# C8300 Series에서 FQDN ACL 패턴 일치를 사용하여 ZBFW 구성

## 목차



## 소개

이 문서에서는 C8300 플랫폼의 자동 모드에서 FQDN ACL 패턴 일치를 사용하여 ZBFW를 구성하는 절차에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

Cisco에서는 다음 항목에 대해 알고 있는 것이 좋습니다.

- ZBFW(Zone-Based Policy Firewall)

- VRF(Virtual Routing and Forwarding)
- NAT(Network Address Translation)

### 사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- C8300-2N2S-6T 17.12.02

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

ZBFW(Zone-Based Policy Firewall)는 Cisco IOS® 및 Cisco IOS XE 디바이스에서 네트워크 내에 보안 영역을 생성할 수 있는 고급 방화벽 컨피그레이션 방법입니다.

ZBFW를 사용하면 관리자가 인터페이스를 영역으로 그룹화하고 이러한 영역 사이를 이동하는 트래픽에 방화벽 정책을 적용할 수 있습니다.

Cisco 라우터의 ZBFW와 함께 사용되는 FQDN ACL(Fully Qualified Domain Name Access Control Lists)을 사용하면 관리자가 IP 주소만 아니라 도메인 이름을 기반으로 트래픽과 일치하는 방화벽 규칙을 생성할 수 있습니다.

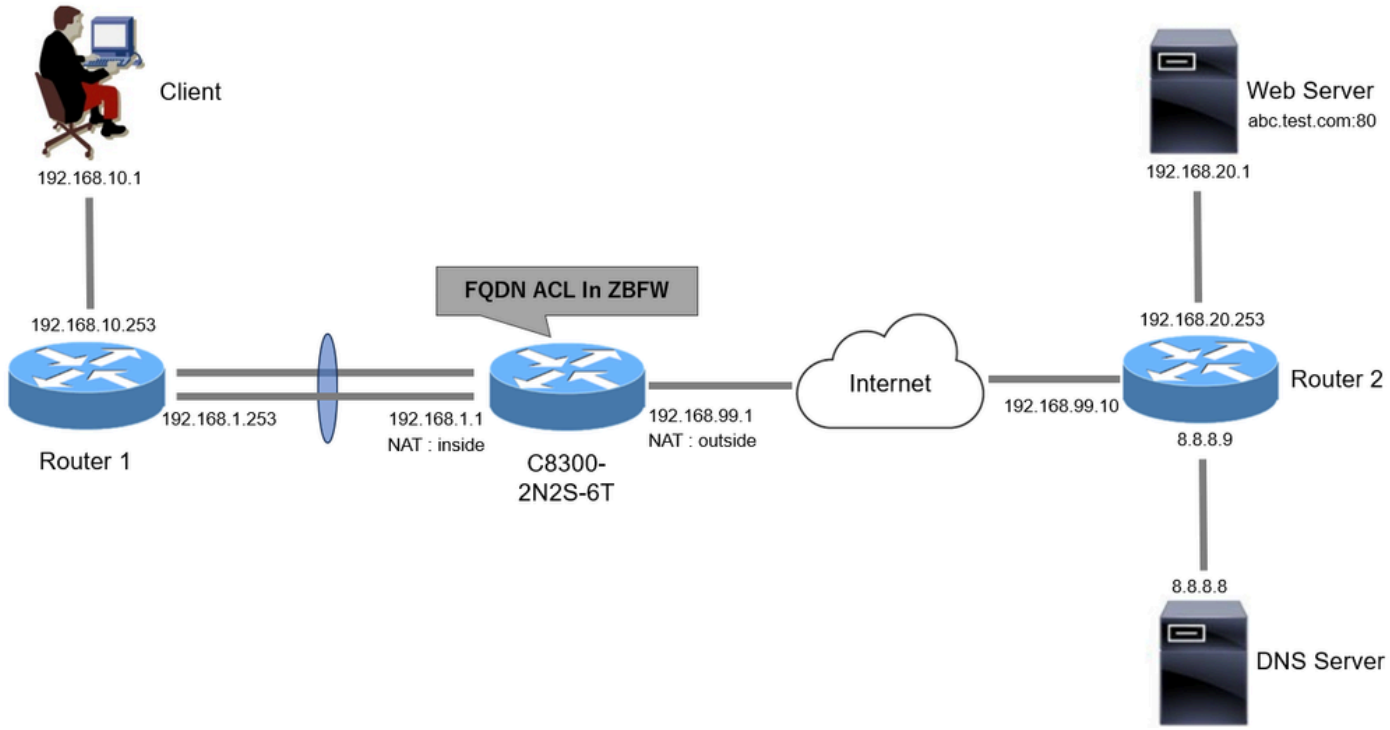
이 기능은 서비스와 연결된 IP 주소가 자주 변경될 수 있는 AWS 또는 Azure와 같은 플랫폼에서 호스팅된 서비스를 처리할 때 특히 유용합니다.

액세스 제어 정책의 관리를 간소화하고 네트워크 내에서 보안 구성의 유연성을 향상합니다.

## 구성

### 네트워크 다이어그램

이 문서에서는 이 다이어그램을 기반으로 ZBFW의 컨피그레이션 및 확인을 소개합니다
. BlackJumboDog를 DNS 서버로 사용하는 시뮬레이션된 환경입니다.




네트워크 다이어그램

## 설정

클라이언트에서 웹 서버로의 통신을 허용하는 컨피그레이션입니다.

1단계. (선택 사항) VRF 구성

VRF(Virtual Routing and Forwarding) 기능을 사용하면 단일 라우터 내에서 여러 독립적인 라우팅 테이블을 생성하고 관리할 수 있습니다. 이 예에서는 WebVRF라는 VRF를 생성하고 관련 통신을 위한 라우팅을 수행합니다.

```
vrf definition WebVRF
rd 65010:10
!
address-family ipv4
route-target export 65010:10
route-target import 65010:10
exit-address-family
!
address-family ipv6
route-target export 65010:10
route-target import 65010:10
exit-address-family

ip route vrf WebVRF 8.8.8.8 255.255.255.255 GigabitEthernet0/0/3 192.168.99.10
ip route vrf WebVRF 192.168.10.0 255.255.255.0 Port-channel1.2001 192.168.1.253
ip route vrf WebVRF 192.168.20.0 255.255.255.0 GigabitEthernet0/0/3 192.168.99.10
```

2단계. 인터페이스 구성

내부 및 외부 인터페이스에 대한 영역 멤버, VRF, NAT 및 IP 주소와 같은 기본 정보를 구성합니다.

```
interface GigabitEthernet0/0/1
no ip address
negotiation auto
lacp rate fast
channel-group 1 mode active

interface GigabitEthernet0/0/2
no ip address
negotiation auto
lacp rate fast
channel-group 1 mode active

interface Port-channel1
no ip address
no negotiation auto

interface Port-channel1.2001
encapsulation dot1Q 2001
vrf forwarding WebVRF
ip address 192.168.1.1 255.255.255.0
ip broadcast-address 192.168.1.255
no ip redirects
no ip proxy-arp
ip nat inside
zone-member security zone_client

interface GigabitEthernet0/0/3
vrf forwarding WebVRF
ip address 192.168.99.1 255.255.255.0
ip nat outside
zone-member security zone_internet
speed 1000
no negotiation auto
```

3단계. (선택 사항) NAT 구성

내부 및 외부 인터페이스에 대한 NAT를 구성합니다. 이 예에서는 클라이언트의 소스 IP 주소 (192.168.10.1)가 192.168.99.100으로 변환됩니다.

```
ip access-list standard nat_source
10 permit 192.168.10.0 0.0.0.255

ip nat pool natpool 192.168.99.100 192.168.99.100 prefix-length 24
ip nat inside source list nat_source pool natpool vrf WebVRF overload
```

4단계. FQDN ACL 구성

대상 트래픽과 일치하도록 FQDN ACL을 구성합니다. 이 예에서는 대상 FQDN을 매칭하려면 FQDN 객체 그룹의 패턴 매칭에서 와일드카드 '*'를 사용합니다.

```
object-group network src_net
192.168.10.0 255.255.255.0

object-group fqdn dst_test_fqdn
pattern .*\.test\.com

object-group network dst_dns
host 8.8.8.8

ip access-list extended Client-WebServer
1 permit ip object-group src_net object-group dst_dns
5 permit ip object-group src_net fqdn-group dst_test_fqdn
```

5단계. ZBFW 구성

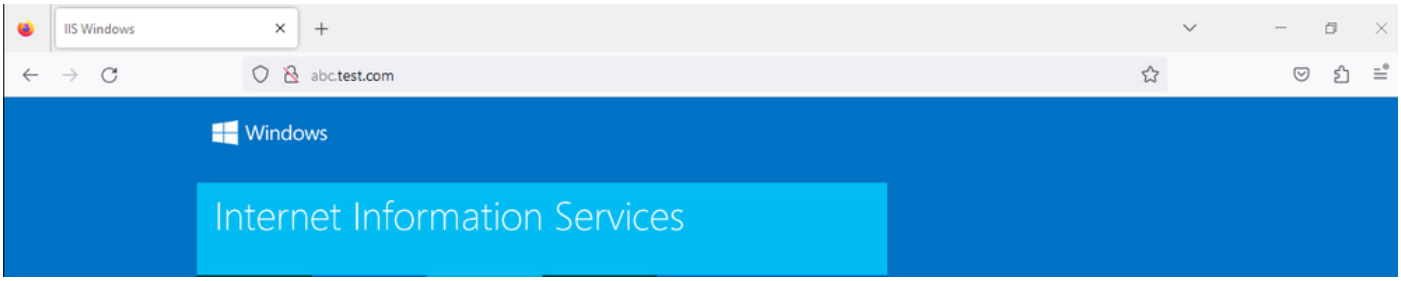
ZBFW에 대한 영역, 클래스 맵, 정책 맵을 구성합니다. 이 예에서는 parameter-map을 사용하여 ZBFW에서 트래픽을 허용할 때 로그가 생성됩니다.

```
zone security zone_client
zone security zone_internet

parameter-map type inspect inspect_log
audit-trail on

class-map type inspect match-any Client-WebServer-Class
match access-group name Client-WebServer

policy-map type inspect Client-WebServer-Policy
class type inspect Client-WebServer-Class
inspect inspect_log
class class-default
drop log

zone-pair security Client-WebServer-Pair source zone_client destination zone_internet
service-policy type inspect Client-WebServer-Policy
```

# 다음을 확인합니다.

## 1단계. 클라이언트에서 HTTP 연결 시작

클라이언트에서 웹 서버로의 HTTP 통신이 성공했는지 확인합니다.




HTTP 연결

## 2단계. IP 캐시 확인

명령을 실행하여show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all 대상 FQDN의 IP 캐시가 C8300-2N2S-6T에서 생성되는지 확인합니다.

### <#root>

02A7382#

**show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all**

```
IP Address Client(s) Expire RegexId Dirty VRF ID Match
-------------------------------------------------------------------------------------------------------------
192.168.20.1 0x1 117 0xdbccd400 0x00 0x0 .*\.test\.com
```

3단계. ZBFW 로그 확인

IP 주소(192.168.20.1)가 FQDN(.*\.test\.com)과 일치하는지 확인하고 1단계의 HTTP 통신이 ZBFW에서 허용되는지 확인합니다.

*Mar 7 11:08:23.018: %IOSXE-6-PLATFORM: R0/0: cpp_cp: QFP:0.0 Thread:003 TS:00000551336606461468 %FW-6-SESS_AUDIT_TRAIL_START

*Mar 7 11:08:24.566: %IOSXE-6-PLATFORM: R0/0: cpp_cp: QFP:0.0 Thread:002 TS:00000551338150591101 %FW-6-SESS_AUDIT_TRAIL: (target:

4단계. 패킷 캡처 확인

대상 FQDN에 대한 DNS 확인 및 클라이언트와 웹 서버 간의 HTTP 연결이 성공적인지 확인합니다.

내부 패킷 캡처:

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.Seq | Next sequence | TCP.Ack | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 15 | 2024-03-07 11:50:36.775945 | 0x0511 (1297) | 192.168.10.1 | 64078 | 8.8.8.8 | 53 | 127 | DNS | 76 | | | | Standard query 0xa505 A abc.test.com |
| 18 | 2024-03-07 11:50:36.782949 | 0xe036 (57398) | 8.8.8.8 | 53 | 192.168.10.1 | 64078 | 126 | DNS | 92 | | | | Standard query response 0xa505 A abc.test.com A 192.168.20.1 |

내부의 *DNS* 패킷

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.Seq | Next sequence | TCP.Ack | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 22 | 2024-03-07 11:50:36.798954 | 0x4575 (17781) | 192.168.10.1 | 51715 | 192.168.20.1 | 80 | 127 | TCP | 70 | 0 | 1 | 0 | 51715 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM |
| 23 | 2024-03-07 11:50:36.798954 | 0x92fb (37627) | 192.168.20.1 | 80 | 192.168.10.1 | 51715 | 126 | TCP | 70 | 0 | 1 | 1 | 80 → 51715 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 WS=256 |
| 24 | 2024-03-07 11:50:36.798954 | 0x4576 (17782) | 192.168.10.1 | 51715 | 192.168.20.1 | 80 | 127 | TCP | 58 | 1 | 1 | 1 | 51715 → 80 [ACK] Seq=1 Ack=1 Win=2102272 Len=0 |
| 26 | 2024-03-07 11:50:36.803944 | 0x4577 (17783) | 192.168.10.1 | 51715 | 192.168.20.1 | 80 | 127 | HTTP | 492 | 1 | 435 | 1 | GET / HTTP/1.1 |
| 27 | 2024-03-07 11:50:36.806949 | 0x92fc (37628) | 192.168.20.1 | 80 | 192.168.10.1 | 51715 | 126 | HTTP | 979 | 1 | 922 | 435 | HTTP/1.1 200 OK (text/html) |

내부의 *HTTP* 패킷

온사이드의 패킷 캡처(192.168.10.1은 192.168.19.100에 대한 NAT):

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.Seq | Next sequence | TCP.Ack | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | 2024-03-07 11:50:36.775945 | 0x0511 (1297) | 192.168.99.100 | 64078 | 8.8.8.8 | 53 | 126 | DNS | 72 | | | | Standard query 0xa505 A abc.test.com |
| 6 | 2024-03-07 11:50:36.782949 | 0xe036 (57398) | 8.8.8.8 | 53 | 192.168.99.100 | 64078 | 127 | DNS | 88 | | | | Standard query response 0xa505 A abc.test.com A 192.168.20.1 |

외부 *DNS* 패킷

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.Seq | Next sequence | TCP.Ack | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 2024-03-07 11:50:36.798954 | 0x4575 (17781) | 192.168.99.100 | 51715 | 192.168.20.1 | 80 | 126 | TCP | 66 | 0 | 1 | 0 | 51715 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK |
| 11 | 2024-03-07 11:50:36.798954 | 0x92fb (37627) | 192.168.20.1 | 80 | 192.168.99.100 | 51715 | 127 | TCP | 66 | 0 | 1 | 1 | 80 → 51715 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460 |
| 12 | 2024-03-07 11:50:36.798954 | 0x4576 (17782) | 192.168.99.100 | 51715 | 192.168.20.1 | 80 | 126 | TCP | 54 | 1 | 1 | 1 | 51715 → 80 [ACK] Seq=1 Ack=1 Win=2102272 Len=0 |
| 14 | 2024-03-07 11:50:36.803944 | 0x4577 (17783) | 192.168.99.100 | 51715 | 192.168.20.1 | 80 | 126 | HTTP | 488 | 1 | 435 | 1 | GET / HTTP/1.1 |
| 15 | 2024-03-07 11:50:36.806949 | 0x92fc (37628) | 192.168.20.1 | 80 | 192.168.99.100 | 51715 | 127 | HTTP | 975 | 1 | 922 | 435 | HTTP/1.1 200 OK (text/html) |

외부 *HTTP* 패킷

문제 해결

FQDN ACL 패턴 일치를 사용하여 ZBFW와 관련된 통신 문제를 해결하려면 문제 중에 로그를 수집하여 Cisco TAC에 제공할 수 있습니다. 문제 해결을 위한 로그는 문제의 특성에 따라 달라집니다.

수집할 로그의 예:

```
!!!! before reproduction
!! Confirm the IP cache
show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all

!! Enable packet-trace
debug platform packet-trace packet 8192 fia-trace
debug platform packet-trace copy packet both
debug platform condition ipv4 access-list Client-WebServer both
debug platform condition feature fw dataplane submode all level verbose

!! Enable debug-level system logs and ZBFW debug logs
debug platform packet-trace drop
debug acl cca event
debug acl cca error
debug ip domain detail
!! Start to debug
debug platform condition start

!! Enable packet capture on the target interface (both sides) and start the capture
monitor capture CAPIN interface Port-channel1.2001 both
monitor capture CAPIN match ipv4 any any
monitor capture CAPIN buffer size 32
monitor capture CAPIN start

monitor capture CAPOUT interface g0/0/3 both
monitor capture CAPOUT match ipv4 any any
monitor capture CAPOUT buffer size 32
monitor capture CAPOUT start

!! (Optional) Clear the DNS cache on the client
ipconfig/flushdns
ipconfig /displaydns
```

!! Run the show command before reproduction
show platform hardware qfp active feature firewall drop all
show policy-map type inspect zone-pair Client-WebServer-Pair sessions
show platform packet-trace statistics
show platform packet-trace summary
show logging process cpp_cp internal start last boot
show platform hardware qfp active feature dns-snoop-agent client hw-pattern-list
show platform hardware qfp active feature dns-snoop-agent client info
show platform hardware qfp active feature dns-snoop-agent datapath stats
show ip dns-snoop all
show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all
show platform software access-list F0 summary

!!!! Reproduce the issue - start

!! During the reproductionof the issue, run show commands at every 10 seconds
!! Skip show ip dns-snoop all command if it is not supported on the specific router
show ip dns-snoop all
show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all

!!!! After reproduction
!! Stop the debugging logs and packet capture
debug platform condition stop
monitor capture CAPIN stop
monitor capture CAPOUT stop

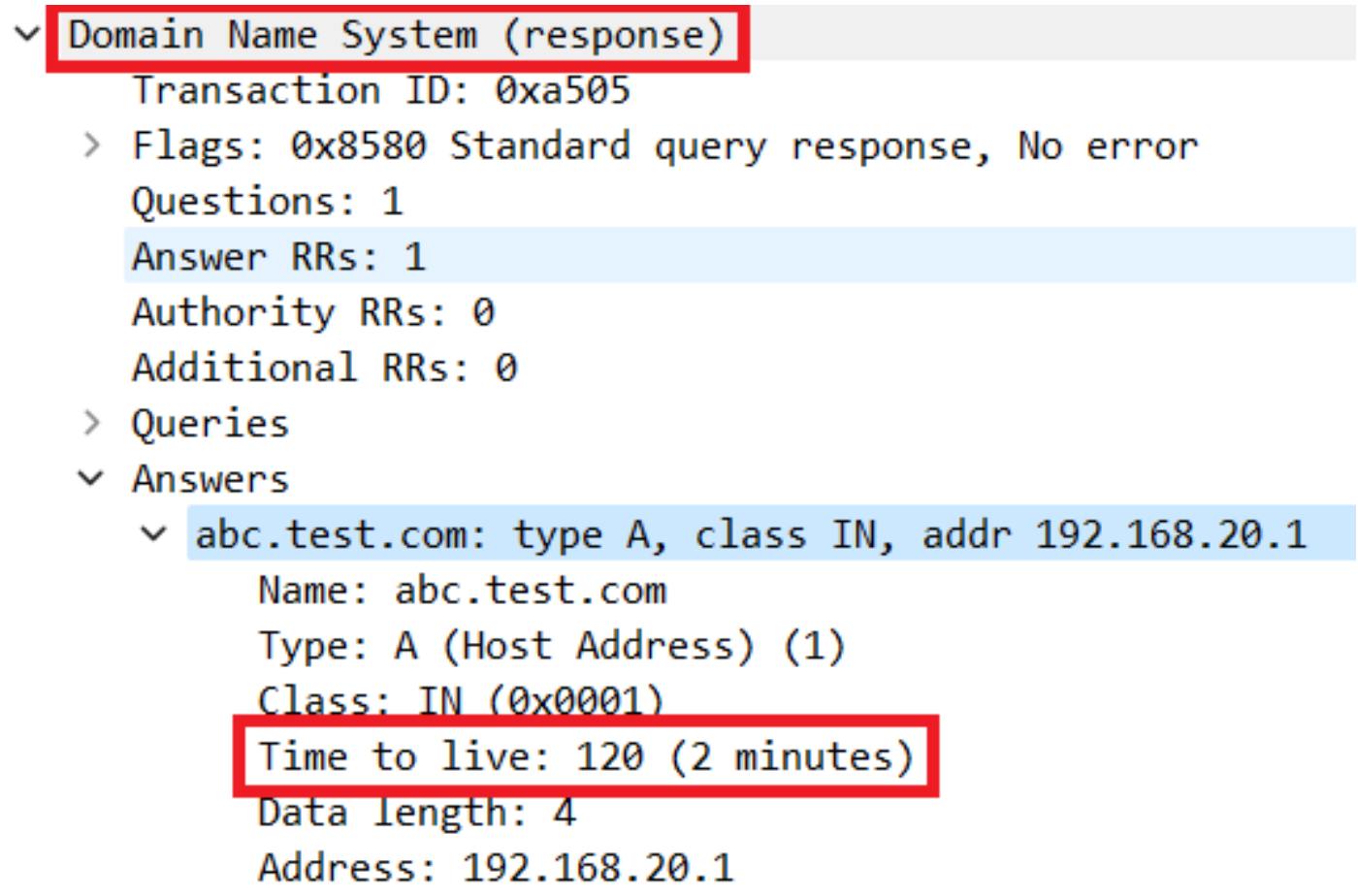
!! Run the show commands
show platform hardware qfp active feature firewall drop all
show policy-map type inspect zone-pair Client-WebServer-Pair sessions
show platform packet-trace statistics
show platform packet-trace summary
show logging process cpp_cp internal start last boot
show platform hardware qfp active feature dns-snoop-agent client hw-pattern-list
show platform hardware qfp active feature dns-snoop-agent client info
show platform hardware qfp active feature dns-snoop-agent datapath stats
show ip dns-snoop all
show platform hardware qfp active feature dns-snoop-agent datapath ip-cache all
show platform software access-list F0 summary

show platform packet-trace packet all decode
show running-config

## 자주 묻는 질문(FAQ)

Q: 라우터에서 IP 캐시의 시간 초과 값은 어떻게 결정됩니까?

A: IP 캐시의 시간 초과 값은 DNS 서버에서 반환된 DNS 패킷의 TTL(Time-To-Live) 값에 따라 결정됩니다. 이 예에서는 120초입니다. IP 캐시가 시간 초과되면 라우터에서 자동으로 제거됩니다. 패킷 캡처의 세부 정보입니다.




*DNS* 확인의 패킷 세부사항

Q: DNS 서버가 A 레코드가 아닌 CNAME 레코드를 반환할 때 허용됩니까?

A: 네, 문제 없어요. CNAME 레코드가 DNS 서버에 의해 반환되면 DNS 확인 및 HTTP 통신이 문제 없이 진행됩니다. 패킷 캡처의 세부 정보입니다.

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.Seq | Next sequence | TCP.Ack | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 350 | 2024-03-07 12:09:55.625959 | 0x0bc5 (3013) | 192.168.10.1 | 63777 | 8.8.8.8 | 53 | 127 | DNS | 76 | | | | Standard query 0x6bd8 A abc.test.com |
| 352 | 2024-03-07 12:09:55.629957 | 0xe4fe (58622) | 8.8.8.8 | 53 | 192.168.10.1 | 63777 | 126 | DNS | 114 | | | | Standard query response 0x6bd8 A abc.test.com CNAME def.test. |

내부의 *DNS* 패킷

```
Domain Name System (response)
    Transaction ID: 0x6bd8
  > Flags: 0x8580 Standard query response, No error
    Questions: 1
    Answer RRs: 2
    Authority RRs: 0
    Additional RRs: 0
  > Queries
  v Answers
    v abc.test.com: type CNAME, class IN, cname def.test.com
        Name: abc.test.com
        Type: CNAME (Canonical NAME for an alias) (5)
        Class: IN (0x0001)
        Time to live: 120 (2 minutes)
        Data length: 6
        CNAME: def.test.com
    v def.test.com: type A, class IN, addr 192.168.20.1
        Name: def.test.com
        Type: A (Host Address) (1)
        Class: IN (0x0001)
        Time to live: 120 (2 minutes)
        Data length: 4
        Address: 192.168.20.1
```

*DNS* 확인의 패킷 세부사항

| No. | Time | Identification | Source | S.Port | Destination | D.Port | Time to Live | Protocol | Length | TCP.S | Next : | TCP.A | Info |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 356 | 2024-03-07 12:09:55.644955 | 0x4589 (17801) | 192.168.10.1 | 51801 | 192.168.20.1 | 80 | 127 | TCP | 70 | 0 | 1 | 0 | 51801 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=2 |
| 357 | 2024-03-07 12:09:55.644955 | 0x9349 (37705) | 192.168.20.1 | 80 | 192.168.10.1 | 51801 | 126 | TCP | 70 | 0 | 1 | 1 | 80 → 51801 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MS |
| 358 | 2024-03-07 12:09:55.644955 | 0x458a (17802) | 192.168.10.1 | 51801 | 192.168.20.1 | 80 | 127 | TCP | 58 | 1 | 1 | 1 | 51801 → 80 [ACK] Seq=1 Ack=1 Win=2102272 Len=0 |
| 359 | 2024-03-07 12:09:55.645962 | 0x458b (17803) | 192.168.10.1 | 51801 | 192.168.20.1 | 80 | 127 | HTTP | 492 | 1 | 435 | 1 | GET / HTTP/1.1 |
| 362 | 2024-03-07 12:09:55.646954 | 0x934a (37706) | 192.168.20.1 | 80 | 192.168.10.1 | 51801 | 126 | HTTP | 979 | 1 | 922 | 435 | HTTP/1.1 200 OK (text/html) |

내부의 *HTTP* 패킷

Q: C8300 라우터에서 수집한 패킷 캡처를 FTP 서버로 전송하는 명령은 무엇입니까?

A: monitor capture <capture name> export bootflash:<capture name>.pcap 및 copy bootflash:<capture name>.pcap ftp://<user>:<password>@<FTP IP Address> 명령을 사용하여 패킷 캡처를 FTP 서버로 전송합니다. CAPIN을 FTP 서버로 전송하는 예입니다.

<#root>

**monitor capture CAPIN export bootflash:CAPIN.pcap**

**copy bootflash:CAPIN.pcap ftp://<user>:<password>@<FTP IP Address>**

참조

영역 기반 정책 방화벽 설계 이해

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.