<#root>

```
Cat9k(config)#  
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#  
app-resource docker
```

```
Cat9k(config-app-hosting-docker)#  
no run-opts 1 "-v /vol/usb1/iox_host_data_share:/(APP_DATA)"
```

응용 프로그램 구성의 모든 실행 옵션을 삭제하려면 "no" 구성 명령을 실행할 수 있습니다.

```
<#root>
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
no
```

```
app-resource docker
```

지원되지 않는 Docker 런타임 옵션 목록

나열된 Docker 명령줄 옵션은 보안상의 이유 또는 Linux 플랫폼에 적용할 수 없는 옵션 때문에 Application Hosting Framework에서 지원되지 않습니다.

Docker 옵션	설명
--attach	STDIN, STDOUT 또는 STDERR에 연결합니다.
—블록 가중치 디바이스	블록 IO 가중치(상대 디바이스 가중치)를 지정합니다.
—그룹-상위	컨테이너에 대한 선택적 상위 그룹입니다.
- cidfile	지정된 파일에 컨테이너 ID를 기록합니다.
—cpu-count	CPU 수를 지정합니다(Windows에만 해당).
—cpu 백분율	CPU 백분율을 지정합니다(Windows에만 해당).
—cpu	CPU 수(API 1.25+)를 지정합니다.
—device-cgroup-rule	cgroup allowed devices 목록에 규칙을 추가합니다.

—장치 읽기-bps	디바이스의 읽기 속도(초당 바이트 수)를 제한합니다.
—디바이스 읽기 iops	디바이스의 읽기 속도(초당 I/O)를 제한합니다.
—디바이스-쓰기-bps	디바이스에 대한 쓰기 속도(초당 바이트 수)를 제한합니다.
—디바이스 쓰기 iops	디바이스에 대한 쓰기 속도(초당 I/O)를 제한합니다.
—disable-content-trust	이미지 확인을 건너뛵니다.
—env-file	파일에서 환경 변수를 로드합니다.
—대화형(-i)	STDIN을 연결하지 않더라도 열어 둡니다.
—io-maxbandwidth	시스템 드라이브에 대한 최대 IO 대역폭 제한 (Windows에만 해당).
—io-maxiops	시스템 드라이브에 대한 최대 IOps 제한 (Windows에만 해당)
--ip	IPv4 주소를 지정합니다(예: 192.168.100.100).
—ip6	IPv6 주소 지정(예: 2001:db8::44).
- 격리	컨테이너 격리 기술을 지정합니다.
—링크	다른 컨테이너에 대한 링크를 추가합니다.
--이름	컨테이너에 이름을 할당합니다.
—oomkilldisable	컨테이너에 대한 OOM 킬러를 비활성화합니다.
--pid	사용할 PID 네임스페이스를 지정합니다.

--플랫폼	플랫폼 지정(실험적; API 1.32+).
- 특별 권한	컨테이너에 확장 권한을 부여합니다.
- 런타임	컨테이너에 사용할 런타임을 지정합니다.
—스토리지 선택	컨테이너에 대한 저장소 드라이버 옵션을 정의합니다.
- sysctl	sysctl 옵션을 지정합니다.
—tty (-t)	의사 TTY를 할당합니다.
- 사용자	사용할 사용자 네임스페이스를 지정하십시오.
—uts	사용할 UTS 네임스페이스를 지정하십시오.
—볼륨 드라이버	컨테이너에 대한 선택적 볼륨 드라이버를 지정합니다.

설정

다음 섹션에서는 Catalyst 9000에서 APP를 구성하고 구축하는 방법에 대해 설명합니다

참고: Catalyst 9000 Series 스위치 Application Marketplace에서 원활하게 작동하는 파트너 솔루션 카탈로그를 [살펴보십시오](#)

- Docker 이미지를 다운로드하고 SCP, FTP, TFTP 또는 USB 스토리지를 사용하여 Cisco 스위치에 복사합니다.

```
<#root>
```

```
Cat9k#
```

```
dir usbflash1:/
```

```
Directory of usbflash1:/
```

```
17      -rw-          5843233 Jan 23 2025 20:50:01 +00:00
```

```
MYAPP.tar
```

- 네트워크 매개변수를 구성합니다(이 예에서는 AppGigabitEthernet 인터페이스가 사용되며 VLAN 10에서 192.168.1.10/24를 사용하며 Google 확인자를 사용합니다).

```
<#root>
```

```
Cat9k(config)#
```

```
interface AppGigabitEthernet 1/0/1
```

```
Cat9k(config-if)#
```

```
switchport trunk allowed vlan 10
```

```
Cat9k(config-if)#
```

```
exit
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-vnic
```

```
AppGigabitEthernet
```

```
trunk
```

```
Cat9k(config-config-app-hosting-trunk)#
```

```
vlan 10 guest-interface 0
```

```
Cat9k(config-config-app-hosting-vlan-access-ip)#
```

```
guest-ipaddress 192.168.1.10 netmask 255.255.255.0
```

```
Cat9k(config-config-app-hosting-vlan-access-ip)#
```

```
exit
```

```
Cat9k(config-config-app-hosting-trunk)#
```

```
exit
```

```
Cat9k(config-app-hosting)#
```

```
app-default-gateway 192.168.1.1 guest-interface 0
```

```
Cat9k(config-app-hosting)#
```

```
name-sever 8.8.8.8
```

```
Cat9k(config-app-hosting)#
```

```
exit
```

- 스위치에서 IOx 프레임워크를 활성화하고 모든 서비스가 실행될 때까지 기다립니다.

```
<#root>
```

```
Cat9k(config)#
```

```
iox
```

```
Cat9k#
```

```
show iox-service
```

```
IOx Infrastructure Summary:
```

```
-----  
IOx service (CAF)           : Running  
IOx service (HA)           : Running  
IOx service (IOxman)       : Running  
IOx service (Sec storage)   : Running  
Libvirtd 5.5.0             : Running  
Dockerd v19.03.13-ce       : Running  
Application DB Sync Info    : Available  
Sync Status                 : Disable
```

- 특정 앱에 필요한 Docker 실행 옵션을 설정합니다.

```
<#root>
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-resource docker
```

```
Cat9k(config-app-hosting-docker)#
```

```
run-opts 1 "
```

```
"
```

```
Cat9k(config-app-hosting-docker)#
```

```
run-opts 2 "
```

"

- SSD에서 애플리케이션을 설치하고 구축을 확인합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting install appid MYAPP package usbflash1:MYAPP.tar
```

```
Installing package 'usbflash1:MYAPP.tar' for 'MYAPP'. Use 'show app-hosting list' for progress.
```

```
Cat9k#
```

```
show app-hosting list
```

```
App id                               State
-----
MYAPP
DEPLOYED
```

- 애플리케이션을 활성화하고 상태를 확인합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting activate appid MYAPP
```

```
MYAPP activated successfully
```

```
Current state is: ACTIVATED
```

```
Cat9k#
```

```
show app-hosting list
```

```
App id                               State
-----
MYAPP
ACTIVATED
```

- 애플리케이션을 시작하고 실행 상태를 확인합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting start appid MYAPP
```

```
MYAPP started successfully
```

```
Current state is: RUNNING
```

```
Cat9k#
```

```
show app-hosting list
```

```
App id                               State
-----
MYAPP
RUNNING
```

- 재부팅 시에도 컨피그레이션 변경 사항이 유지되도록 저장합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting start appid MYAPP
```

```
MYAPP started successfully
```

```
Current state is: RUNNING
```

```
Cat9k#
```

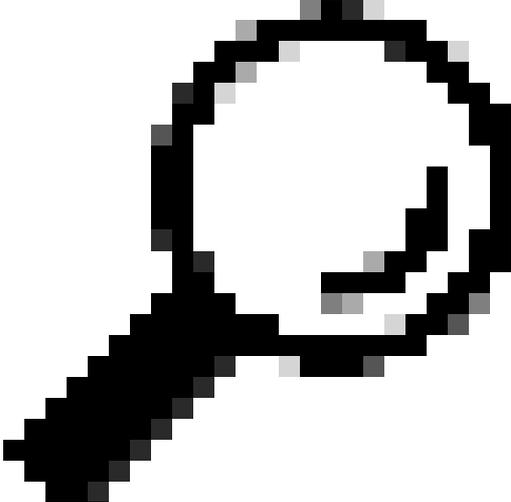
```
show app-hosting list
```

```
App id                               State
-----
MYAPP
RUNNING
```

문제 해결

이 표에는 앱 호스팅 문제를 해결하는 데 사용할 수 있는 다양한 명령이 나열되어 있습니다.

명령을 사용합니다	목적
-----------	----

iox 서비스 표시	인터페이스 오류 카운터를 표시합니다
앱 호스팅 목록 표시	설치된 어플라이언스의 목록을 표시합니다
앱 호스팅 세부 정보 표시 appid MYAPP	어플라이언스에 대한 자세한 정보를 표시합니다.
앱 호스팅 리소스 표시	사용 가능한 리소스를 표시합니다.
show app-hosting utilization appid MYAPP	어플라이언스에 대한 사용률 정보를 표시합니다.
app-hosting appid MYAPP 로그를 bootflash로 이동:	앱별 추적 로그(있는 경우)를 캡처하려면 이 명령을 사용합니다.
<p>애플리케이션 호스팅 시스템 기술 지원을 로 이동</p> <p>부트플래시: 대상 경로 crashinfo-1: 대상 경로 crashinfo: 대상 경로 flash-1: 대상 경로 flash: 대상 경로 웨부이: 대상 경로</p>	이 명령을 사용하여 시스템 techsupport를 대체 디렉토리로 이동합니다.
show app-hosting infrastructure	<p>이 명령을 사용하여 서명 확인이 활성화되었는지 확인합니다.</p>  <p>팁: 응용 프로그램 호스팅이</p>

	USB/SSD를 미디어로 사용하는 경우에만 응용 프로그램 서명 확인을 비활성화할 수 있습니다.
앱 호스팅 확인? 앱 확인 비활성화 비활성화 enable 앱 확인 enable	서명 확인(CLI 권한 모드 형식)을 활성화/비활성화하려면 이 명령을 사용합니다.
app-hosting connect appid MYAPP 세션	이 명령을 사용하여 애플리케이션 콘솔에 액세스하고 컨테이너 내의 프로세스 상태를 확인합니다(CLI 권한 모드에서).

Docker 컨테이너 수정

Docker에 수정이 필요한 경우 다음 절차를 따르십시오.

- 애플리케이션을 중지합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting stop appid MYAPP
```

```
MYAPP stopped successfully
```

```
Current state is:
```

```
STOPPED
```

- 애플리케이션을 비활성화합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting deactivate appid MYAPP
```

```
MYAPP deactivated successfully
```

```
Current state is:
```

```
DEPLOYED
```

- Docker 옵션을 수정합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting start appid MYAPP
```

```
Cat9k(config)#
```

```
app-hosting appid MYAPP
```

```
Cat9k(config-app-hosting)#
```

```
app-resource docker
```

```
Cat9k(config-app-hosting-docker)#
```

```
prepend-pkg-opts
```

```
Cat9k(config-app-hosting-docker)#
```

```
Cat9k(config-app-hosting-docker)#
```

```
exit
```

```
Cat9k(config-app-hosting)#
```

```
exit
```

```
Cat9k(config)#
```

```
exit
```

- 응용 프로그램을 다시 활성화합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting activate appid MYAPP
```

```
MYAPP activated successfully
```

```
Current state is:
```

STOPPED

- 애플리케이션을 시작합니다.

```
<#root>
```

```
Cat9k#
```

```
app-hosting start appid MYAPP
```

```
MYAPP started successfully
```

```
Current state is:
```

```
RUNNING
```

관련 정보

- [Cisco Catalyst 9000 Series 스위치의 애플리케이션 호스팅 백서](#)
- [애플리케이션 마켓플레이스](#)
- [엔터프라이즈에서 애플리케이션 호스팅](#)

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.