

SD-WAN CLI 템플릿에서 ZBFW 구성

목차

- [소개](#)
 - [사전 요구 사항](#)
 - [요구 사항](#)
 - [사용되는 구성 요소](#)
 - [배경 정보](#)
 - [구성](#)
 - [네트워크 다이어그램](#)
 - [설정](#)
 - [컨트롤 플레인](#)
 - [데이터 플레인](#)
 - [다음을 확인합니다.](#)
-

소개

이 문서에서는 Cisco Catalyst SD-WAN Manager에서 CLI 애드온 기능 템플릿을 사용하여 ZBFW(Zone-Based Firewall) 정책을 구성하는 방법에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco Catalyst SD-WAN(Software-Defined Wide Area Network)
- ZBFW(Zone-Based Firewall) 기본 운영

사용되는 구성 요소

- Cisco Catalyst SD-WAN Manager 20.9.3.2
- Cisco IOS® XE Catalyst SD-WAN Edge 17.6.5a

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.

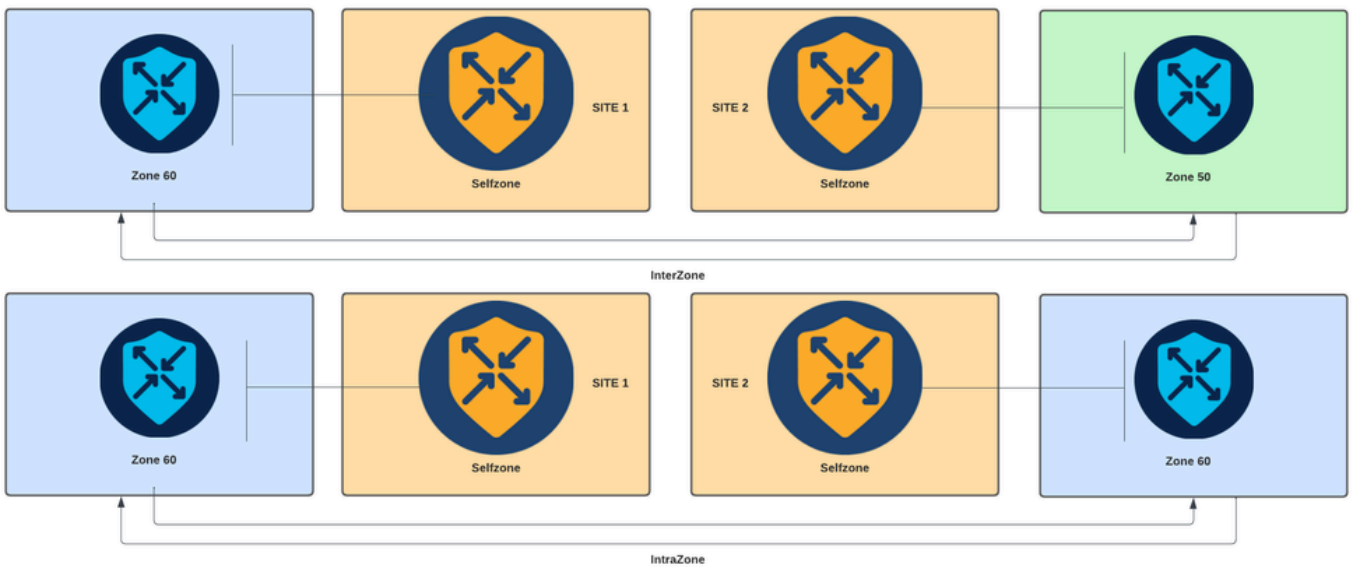
배경 정보

방화벽 정책은 TCP, UDP 및 ICMP 데이터 트래픽 흐름의 상태 저장 검사를 허용하는 지역화된 보안 정책의 한 유형입니다. 그것은 지역이라는 개념을 사용합니다. 따라서 지정된 영역에서 시작된

트래픽 흐름이 두 영역 간의 정책에 따라 다른 영역으로 진행될 수 있습니다.

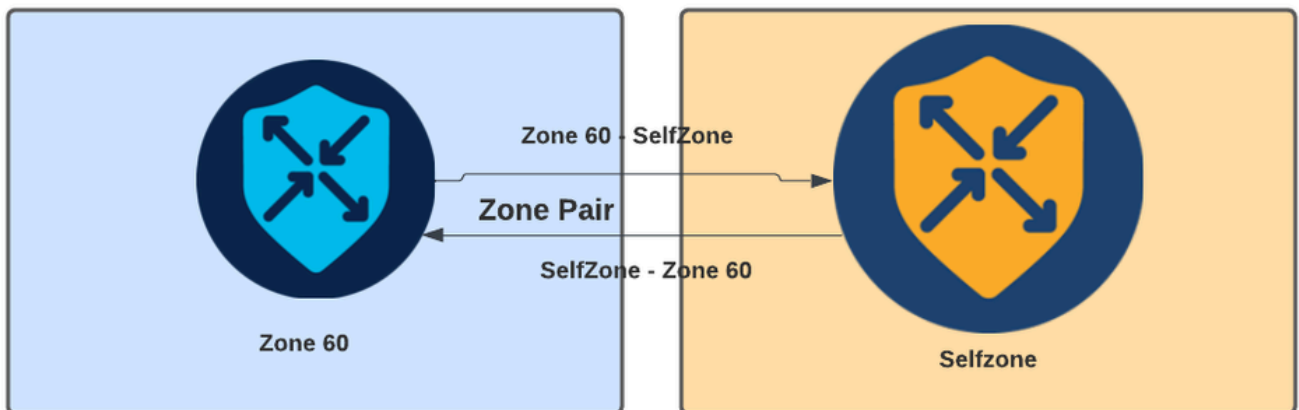
영역은 하나 이상의 VPN의 그룹입니다. ZBFW에 있는 영역의 유형은 다음과 같습니다.

- 소스 영역: 데이터 트래픽 흐름을 시작하는 VPN 그룹. VPN은 하나의 영역에만 속할 수 있습니다.
- 대상 영역: 데이터 트래픽 흐름을 종료하는 VPN 그룹. VPN은 하나의 영역에만 속할 수 있습니다.
- 영역 간: 서로 다른 영역 간의 트래픽 폴로우가 발생할 때 interzone이라고 합니다(기본적으로 통신이 거부됨).
- 영역 내: 트래픽 흐름이 동일한 영역을 통과할 때 intrazone이라고 합니다(기본적으로 통신이 허용됨).
- Selfzone: 라우터 자체에서 발생하거나 라우터 자체로 전달되는 트래픽을 제어하는 데 사용됩니다(기본적으로 통신이 허용된 기본 영역은 시스템에 의해 생성 및 미리 구성됨).



영역 기반 방화벽 다이어그램

ZBFW에서 사용되는 또 다른 개념은 영역 쌍으로, 소스 영역과 목적지 영역을 연결하는 컨테이너입니다. 영역 쌍은 두 영역 사이를 이동하는 트래픽에 방화벽 정책을 적용합니다.

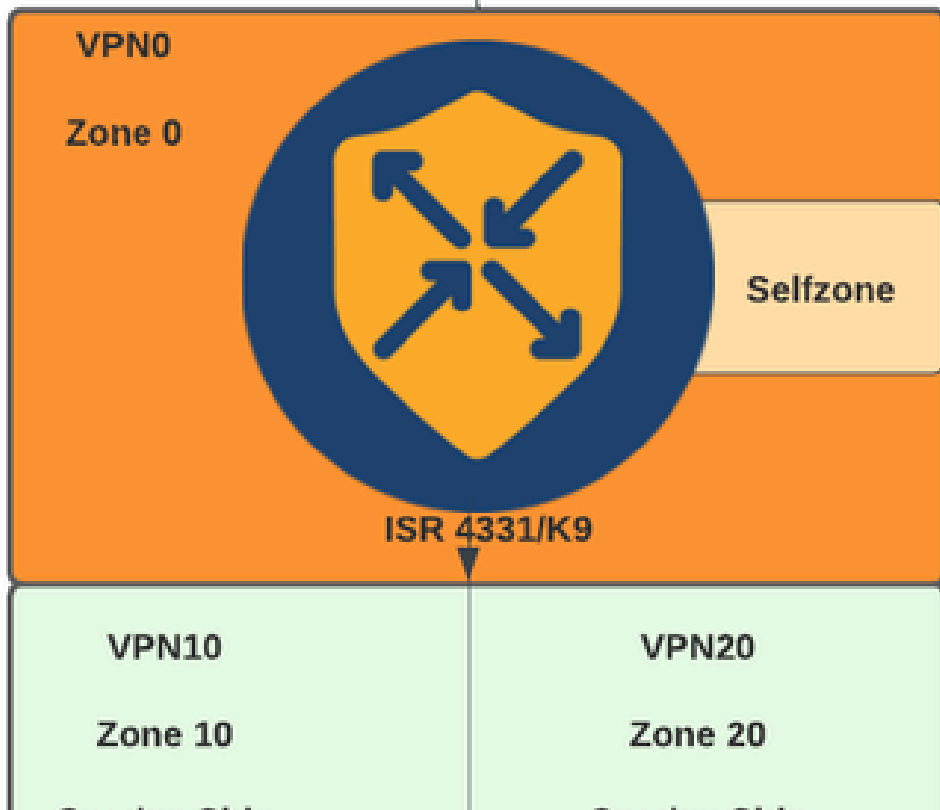
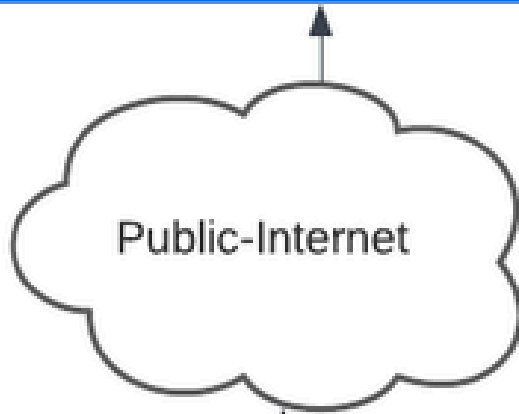
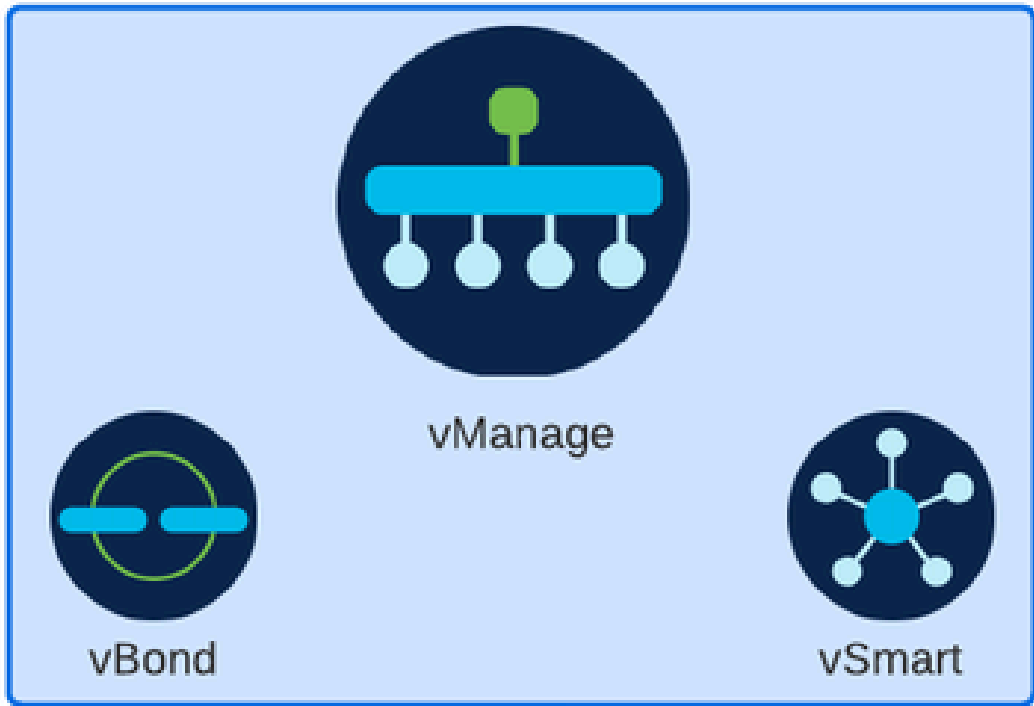


영역 쌍이 정의되면 플로우에 적용되는 작업은 다음과 같습니다.


- 삭제: 일치 흐름을 무시합니다.
- 통과: 액세스 목록의 허용 작업과 마찬가지로 상태 저장 검사 없이 패킷 흐름을 허용합니다. 흐름에 통과 작업이 설정되었는지 여부는 해당 흐름에 대한 반환 전달이 필요합니다.
- 검사: 소스에서 대상 영역으로 이동하는 트래픽에 대한 상태 기반 검사를 허용하고, 트래픽 흐름이 반환되도록 자동으로 허용합니다.

구성

네트워크 다이어그램



는 데이터 플레인에 적용되도록 설계되었습니다. 그러나 컨트롤 플레인이 구성되어 있고 디바이스가 SD-WAN 컨트롤러와 통신하면서 제어 연결을 구축할 수 있는지 확인해야 합니다.

 참고: WAN 인터페이스가 DHCP를 통해 구성되는지 여부와 상관없이, 디바이스 및 라우터를 다시 로드하기 위해 새 IP 주소를 가져와야 하는 경우 셀프 영역(인터페이스)이 next-hop IP 주소에 도달할 수 있도록 규칙을 만들어야 합니다.

컨트롤 플레인

1. inspect 매개변수 맵을 생성합니다.

```
parameter-map type inspect-global
multi-tenancy
vpn zone security
  alert on
  log dropped-packets
max-incomplete tcp timeout
```


`configuration max-incomplete tcp`

명령은 TCP 세션이 삭제되기 전에 불완전한 연결의 최대 수를 지정하는 데 사용됩니다.

`configuration multi-tenancy` 명령은 ZBFW 컨피그레이션에 필요한 전역 매개변수입니다. SD-WAN Manager GUI를 통해 ZBFW를 구성하면 라인이 기본적으로 추가됩니다. CLI(Command Line Interface)를 통해 ZBFW를 구성할 경우 이 라인을 추가해야 합니다.

2. WAN 영역을 생성합니다.

```
zone security wan
vpn 0
```

 참고: 자체 영역은 기본적으로 생성되므로 구성할 필요가 없습니다.

3. 소스 및 대상 주소에 대한 개체 그룹을 구성합니다.

```
object-group network CONTROLLERS
host 172.18.121.103
host 172.18.121.106
host 192.168.20.152
host 192.168.22.203
object-group network WAN_IPs
host 10.122.163.207
```

4. IP 액세스 목록을 생성합니다.

```
ip access-list extended self-to-wan-acl
 10 permit tcp object-group WAN_IPs object-group CONTROLLERS
 20 permit udp object-group WAN_IPs object-group CONTROLLERS
 30 permit ip object-group WAN_IPs object-group CONTROLLERS
ip access-list extended wan-to-self-acl
 10 permit tcp object-group CONTROLLERS object-group WAN_IPs
 20 permit udp object-group CONTROLLERS object-group WAN_IPs
 30 permit ip object-group CONTROLLERS object-group WAN_IPs
```

5. 클래스 맵을 만듭니다.

```
class-map type inspect match-all self-to-wan-cm
 match access-group name self-to-wan-acl
class-map type inspect match-all wan-to-self-cm
 match access-group name wan-to-self-acl
```

6. 영역 쌍에 추가할 정책 맵을 생성합니다.

```
policy-map type inspect wan-to-self-pm
 class type inspect wan-to-self-cm
 inspect
 class class-default
policy-map type inspect self-to-wan-pm
 class type inspect self-to-wan-cm
 inspect
 class class-default
```

7. 영역 쌍을 생성하고 정책 맵을 연결합니다.

```
zone-pair security self-to-wan source self destination wan
 service-policy type inspect self-to-wan-pm
zone-pair security wan-to-self source wan destination self
 service-policy type inspect wan-to-self-pm
```

제어 평면 흐름이 허용되면 데이터 평면 컨피그레이션을 적용할 수 있습니다.

제어 연결을 검증하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show sdwan control connections
```

자체 영역 및 wan 영역에 대한 ZBFW가 올바르게 구성되지 않은 경우 디바이스에서는 제어 연결이 끊기고 다음과 유사한 콘솔 오류가 발생합니다.

<#root>

```
*Oct 30 19:44:17.731: %IOSXE-6-PLATFORM: R0/0: cpp_cp: QFP:0.0 Thread:000 TS:00000004865486441431 %FW-6-
```

데이터 플레인

1. 필요한 각 VRF(Virtual Routing and Forwarding)에 대한 보안 영역을 생성합니다.

```
zone security user
vpn 10
zone security server
vpn 20
```

3. 소스 및 대상 주소에 대한 개체 그룹을 구성합니다.

```
object-group network USER
host 10.10.10.1
host 10.10.10.2
host 10.10.10.3
object-group network SERVER
host 10.20.20.1
host 10.20.20.2
```

4. IP 액세스 목록을 생성합니다.

```
ip access-list extended user-to-server-acl
10 permit tcp object-group USER object-group SERVER
20 permit udp object-group USER object-group SERVER
30 permit ip object-group USER object-group SERVER
ip access-list extended server-to-user-acl
10 permit tcp object-group SERVER object-group USER
20 permit udp object-group SERVER object-group USER
30 permit ip object-group SERVER object-group USER
```

5. 클래스 맵을 만듭니다.


```
class-map type inspect match-all user-to-server-cm
match access-group name user-to-server-ac1
class-map type inspect match-all server-to-wan-cm
match access-group name server-to-user-ac1
```

6. 영역 쌍에 추가할 정책 맵을 생성합니다.

```
policy-map type inspect user-to-server-pm
class type inspect user-to-server-cm
inspect
class class-default
policy-map type inspect server-to-user-pm
class type inspect server-to-user-cm
inspect
class class-default
```

7. 영역 쌍을 생성하고 정책 맵을 연결합니다.

```
zone-pair security user-to-server source user destination server
service-policy type inspect user-to-server-pm
zone-pair security server-to-user source server destination user
service-policy type inspect server-to-user-pm
```

 참고: CLI 템플릿 사용에 대한 자세한 내용은 CLI [애드온 기능 템플릿](#) 및 [CLI 템플릿을 참조하십시오](#).

다음을 확인합니다.

구성된 inspect class-map을 검증하려면 EXEC 명령을 사용합니다.

```
<#root>
```

```
Device#
```

```
show class-map type inspect
```

구성된 inspect policy-map을 검증하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show policy-map type inspect
```

구성된 영역 쌍을 검증하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show zone-pair security
```

구성된 access-list를 검증하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show ip access-list
```

구성된 object-group을 검증하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show object-group
```

ZBFW 세션 상태를 표시하려면 EXEC 명령을 사용합니다.

<#root>

Device#

```
show sdwan zonebfpwdp sessions
```

```
  SRC DST TOTAL TOTAL UTD
SESSION SRC DST SRC DST VPN VPN NAT INTERNAL INITIATOR RESPONDER APPLICATION POLICY
ID STATE SRC IP DST IP PORT PORT PROTOCOL VRF VRF ID ID ZP NAME CLASSMAP NAME FLAGS FLAGS BYTES BYTES T
-----
 8 open 172.18.121.106 10.122.163.207 48960 32168 PROTO_L4_UDP 0 0 0 65534 wan-to-self wan-to-self-cm - 0
 5 open 10.122.163.207 172.18.121.106 32168 32644 PROTO_L4_UDP 0 0 65534 0 self-to-wan self-to-wan-cm - 0
 7 open 10.122.163.207 172.18.121.103 32168 32168 PROTO_L4_UDP 0 0 65534 0 self-to-wan self-to-wan-cm - 0
```

```
6 open 172.18.121.106 10.122.163.207 60896 32168 PROTO_L4_UDP 0 0 0 65534 wan-to-self wan-to-self-cm -
9 open 10.122.163.207 172.18.121.106 32168 34178 PROTO_L4_UDP 0 0 65534 0 self-to-wan self-to-wan-cm -
```

영역 쌍 통계를 표시하려면 EXEC 명령을 사용합니다.

```
<#root>
```

```
Device#
```

```
show sdwan zbfw zonepair-statistics
```

```
zbfw zonepair-statistics user-to-server
src-zone-name user
dst-zone-name server
policy-name user-to-server-pm
fw-traffic-class-entry user-to-server-cm
zonepair-name user-to-server
```

```
class-action Inspect
```

```
pkts-counter 0
bytes-counter 0
attempted-conn 0
```

```
current-active-conn 0
```

```
max-active-conn 0
current-halfopen-conn 0
max-halfopen-conn 0
current-terminating-conn 0
max-terminating-conn 0
```

```
time-since-last-session-create 0
```

ZBFW 삭제 통계를 표시하려면 EXEC 명령을 사용합니다.

```
<#root>
```

```
Device#
```

```
show sdwan zbfw drop-statistics
```

```
zbfw drop-statistics catch-all
```

```
0
```

```

zbfw drop-statistics l4-max-halfsession          0
zbfw drop-statistics l4-session-limit            0
zbfw drop-statistics l4-scb-close                0

zbfw drop-statistics insp-policy-not-present      0

zbfw drop-statistics insp-sess-miss-policy-not-present 0

zbfw drop-statistics insp-classification-fail     0
zbfw drop-statistics insp-class-action-drop      0
zbfw drop-statistics insp-policy-misconfigure    0

zbfw drop-statistics l4-icmp-err-policy-not-present 0

zbfw drop-statistics invalid-zone                0

zbfw drop-statistics ha-ar-standby               0
zbfw drop-statistics no-forwarding-zone          0

zbfw drop-statistics no-zone-pair-present        105 <<< If no zone-pair configured

```

QFP(QuantumFlow Processor) 삭제 통계를 표시하려면 EXEC 명령을 사용합니다.

```
<#root>
```

```
Device#
```

```
show platform hardware qfp active statistic drop
```

```
Last clearing of QFP drops statistics: never
```

```
-----
Global Drop Stats                Packets                Octets
-----
```

BFDoffload	194	14388
FirewallBackpressure	0	0
FirewallInvalidZone	0	0
FirewallL4	1	74
FirewallL4Insp	372	40957
FirewallL7	0	0
FirewallNoForwardingZone	0	0
FirewallNoNewSession	0	0
FirewallNonsession	0	0
FirewallNotFromInit	0	0
FirewallNotInitiator	11898	885244
FirewallPolicy	0	0

QFP 방화벽 삭제를 표시하려면 EXEC 명령을 사용합니다.

<#root>

Device#

show platform hardware qfp active feature firewall drop all

Drop Reason	Packets
TCP out of window	0
TCP window overflow	0
<snipped>	
TCP - Half-open session limit exceed	0
Too many packet per flow	0
<snipped>	
ICMP ERR PKT:no IP or ICMP	0
ICMP ERR Pkt:exceed burst lmt	0
ICMP Unreach pkt exceeds lmt	0
ICMP Error Pkt invalid sequence	0
ICMP Error Pkt invalid ACK	0
ICMP Error Pkt too short	0
Exceed session limit	0
Packet rcvd in SCB cclose state	0
Pkt rcvd after CX req teardown	0
CXSC not running	0

Zone-pair without policy

0 <<< Existing zone-pair, but not

Same zone without Policy

0 <<< Zone without policy configu

<snipped>

No Zone-pair found

105 <<< If no zone-pair configured

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.