

# Catalyst 9000X Series 스위치에서 IPsec 구성

## 목차

---

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[용어](#)

[구성](#)

[네트워크 다이어그램](#)

[HSEC 라이선스 설치](#)

[SVTI 터널 보호](#)

[다음을 확인합니다.](#)

[IPsec 터널](#)

[IOSd 컨트롤 플레인](#)

[PD 컨트롤 플레인](#)

[문제 해결](#)

[IOS d](#)

[PD 컨트롤 플레인](#)

[PD 데이터 플레인](#)

[데이터 플레인 패킷 추적기](#)

[PD 데이터 플레인 디버깅](#)

[관련 정보](#)

---

## 소개

이 문서에서는 Catalyst 9300X 스위치에서 IPsec(Internet Protocol Security) 기능을 확인하는 방법에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- IPSec

### 사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- C9300X
- C9400X
- Cisco IOS® XE 17.6.4 이상

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

Cisco IOS® XE 17.5.1부터 Catalyst 9300-X Series 스위치는 IPsec을 지원합니다. IPsec은 암호화 및 인증을 통해 높은 수준의 보안을 제공할 뿐만 아니라 무단 액세스로부터 데이터를 보호합니다. C9300X의 IPsec 구현은 sVTI(Static Virtual Tunnel Interface) 컨피그레이션을 사용하여 두 피어 간에 보안 터널을 제공합니다.

Catalyst 9400-X Series 스위치에 대한 IPsec 지원은 Cisco IOS® XE 17.10.1에 도입되었으며, Catalyst 9500-X에 대한 지원은 17.12.1로 예정되어 있습니다.

## 용어

IOS d	IOS 데몬	Linux 커널에서 실행되는 Cisco IOS 데몬입니다. 커널 내에서 소프트웨어 프로세스로 실행됩니다. IOS는 상태 및 컨피그레이션을 구축하는 CLI 명령 및 프로토콜을 처리합니다.
피디	플랫폼에 따라 다름	실행 중인 플랫폼과 관련된 데이터 및 명령
IPSec	인터넷 프로토콜 보안	인터넷 프로토콜 네트워크를 통해 두 컴퓨터 간에 암호화된 보안 통신을 제공하기 위해 데이터의 패킷을 인증 및 암호화하는 보안 네트워크 프로토콜 제품군입니다.
SVTI	정적 가상 터널 인터페이스	보안 기능을 적용할 수 있는 정적으로 구성된 가상 인터페이스
SA	보안 연계	SA는 두 개 이상의 엔터티 간의 관계이며, 보안 서비스를 사용하여 안전하게 통신하는 방법을 설명합니다
연방	포워딩 엔진 드라이버	UADP ASIC의 하드웨어 프로그래밍을 담당하는 스위치 구성 요소입니다.

## 구성

## 네트워크 다이어그램

이 예의 목적상 Catalyst 9300X 및 ASR1001-X는 IPsec 가상 터널 인터페이스를 사용하는 IPsec 피어로 작동합니다.



## HSEC 라이선스 설치

Catalyst 9300X 플랫폼에서 IPsec 기능을 활성화하려면 HSEC 라이선스(C9000-HSEC)가 필요합니다. 이는 허용된 암호화 처리량을 늘리기 위해서만 HSEC 라이선스가 필요한 IPsec을 지원하는 다른 Cisco IOS XE 기반 라우팅 플랫폼과는 다릅니다. Catalyst 9300X 플랫폼에서 HSEC 라이선스가 설치되지 않은 경우 터널 모드 및 터널 보호 CLI가 차단됩니다.

```
<#root>
```

```
C9300X(config)#
```

```
int tunnel1
```

```
C9300X(config-if)#
```

```
tunnel mode ipsec ipv4
```

```
%'tunnel mode' change not allowed
```

```
*Sep 19 20:54:41.068: %PLATFORM_IPSEC_HSEC-3-INVALID_HSEC: HSEC
```

```
license not present: IPSec mode configuration is rejected
```

스위치가 Smart Licensing을 사용하여 CSSM 또는 CSLU에 연결될 때 HSEC 라이선스를 설치합니다.

```
<#root>
```

```
C9300X#
```

```
license smart authorization request add hseck9 local
```

```
*Oct 12 20:01:36.680: %SMART_LIC-6-AUTHORIZATION_INSTALL_SUCCESS: A new licensing authorization code wa
```

HSEC 라이선스가 올바르게 설치되었는지 확인합니다.

```
<#root>
```

```
C9300X#
```

```
show license summ
```

```
Account Information:
```

```
Smart Account: Cisco Systems, TAC As of Oct 13 15:50:35 2022 UTC
```

```
Virtual Account: CORE TAC
```

```
License Usage:
```

License	Entitlement Tag	Count	Status
network-advantage	(C9300X-12Y Network Adv...)	1	IN USE
dna-advantage	(C9300X-12Y DNA Advantage)	1	IN USE
C9K HSEC	(Cat9K HSEC)	0	

```
NOT IN USE
```

터널 인터페이스에서 터널 모드로 IPsec을 활성화합니다.

```
<#root>
```

```
C9300X(config)#
```

```
int tunnel1
```

```
C9300X(config-if)#
```

```
tunnel mode ipsec ipv4
```

```
C9300X(config-if)#
```

```
end
```

IPsec을 활성화하면 HSEC 라이선스가 사용 중으로 됩니다.

```
<#root>
```

```
C9300X#
```

```
show license summ
```

Account Information:

Smart Account: Cisco Systems, TAC As of Oct 13 15:50:35 2022 UTC  
Virtual Account: CORE TAC

License Usage:

License	Entitlement Tag	Count	Status
network-advantage	(C9300X-12Y Network Adv...)	1	IN USE
dna-advantage	(C9300X-12Y DNA Advantage)	1	IN USE
C9K HSEC	(Cat9K HSEC)	1	

IN USE

## SVTI 터널 보호

C9300X의 IPsec 컨피그레이션에서는 표준 Cisco IOS XE IPsec 컨피그레이션을 사용합니다. 이는 IKEv2 [Smart Defaults](#)를 사용하는 간단한 SVTI 컨피그레이션이며, 여기서는 IKEv2에 대한 기본 IKEv2 정책, IKEv2 제안, IPsec 변환 및 IPsec 프로필을 사용합니다.

### C9300X 구성

```
<#root>
ip routing


!
crypto ikev2 profile default

match identity remote address 192.0.2.2 255.255.255.255
authentication remote pre-share key cisco123
authentication local pre-share key cisco123
!
interface Tunnel1

ip address 192.168.1.1 255.255.255.252
tunnel source 198.51.100.1
tunnel mode ipsec ipv4
tunnel destination 192.0.2.2

tunnel protection ipsec profile default
```

---

 참고: Catalyst 9300X는 기본적으로 액세스 레이어 스위치이므로 VTI와 같은 라우팅 기반 기능이 작동하려면 ip 라우팅을 명시적으로 활성화해야 합니다.

---

### 피어 컨피그레이션

```
<#root>
```

```
crypto ikev2 profile default
```

```
match identity remote address 198.51.100.1 255.255.255.255
authentication remote pre-share key cisco123
authentication local pre-share key cisco123
!
```

```
interface Tunnel1
```

```
ip address 192.168.1.2 255.255.255.252
tunnel source 192.0.2.2
tunnel mode ipsec ipv4
tunnel destination 198.51.100.1

tunnel protection ipsec profile default
```

다양한 IKEv2 및 IPsec 컨피그레이션 구조에 대한 자세한 내용은 [C9300X IPsec 컨피그레이션 가이드를 참조하십시오.](#)

## 다음을 확인합니다.

### IPsec 터널

C9300X 플랫폼의 IPsec 구현은 QFP(Quantum Flow Processor) 마이크로코드에 IPsec 기능 처리가 구현된 라우팅 플랫폼(ASR1000, ISR4000, Catalyst 8200/8300 등)의 아키텍처와는 다릅니다.

C9300X 포워딩 아키텍처는 UADP ASIC를 기반으로 하므로 대부분의 QFP 기능 FIA 구현은 여기에 적용되지 않습니다.

주요 차이점은 다음과 같습니다.

- show crypto ipsec sa peer x.x.x.x platform은 FMAN에서 QFP까지의 플랫폼 프로그래밍 정보를 표시하지 않습니다.
- 패킷 추적도 작동하지 않습니다(자세한 내용은 아래 참조).
- UADP ASIC는 암호화 트래픽 분류를 지원하지 않으므로 show crypto ruleset platform이 적용되지 않습니다

### IOSd 컨트롤 플레인

IPsec 컨트롤 플레인 검증은 라우팅 플랫폼의 검증과 정확히 동일합니다(참고). IOSd에 설치된 IPsec SA를 표시하려면

```
<#root>
```

```
C9300X#
```

```
show crypto ipsec sa
```

```
interface: Tunnel1
```

Crypto map tag: Tunnel1-head-0, local addr 198.51.100.1

protected vrf: (none)

local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)

remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)

current\_peer 192.0.2.2 port 500

PERMIT, flags={origin\_is\_acl,}

#pkts encaps: 200, #pkts encrypt: 200, #pkts digest: 200

#pkts decaps: 200, #pkts decrypt: 200, #pkts verify: 200

#pkts compressed: 0, #pkts decompressed: 0

#pkts not compressed: 0, #pkts compr.

failed: 0

#pkts not decompressed: 0, #pkts decompress failed: 0

#send errors 0, #recv errors 0

local crypto endpt.: 198.51.100.1, remote crypto endpt.: 192.0.2.2

plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb TwentyFiveGigE1/0/1

current outbound spi: 0x42709657(1114674775)

PFS (Y/N): N, DH group: none

**inbound esp sas:**

spi: 0x4FE26715(1340237589)

transform: esp-aes esp-sha-hmac ,

in use settings ={Tunnel, }

conn id: 2098,

flow\_id: CAT9K:98

, sibling\_flags FFFFFFFF80000048, crypto map: Tunnel1-head-0

sa timing: remaining key lifetime (k/sec): (26/1605)

IV size: 16 bytes

replay detection support: Y

Status: ACTIVE(ACTIVE)

inbound ah sas:

inbound pcp sas:

**outbound esp sas:**

spi: 0x42709657(1114674775)

transform: esp-aes esp-sha-hmac ,

in use settings ={Tunnel, }

conn id: 2097,

flow\_id: CAT9K:97

, sibling\_flags FFFFFFFF80000048, crypto map: Tunnel1-head-0

sa timing: remaining key lifetime (k/sec): (32/1605)

IV size: 16 bytes

replay detection support: Y

Status: ACTIVE(ACTIVE)

outbound ah sas:

outbound pcp sas:

출력에서 flow\_id를 확인합니다. 이 값은 전달 평면에 설치된 flow id와 일치해야 합니다.

## PD 컨트롤 플레인

IOSd와 PD 컨트롤 플레인 간의 통계

<#root>

C9300X#

show platfor software ipsec policy statistics

PAL CMD	REQUEST	REPLY OK	REPLY ERR	ABORT
SADB_INIT_START	3	3	0	0
SADB_INIT_COMPLETED	3	3	0	0
SADB_DELETE	2	2	0	0
SADB_ATTR_UPDATE	4	4	0	0
SADB_INTF_ATTACH	3	3	0	0
SADB_INTF_UPDATE	0	0	0	0
SADB_INTF_DETACH	2	2	0	0
ACL_INSERT	4	4	0	0
ACL_MODIFY	0	0	0	0
ACL_DELETE	3	3	0	0
PEER_INSERT	7	7	0	0
PEER_DELETE	6	6	0	0
SPI_INSERT	39	37	2	0
SPI_DELETE	36	36	0	0
CFLOW_INSERT	5	5	0	0
CFLOW_MODIFY	33	33	0	0
CFLOW_DELETE	4	4	0	0
IPSEC_SA_DELETE	76	76	0	0
TBAR_CREATE	0	0	0	0
TBAR_UPDATE	0	0	0	0
TBAR_REMOVE	0	0	0	0
	0	0	0	0
PAL NOTIFY	RECEIVE	COMPLETE	PROC ERR	IGNORE
NOTIFY_RP	0	0	0	0
SA_DEAD	0	0	0	0
SA_SOFT_LIFE	46	46	0	0
IDLE_TIMER	0	0	0	0
DPD_TIMER	0	0	0	0
INVALID_SPI	0	0	0	0
	0	5	0	0
VTI SADB	0	33	0	0
TP SADB	0	40	0	0

IPSec PAL database summary:

DB NAME	ENT ADD	ENT DEL	ABORT
PAL_SADB	3	2	0
PAL_SADB_ID	3	2	0
PAL_INTF	3	2	0
PAL_SA_ID	76	74	0
PAL_ACL	0	0	0
PAL_PEER	7	6	0
PAL_SPI	39	38	0
PAL_CFLOW	5	4	0
PAL_TBAR	0	0	0



## SADB 개체 테이블

<#root>

C9300X#

```
show plat software ipsec switch active f0 sadb all
```

IPsec SADB object table:

SADB-ID	Hint	Complete	#RefCnt	#CfgCnt	#ACL-Ref
3	vir-tun-int	true	2	0	0

## SADB 항목

<#root>

C9300X#

```
show plat software ipsec switch active f0 sadb identifier 3
```

```
===== SADB id: 3
          hint: vir-tun-int
          completed: true
reference count: 2
configure count: 0
ACL reference: 0
```

```
SeqNo (Static/Dynamic)      ACL id
-----
```

## IPsec 흐름 정보

<#root>

C9300X#

```
show plat software ipsec switch active f0 flow all
```

```
=====
```

Flow id: 97

```
          mode: tunnel
          direction: outbound
          protocol: esp
             SPI: 0x42709657
local IP addr: 198.51.100.1
remote IP addr: 192.0.2.2
crypto map id: 0
```

```
        SPD id: 3
        cpp SPD id: 0
    ACE line number: 0
        QFP SA handle: INVALID
    crypto device id: 0
    IOS XE interface id: 65
        interface name: Tunnel1
        use path MTU: FALSE
        object state: active
    object bind state: new
=====
```

Flow id: 98

```
        mode: tunnel
    direction: inbound
    protocol: esp
        SPI: 0x4fe26715
    local IP addr: 198.51.100.1
    remote IP addr: 192.0.2.2
    crypto map id: 0
        SPD id: 3
        cpp SPD id: 0
    ACE line number: 0
        QFP SA handle: INVALID
    crypto device id: 0
    IOS XE interface id: 65
        interface name: Tunnel1
        object state: active
```

## 문제 해결

### IOS d

이러한 debug 및 show 명령은 일반적으로 수집됩니다.

```
<#root>
```

```
show crypto eli all
```

```
show crypto socket
```

```
show crypto map
```

```
show crypto ikev2 sa detail
```

```
show crypto ipsec sa
```

```
show crypto ipsec internal
```

```
<#root>
```

```
debug crypto ikev2
```

```
debug crypto ikev2 error
```

```
debug crypto ikev2 packet
```

```
debug crypto ipsec
```

```
debug crypto ipsec error
```

```
debug crypto kmi
```

```
debug crypto socket
```

```
debug tunnel protection
```

## PD 컨트롤 플레인

PD 컨트롤 플레인 작업을 확인하려면 이전에 표시된 확인 단계를 사용하십시오. PD 제어 플레인과 관련된 문제를 디버깅하려면 PD 제어 플레인 디버깅을 활성화합니다.

1. 자세한 정보를 보려면 블로그 기록 수준을 높입니다.

```
<#root>
```

```
C9300X#
```

```
set platform software trace forwarding-manager switch active f0 ipsec verbose
```

```
C9300X#
```

```
show platform software trace level forwarding-manager switch active f0 | in ipsec
```

```
ipsec
```

```
Verbose
```

2. PD controlplane 조건부 디버깅을 활성화합니다.

```
<#root>
```

```
C9300X#
```

```
debug platform condition feature ipsec controlplane submode level verbose
```

```
C9300X#
```

```
show platform conditions
```

```
Conditional Debug Global State: Stop
```

Feature	Type	Submode	Level
IPSEC			
	controlplane	N/A	
verbose			

3. fman\_fp btrace 출력에서 디버그 출력을 수집합니다.

```
<#root>
```

```
C9300X#
```

```
show logging process fman_fp module ipsec internal
```

```
Logging display requested on 2022/10/19 20:57:52 (UTC) for Hostname: [C9300X], Model: [C9300X-24Y], Ver
```

```
Displaying logs from the last 0 days, 0 hours, 10 minutes, 0 seconds
```

```
executing cmd on chassis 1 ...
```

```
Unified Decoder Library Init .. DONE
```

```
Found 1 UTF Streams
```

```
2022/10/19 20:50:36.686071658 {fman_fp_F0-0}{1}: [ipsec] [22441]: (ERR): IPSEC-PAL-IB-Key::
```

```
2022/10/19 20:50:36.686073648 {fman_fp_F0-0}{1}: [ipsec] [22441]: (ERR): IPSEC-b0 d0 31 04 85 36 a6 08
```

## PD 데이터 플레인

HMAC 또는 재생 실패와 같은 일반 IPsec 삭제를 포함한 데이터 플레인 IPsec 터널 통계를 확인합니다.

```
<#root>
```

```
C9300X#
```

```
show platform software fed sw active ipsec counters if-id all
```

#####

Flow Stats for if-id 0x41

#####

-----  
Inbound Flow Info for

flow id: 98

-----  
SA Index: 1

-----  
Asic Instance 0: SA Stats

Packet Format Check Error:	0
Invalid SA:	0
Auth Fail:	0
Sequence Number Overflows:	0
Anti-Replay Fail:	0
Packet Count:	200
Byte Count:	27600

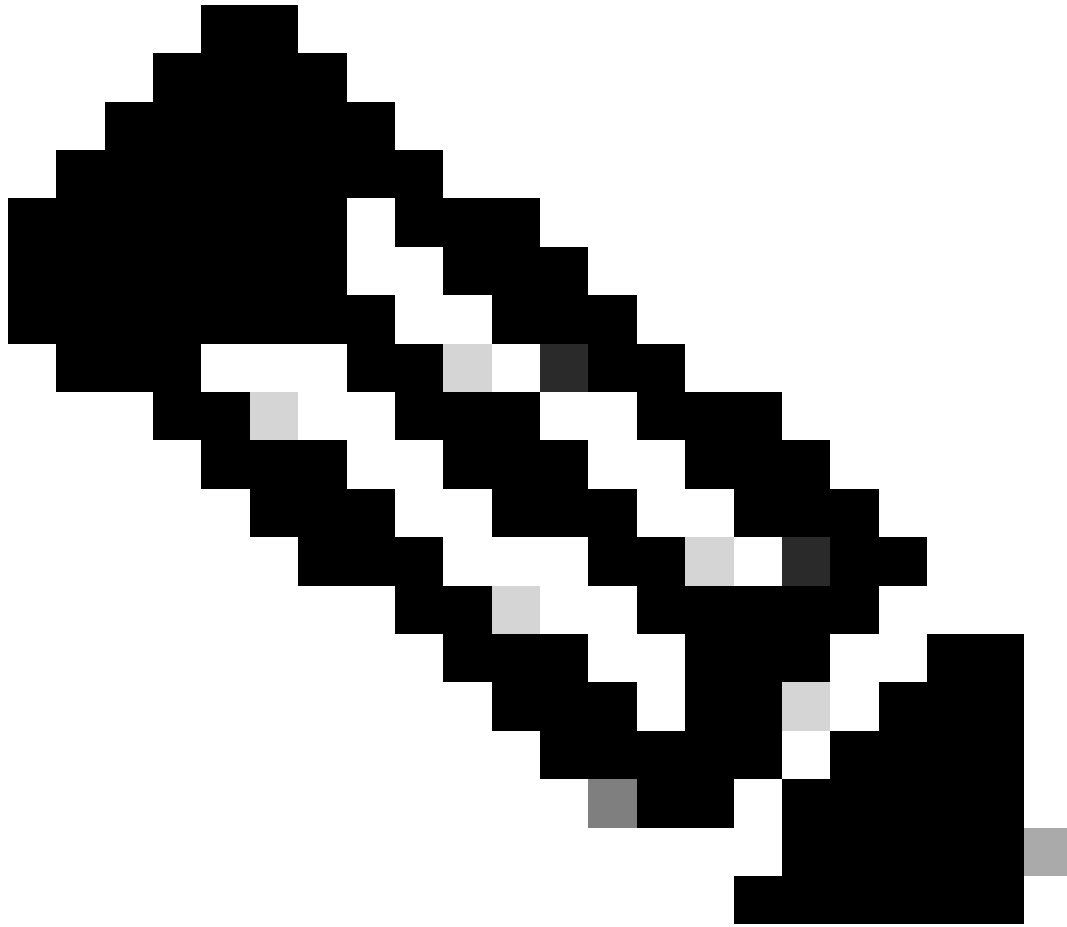
-----  
Outbound Flow Info for

flow id: 97

-----  
SA Index: 1025

-----  
Asic Instance 0: SA Stats

Packet Format Check Error:	0
Invalid SA:	0
Auth Fail:	0
Sequence Number Overflows:	0
Anti-Replay Fail:	0
Packet Count:	200
Byte Count:	33600



참고: 흐름 ID는 show crypto ipsec sa 출력의 흐름 ID와 일치합니다. 개별 흐름 통계는 show platform software fed switch active ipsec counters sa <sa\_id> 명령을 사용하여 얻을 수도 있습니다. 여기서 sa\_id는 이전 출력의 SA Index입니다.

## 데이터 플레인 패킷 추적기

UADP ASIC 플랫폼의 패킷 추적기는 QFP 기반 시스템의 패킷 추적기와 매우 다르게 작동합니다. 수동 트리거 또는 PCAP 기반 트리거를 사용하여 활성화할 수 있습니다. 다음은 PCAP(EPC) 기반 트리거를 사용하는 예입니다.

1. EPC를 활성화하고 캡처를 시작합니다.

```
<#root>
```

```
C9300X#
```

```
monitor capture test interface twentyFiveGigE 1/0/2 in match ipv4 10.1.1.2/32 any
```

<#root>

C9300X#

**show monitor capture test**

Status Information for Capture test

Target Type:

Interface: TwentyFiveGigE1/0/2, Direction: IN

Status : Inactive

Filter Details:

IPv4

Source IP: 10.1.1.2/32

Destination IP: any

Protocol: any

Buffer Details:

Buffer Type: LINEAR (default)

Buffer Size (in MB): 10

File Details:

File not associated

Limit Details:

Number of Packets to capture: 0 (no limit)

Packet Capture duration: 0 (no limit)

Packet Size to capture: 0 (no limit)

Maximum number of packets to capture per second: 1000

Packet sampling rate: 0 (no sampling)

2. 나머지를 실행하고 캡처를 중지합니다.

<#root>

C9300X#

**monitor capture test start**

Started capture point : test

\*Oct 18 18:34:09.656: %BUFCAP-6-ENABLE: Capture Point test enabled.

<run traffic test>

C9300X#

**monitor capture test stop**

Capture statistics collected at software:

Capture duration - 23 seconds

Packets received - 5

Packets dropped - 0

Packets oversized - 0

Bytes dropped in asic - 0

Capture buffer will exist till exported or cleared

Stopped capture point : test

3. 캡처를 플래시로 내보내기

```
<#root>
```

```
C9300X#
```

```
show monitor capture test buff
```

```
*Oct 18 18:34:33.569: %BUFCAP-6-DISABLE
```

```
Starting the packet display ..... Press Ctrl + Shift + 6 to exit
```

```
 1  0.000000    10.1.1.2 -> 10.2.1.2    ICMP 114 Echo (ping) request id=0x0003, seq=0/0, ttl=255
 2  0.000607    10.1.1.2 -> 10.2.1.2    ICMP 114 Echo (ping) request id=0x0003, seq=1/256, ttl=2
 3  0.001191    10.1.1.2 -> 10.2.1.2    ICMP 114 Echo (ping) request id=0x0003, seq=2/512, ttl=2
 4  0.001760    10.1.1.2 -> 10.2.1.2    ICMP 114 Echo (ping) request id=0x0003, seq=3/768, ttl=2
 5  0.002336    10.1.1.2 -> 10.2.1.2    ICMP 114 Echo (ping) request id=0x0003, seq=4/1024, ttl=
```

```
C9300X#
```

```
monitor capture test export location flash:test.pcap
```

#### 4. 패킷추적기를 실행합니다.

```
<#root>
```

```
C9300X#
```

```
show platform hardware fed switch 1 forward interface TwentyFiveGigE 1/0/2 pcap flash:test.pcap number 1
```

```
Show forward is running in the background. After completion, syslog will be generated.
```

```
C9300X#
```

```
*Oct 18 18:36:56.288: %SHFWD-6-PACKET_TRACE_DONE: Switch 1 F0/0: fed: Packet Trace Complete: Execute (
```

```
*Oct 18 18:36:56.288: %SHFWD-6-PACKET_TRACE_FLOW_ID: Switch 1 F0/0: fed: Packet Trace Flow id is 131077
```

```
C9300X#
```

```
C9300X#show plat hardware fed switch 1 forward last summary
```

```
Input Packet Details:
```

```
###[ Ethernet ]###
```

```
dst      = b0:8b:d0:8d:6b:d6
```

```
src=78:ba:f9:ab:a7:03
```

```
type     = 0x800
```

```
###[ IP ]###
```

```
version  = 4
```

```
ihl      = 5
```

```
tos      = 0x0
```

```
len      = 100
```

```
id       = 15
```

```
flags    =
```

```
frag     = 0
```

```
ttl      = 255
```

```
proto    = icmp
```

```
chksum   = 0xa583
```

```
src=10.1.1.2
```

```
dst      = 10.2.1.2
```

```
options  = ''
```

```
###[ ICMP ]###
```

```
type     = echo-request
```

```
code     = 0
```

```
chksum   = 0xae17
```

```
id       = 0x3
```

```
seq      = 0x0
```





```

L3 Interface          : 38
  IPv4 Routing        : enabled
  IPv6 Routing        : enabled
  Vrf Id              : 0
Adjacency:
  Station Index       : 177
  Destination Index   : 21304
  Rewrite Index       : 21
  Replication Bit Map : 0x1    ['remoteData']
Decision:
  Destination Index   : 21304
  Rewrite Index       : 21
  Dest Mod Index      : 0      [IGR_FIXED_DMI_NULL_VALUE]
  CPU Map Index       : 0      [CMI_NULL]
  Forwarding Mode     : 3      [Other or Tunnel]
  Replication Bit Map :        ['remoteData']
  Winner              :        L3FWDIPV4_LOOKUP
  Qos Label           : 1
  SGT                 : 0
  DGTID               : 0

```

```

Egress:
  Possible Replication :
    Port                : TwentyFiveGigE1/0/1
  Output Port Data    :
    Port                : TwentyFiveGigE1/0/1
    Global Port Number  : 1
    Local Port Number   : 1
    Asic Port Number    : 0
    Asic Instance       : 1
    Unique RI           : 0
    Rewrite Type        : 0      [Unknown]
    Mapped Rewrite Type : 13    [L3_UNICAST_IPV4_PARTIAL]
    Vlan                : 0
    Mapped Vlan ID     : 0

```

```

Output Packet Details:
  Port                : TwentyFiveGigE1/0/1

```

```

###[ Ethernet ]###
  dst      = 00:62:ec:da:e0:02
  src=b0:8b:d0:8d:6b:e4
  type     = 0x800

```

```

###[ IP ]###
  version  = 4
  ihl      = 5
  tos      = 0x0
  len      = 168
  id       = 2114
  flags    = DF
  frag     = 0
  ttl      = 254
  proto    = ipv6_crypt
  checksum = 0x45db
  src=198.51.100.1
  dst      = 192.0.2.2
  options  = ''

```

```

###[ Raw ]###      load      = '

```

```
6D 18 45 C9
```

```

00 00 00 06 09 B0 DC 13 11 FA DC F8 63 98 51 98 33 11 9C C0 D7 24 BF C2 1C 45 D3 1B 91 0B 5F B4 3A C0
*****

```

```
C9300X#
```

```
show crypto ipsec sa | in current outbound
```

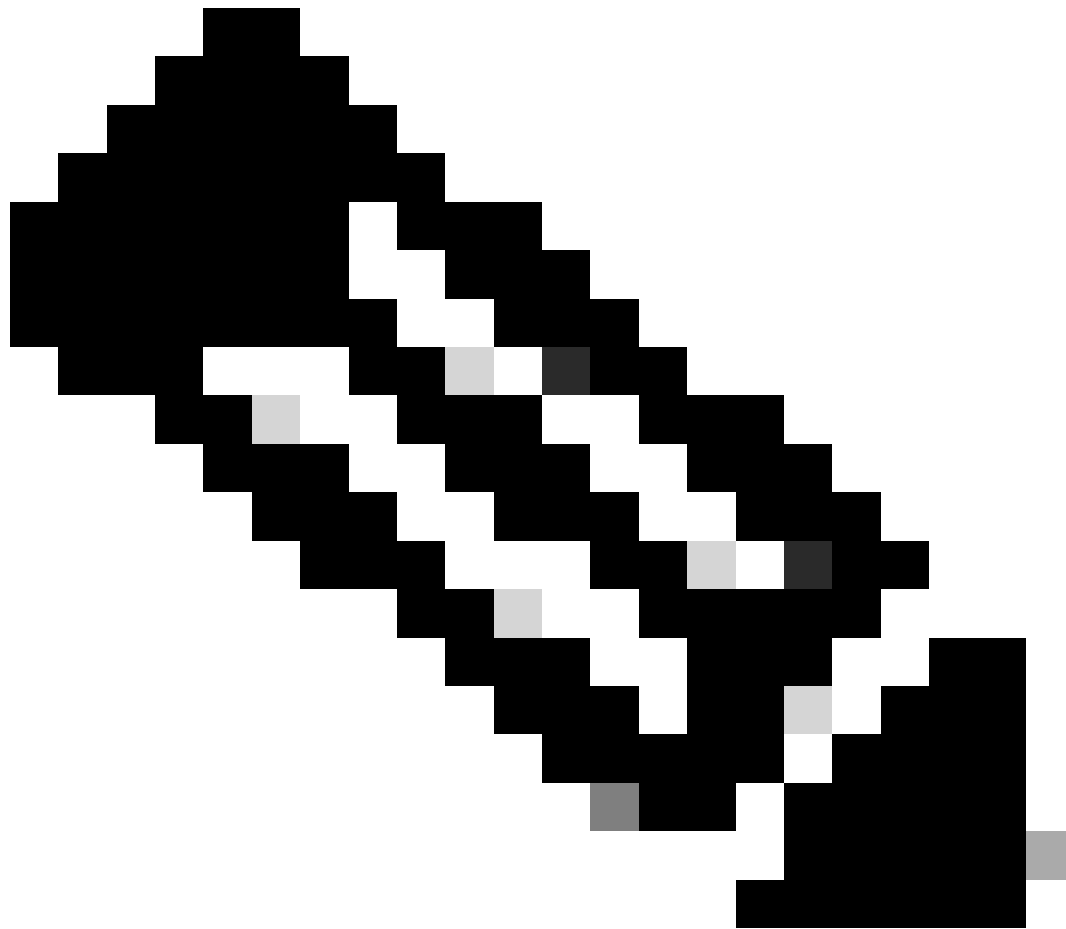
```
current outbound spi:
```

```
0x6D1845C9
```

```
(1830307273)
```

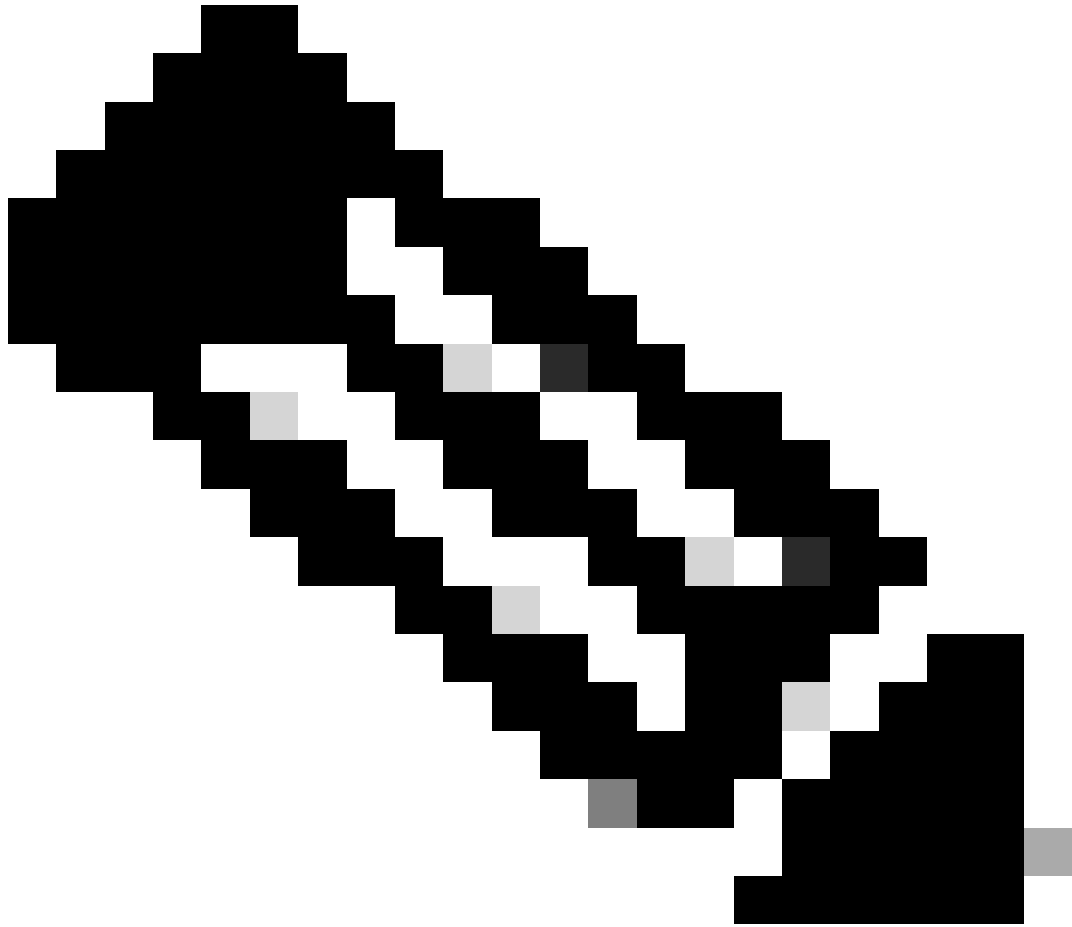
```
<-- Matches the load result in packet trace
```

---



참고: 이전 출력에서 전달된 패킷은 현재 아웃바운드 SA SPI를 사용하는 ESP 패킷입니다. 더 자세한 FED 포워딩 결정 분석을 위해 동일한 명령의 세부 변형. 예: show plat hardware fed switch 1 forward last detail can use.

---



참고: PD 데이터 플레인 디버깅은 TAC의 지원을 통해서만 활성화해야 합니다. 정상적인 CLI/디버그를 통해 문제를 식별할 수 없는 경우 엔지니어링에서 필요로 하는 매우 낮은 수준의 추적입니다.

---

```
<#root>
```

```
C9300X#
```

```
set platform software trace fed switch active ipsec verbose
```

C9300X#

```
debug platform condition feature ipsec dataplane submode all level verbose
```

C9300X#

```
show logging process fed module ipsec internal
```

**IPsec PD SHIM 디버깅**

<#root>

```
debug platform software ipsec info
```

```
debug platform software ipsec error
```

```
debug platform software ipsec verbose
```

```
debug platform software ipsec all
```

## 관련 정보

- [Catalyst 9300 스위치에서 IPsec 구성](#)

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.