

ACI OSPF 인접성 문제 해결

목차

[소개](#)

[토폴로지](#)

[OSPF 피어링 컨피그레이션 요건](#)

[OSPF 인접성 문제 해결 - 일반 체크리스트](#)

[OSPF 인접성 문제 해결 - 결함](#)

[노드의 컨트롤 플레인 트래픽 캡처](#)

[Wireshark 확인](#)

[문제 해결 시나리오](#)

[OSPF 인접성 문제 해결: 영역 ID 불일치](#)

[OSPF 인접성 문제 해결:영역 유형 불일치](#)

[OSPF 인접성 문제 해결:중복 라우터 ID](#)

[OSPF 인접성 문제 해결:MTU 불일치](#)

[OSPF 인접성 문제 해결:인증 불일치](#)

[OSPF 인접성 문제 해결: Hello/Dead 타이머 불일치](#)

[OSPF 인접성 문제 해결:인터페이스 유형 불일치](#)

[브로드캐스트](#)

[포인트 투 포인트](#)

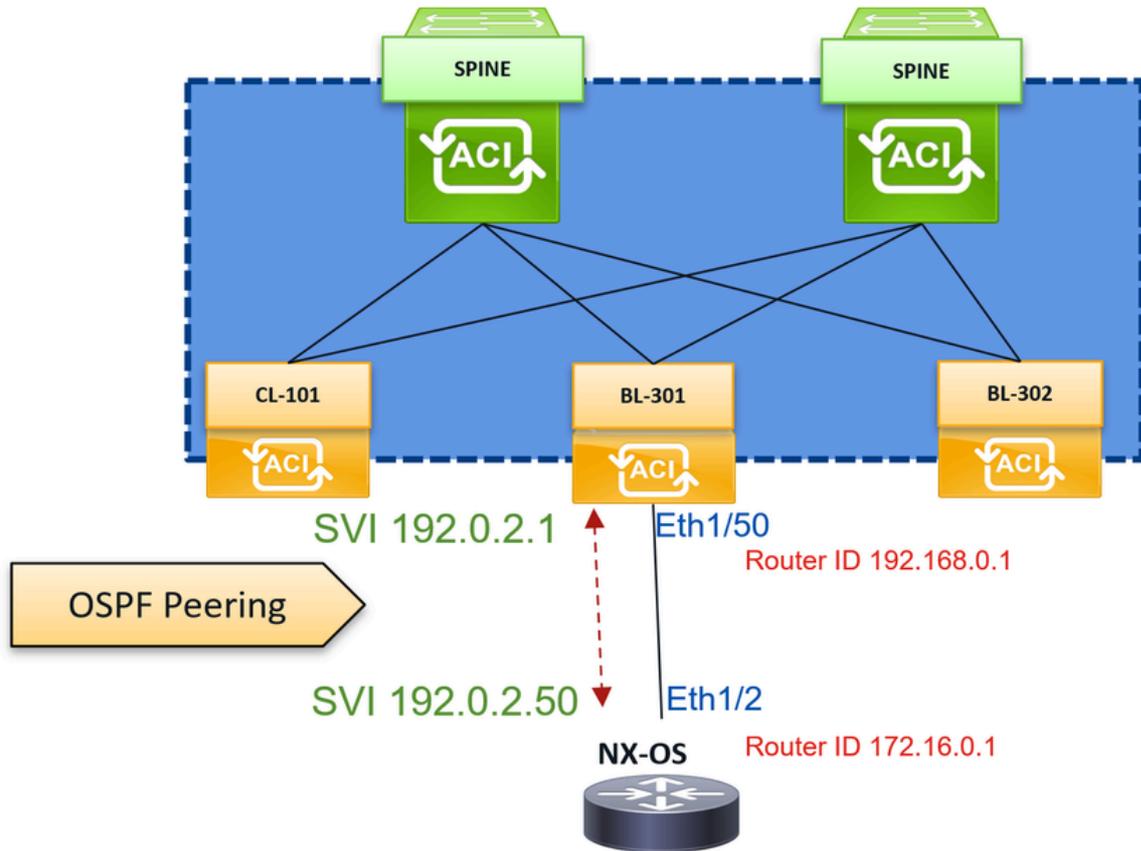
[확인 명령 검사표](#)

[관련 정보](#)

소개

이 문서에서는 ACI(Application Centric Infrastructure) OSPF(Open Shortest Path First) 인접성 문제 해결에 대해 설명합니다.

토폴로지



토폴로지

OSPF 피어링 컨피그레이션 요건

OSPF는 Cisco ACI와 외부 라우터 간에 활성화할 수 있는 프로토콜 중 하나입니다. Cisco ACI는 백본을 비롯한 OSPF 영역, 다양한 스텝 옵션, 네이버 인증 및 기타 유사한 옵션과 같은 모든 일반 옵션을 지원합니다.

L3Out에는 라우팅 프로토콜 옵션, 스위치별 컨피그레이션(노드 프로파일) 및 인터페이스별 설정(인터페이스 프로파일)이 포함됩니다. OSPF 관련 매개변수는 일반 라우터와 마찬가지로 주로 두 곳에서 구성할 수 있습니다. 첫 번째는 L3Out 자체에서 구성할 수 있는 영역 ID 및 영역 유형과 같은 VRF(Virtual Routing and Forwarding) 또는 노드 전체 컨피그레이션입니다. 두 번째는 OSPF hello 간격 또는 인터페이스 유형(Broadcast, Point-to-Point(P2P))과 같은 인터페이스 레벨 매개변수입니다.

ACI 보더 리프와 외부 라우터 간에 OSPF 인접성을 설정하기 위한 요구 사항은 다음과 같습니다.

1. OSPF 영역 ID 및 유형이 일치해야 합니다.
2. OSPF 라우터 ID는 고유해야 합니다.
3. MTU(최대 전송 단위)는 일치해야 합니다(기본적으로 패브릭은 9000으로 설정하고 대부분의 Cisco IOS®/NXOS는 1500으로 설정).
4. OSPF 인증 키와 유형이 일치해야 합니다(사용되는 경우).
5. OSPF Hello 간격과 Dead 간격이 일치해야 함
6. OSPF 네트워크 유형이 일치해야 합니다.

이 [백서](#)는 지원 라우팅 프로토콜에 대한 ACI L3Out과 관련된 설계 개념과 옵션에 대한 자세한 설명을 제공했습니다.

L3Out 설정 및 기타 기본적인 요구 사항에 익숙하지 않은 경우 백서를 참조하십시오.

OSPF 인접성 문제 해결 - 일반 체크리스트

OSPF 인접성이 이전에 설정되었는지 또는 설정된 적이 없는지에 관계없이 기본 요구 사항을 먼저 검증하는 것이 가장 좋습니다.

1단계. 원격 엔드 인터페이스를 ping합니다. 그러면 OSPF가 가동되기 위한 기본 요건인 원단에 대한 IP 연결 기능이 있는지 확인하는 데 도움이 됩니다.

```
iping -V <vrf> <remote_end_IP>
example:
BL-301# iping -V abc1:vrf-1 192.0.2.50
```

2단계. 기본 컨피그레이션 매개변수를 확인합니다.

1. OSPF 영역 ID 및 유형이 일치해야 합니다.
2. OSPF 라우터 ID는 고유해야 합니다.
3. MTU는 일치해야 합니다(기본적으로 패브릭은 9000으로 설정하고 대부분의 Cisco IOS/NXOS는 1500으로 설정).
4. OSPF 인증 키와 유형이 일치해야 합니다(사용되는 경우).
5. OSPF Hello 간격과 Dead 간격이 일치해야 함
6. OSPF 네트워크 유형이 일치해야 합니다.

명령 출력에는 leaf에 푸시된 컨피그레이션 특성이 표시됩니다.

<#root>

```
BL-301# show ip int bri vrf abc1:vrf-1
IP Interface Status for VRF "abc1:vrf-1"(137)
Interface          Address                Interface Status
vlan1              192.0.2.1/24          protocol-up/link-up/admin-up --> l3out SVI
lo9                192.168.0.1/32        protocol-up/link-up/admin-up --> Router ID SVI
```

BL-301#

```
show ip ospf interface vlan 1
```

```
Vlan1 is up, line protocol is up
IP address
```

```
192.0.2.1/24
```

```
, Process ID default VRF
```

```
abc1:vrf-1
```

```
,
area backbone

Enabled by interface configuration
State P2P,

Network type P2P

, cost 4
Index 84, Transmit delay 1 sec
1 Neighbors, flooding to 1, adjacent with 1
Timer intervals:

Hello 10, Dead 40

, Wait 40, Retransmit 5
Hello timer due in 00:00:03

No authentication

Number of opaque link LSAs: 0, checksum sum 0

BL-301#
show int vlan
1 | egrep "MTU"
```

```
MTU
9000
bytes, BW 10000000 Kbit, DLY 1 usec

BL-301#
show ip ospf vrf abc1:vrf-1 | grep Routing

Routing Process default with ID
192.168.0.1
VRF abc1:vrf-1 -->

Router ID
```

강조 표시된 모든 세부 정보를 기록하고 해당 원격 최종 매개변수가 동기화되었는지 확인합니다.

OSPF 인접성 문제 해결 - 결함

<#root>

[+]From the border Leaf we can identify the state of the neighbor state
BL-301# show ip ospf neighbors vrf

```
abc1:vrf-
```

```
1
```

```
<<EMPTY>>
```

[+] You can check the associated faults to the VRF.

```
BL-301# moquery -c faultInst -x 'query-target-filter=wcard(faultInst.dn,"
```

```
abc1:vrf-1
```

```
")' | egrep "code|rule|dn|descr|lastTransition"
```

```
<<EMPTY>>
```

환경에 활성 장애가 없는 몇 가지 시나리오가 있지만 leaf에 하나의 장애 레코드 F1385(protocol-ospf-adjacency-down)가 있을 수 있습니다. 이 leaf는 이 네이버가 마지막으로 가동되었을 때 또는 풀 상태가 아닐 때 이를 알려 줍니다.

명령을 사용하여 이를 식별할 수 `moquery -c faultRecord -f 'fault.Inst.code=="F1385"' -x 'query-target-filter=wcard(faultRecord.dn,"abc1:vrf-1")' | grep dn` 있습니다.

명령을 사용하여 특정 날짜에 대한 결함 레코드의 수를 `moquery -c faultRecord -f 'fault.Inst.code=="F1385"' -x 'query-target-filter=wcard(faultRecord.dn,"abc1:vrf-1") -x 'query-target-filter=wcard(faultRecord.created,"2024-01-01")' | egrep "dn" | wc -l` 확인합니다.

OSPF 인터페이스와 로컬 및 원격 구성 IP를 식별해야 합니다.

```
<#root>
```

[+] Identify the IP applied on the external device from the ARP associated to the interface

```
BL-301# moquery -c arpAdjEp -x 'query-target-filter=wcard(arpAdjEp.ifId,"
```

```
vlan1
```

```
")' | grep "ip "
```

```
ip          :
```

```
192.0.2.50
```

노드의 컨트롤 플레인 트래픽 캡처

보더 리프의 예상 소스 및 대상 스위치 SVI(가상 인터페이스)를 사용하여 tcpdump 유틸리티를 사용하여 확인할 수 있습니다.

참고: 이 경우 모든 CPU 인밴드 컨트롤 플레인 네트워크 트래픽을 볼 수 있는 인터페이스 kpm_inb가 사용됩니다.

<#root>

[+] Capture a single OSPF hello packet using TCPDUMP coming for local BL OSPF IP 192.0.2.1

BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb

tcpdump: listening on kpm_inb, link-type EN10MB (Ethernet), capture size 262144 bytes

192.0.2.1 > ospf-all.mcast.net:

OSPFv2

, Hello, length 44

Router-ID 192.168.0.1,

Backbone Area

,
Authentication Type: none (0)

Options [

External

]

Hello Timer 10s, Dead Timer 40s,

Mask 255.255.255.0, Priority 1

[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP 192.0.2.50
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb
tcpdump: listening on kpm_inb, link-type EN10MB (Ethernet), capture size 262144 bytes
192.0.2.50 > ospf-all.mcast.net:

OSPFv2

, Hello, length 44
Router-ID 172.16.0.1,

Backbone Area, Authentication Type: none (0)

Options [

External

]

Hello Timer 10s, Dead Timer 40s,

Mask 255.255.255.0, Priority 1

Wireshark 확인

Wireshark에서 분석하기 위해 OSPF 및 호스트별 트래픽을 캡처할 수 있습니다.

```
BL-301# tcpdump -i kpm_inb proto ospf -vv -e -w - | tee /data/techsupport/Node-XXX_OSPF.pcap | tcpdump -r - host any
```

```
BL-301# tcpdump -xxxvi kpm_inb 'proto ospf and (host <<X.X.X.X>> or host <<Y.Y.Y.Y>>)' -w /data/techsupport/Node-XXX_OSPF_HOST.pcap
```

```
BL-301# tcpdump -i kpm_inb proto ospf -vv -e -w - | tee /data/techsupport/Node-XXX_OSPF_HOST.pcap | tcpdump -r - host X.X.X.X
```

pcap 캡처의 경우 Analyze(분석) > Apply as a Column(열로 적용)을 검색하고 사용하여 Wireshark 필터를 사용할 수 있습니다.

ospf.area_id = AreaID를 식별하기 위해

ospf.auth.type = 구성된 인증 유형이 일치하는지 확인하기 위해

ospf.hello.hello_interval = 다른 MTU를 확인하기 위해

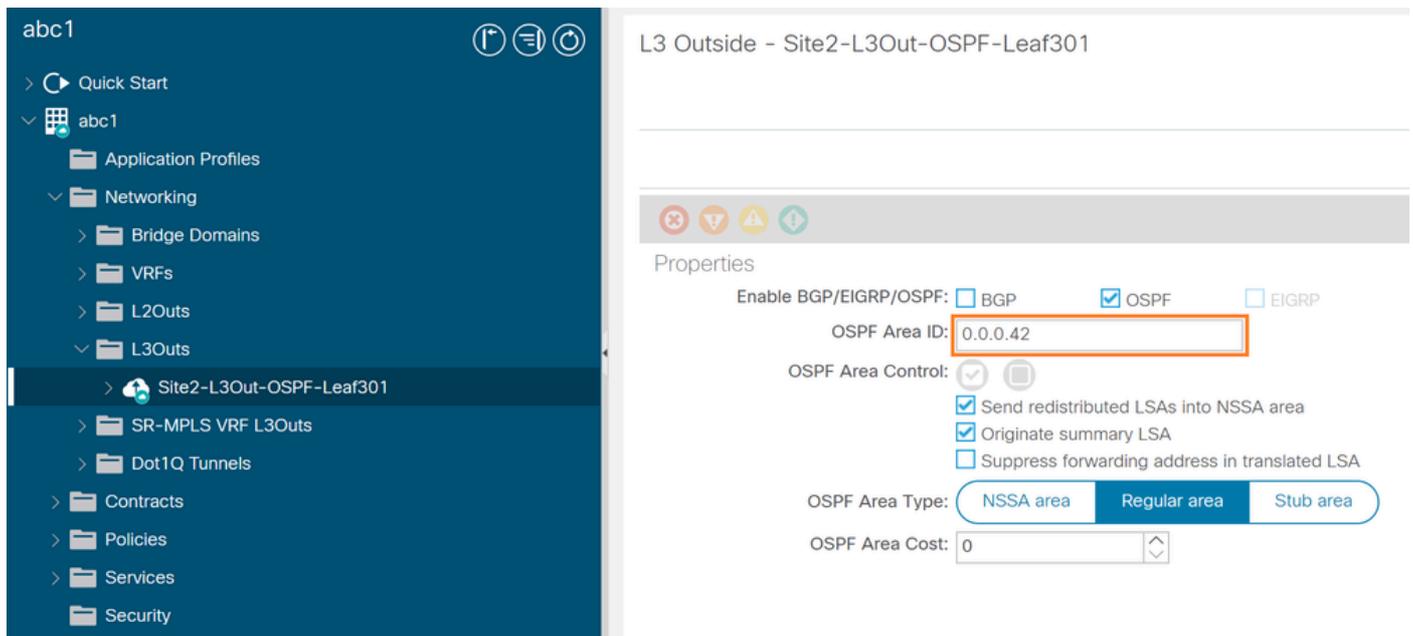
ospf.hello.router_dead_interval = 다른 dead 간격 컨피그레이션을 확인하기 위한 경우

ospf.srcrouter = 라우터 ID

문제 해결 시나리오

OSPF 인접성 문제 해결: 영역 ID 불일치

Area ID가 0.0.0.42인 APIC 컨피그레이션에서 Fabric(패브릭) > Tenants(테넌트) > Networking(네트워킹) > L3Outs(L3Outs) > <<L3outName>> Policy(정책) > Main(기본)으로 이동합니다.



잘못된 OSPF 영역 ID가 0.0.0.42로 구성되었습니다.

테두리 나뭇잎에서:

<#root>

[+] Check OSPF interface details to confirm current area

```
BL-301# show ip ospf interface vlan 1 | grep area
IP address 192.0.2.1/24, Process ID default VRF abc1:vrf-1, area
```

0.0.0.42

Or

```
BL-301# moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.id,"vlan1")' | grep area
area          :
0.0.0.42
```

```
[+] Capture a single packet TCPDUMP for local BL OSPF IP
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 192.168.0.1,
```

Area

0.0.0.42

```
, Authentication Type: none (0)
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

```
[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1
192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 172.16.0.1,
```

Backbone Area

```
, Authentication Type: none (0)
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

외부 장치에서:

<#root>

```
NX-OS# show logging log | tail -n 100 | grep ospf-bootcamp
2023 Dec 28 15:17:09 NX-OS %OSPF-4-AREA_ERR: ospf-bootcamp [22263] (301-l3-abc1) Packet from 192.0.2.1 on Ethernet1/2 received for wrong area
```

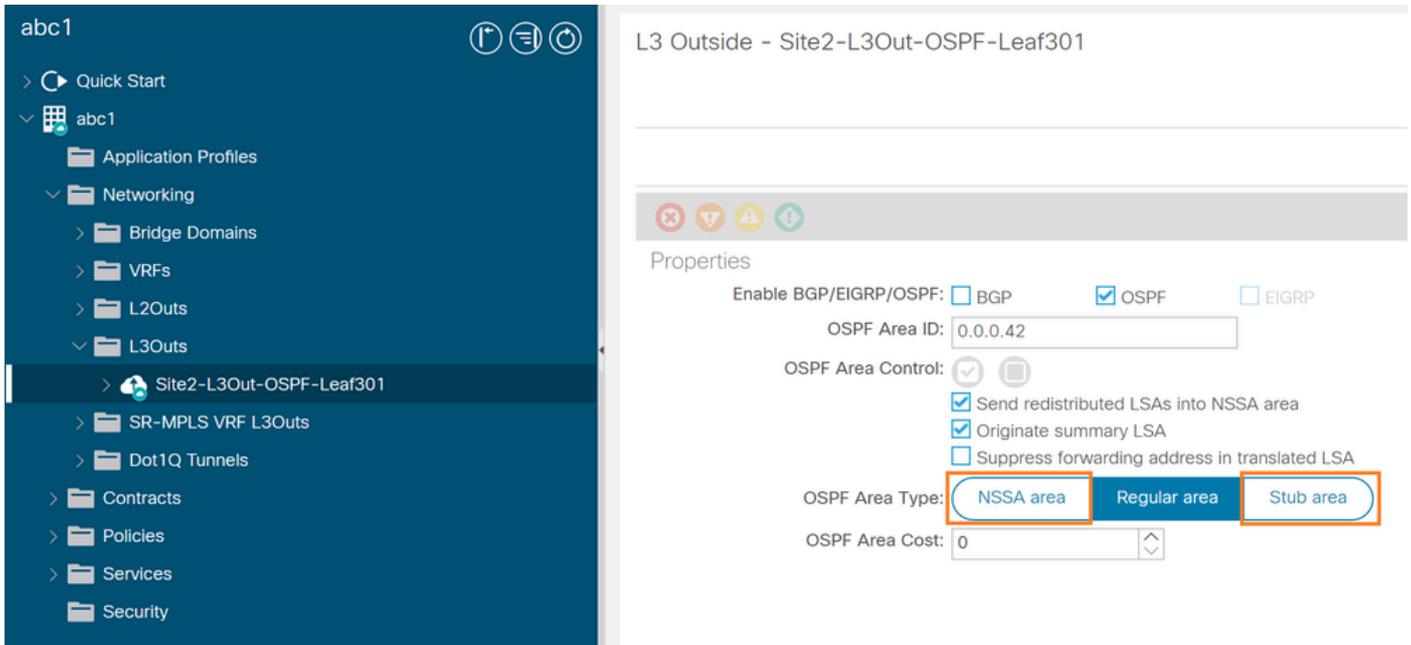
0.0.0.42

```
NX-OS# show ip ospf interface Ethernet1/2 | grep area
Process ID bootcamp VRF 301-l3-abc1,
area 0.0.0.0
```

해결 방법: OSPF Area를 0.0.0.0 또는 백본을 BL에, 또는 0.0.0.42를 외부 디바이스에 일치시킵니다.

OSPF 인접성 문제 해결: 영역 유형 불일치

Area 유형이 NSSA 또는 Stub인 ACI GUI의 컨피그레이션에서 Fabric(패브릭) > Tenants(테넌트) > Networking(네트워킹) > L3Outs(L3Outs) > "L3outName" > Policy(정책) > Main(기본)으로 이동합니다.



NSSA 또는 스텝 영역 컨피그레이션

테두리 나뉘었에서:

<#root>

```
[+] Capture a single packet TCPDUMP for local BL OSPF IP
BL-301# moquery -c ospfArea -x 'query-target-filter=wcard(ospfArea.dn,"abc1:vrf-1")' | egrep "type"
type
:
```

nssa

```
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 192.168.0.1, Area 0.0.0.42, Authentication Type: none (0)
Options [
```

NSSA

```
]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

or

```
BL-301# moquery -c ospfArea -x 'query-target-filter=wcard(ospfArea.dn,"abc1:vrf-1")' | egrep "type"
type
:
```

stub

```
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 192.168.0.1, Area 0.0.0.42, Authentication Type: none (0)
Options [
```

none

```
]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP

```
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1
192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 172.16.0.1, Area 0.0.0.42, Authentication Type: none (0)
Options [
```

External

```
]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

외부 장치에서:

<#root>

[+] Check OSPF interfaces con vrf

```

NX-OS# show ip int bri vrf 301-13-abc1
IP Interface Status for VRF "301-13-abc1"(21)
Interface          IP Address      Interface Status
Lo1001             110.1.0.1      protocol-up/link-up/admin-up
Eth1/2.1120        192.0.2.50     protocol-up/link-up/admin-up

```

```

NX-OS# show ip ospf interface Ethernet1/2 | grep area
Process ID bootcamp VRF 301-13-abc1,

```

area 0.0.0.0

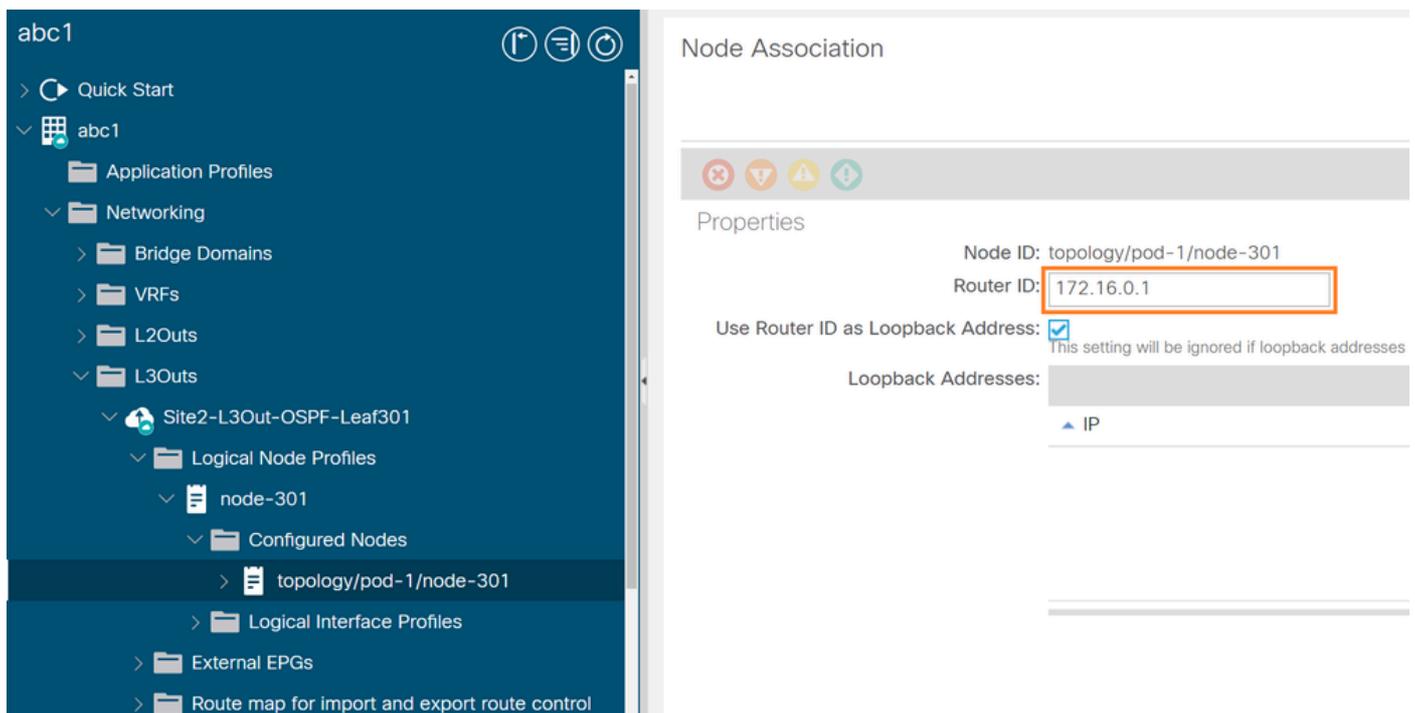
해결 방법: L3Out에서 OSPF 영역 유형을 정기적으로 확인하거나 외부 디바이스에서 일치시킵니다.

OSPF 인접성 문제 해결: 라우터 ID 중복

라우터 ID가 중복되면 OSPF 인접성이 형성되지 않습니다. ACI 패브릭에서 OSPF 라우터 ID를 구성한 후 leaf는 라우터 ID IP 주소로 루프백을 생성합니다. 이 주소는 루프백에 사용되므로 실패하는 경우 사용되는 인터페이스 IP와 중복될 수 없습니다.

이 예에서는 인접 디바이스의 라우터 ID로 잘못 구성되었음을 확인할 수 있습니다.

ACI GUI에서 로 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Configured Nodes > topology/pod-Y/node-X 이동합니다.



인접 디바이스의 라우터 ID로 잘못 구성되었습니다.

테두리 나뉘기에서:

<#root>

[+] Check OSPF interfaces associated with the VRF

BL-301# show ip int bri vrf abc1:vrf-1

IP Interface Status for VRF "abc1:vrf-1"(137)

Interface	Address	Interface Status
vlan1	192.0.2.1/24	protocol-up/link-up/admin-up
lo9		

172.16.0.1

/32 protocol-up/link-up/admin-up

[+] Capture a single packet TCPDUMP for local BL OSPF IP

BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1

192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44

Router-ID

172.16.0.1

, Backbone Area, Authentication Type: none (0)

Options [External]

Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1

[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP

BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1

192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 48

Router-ID

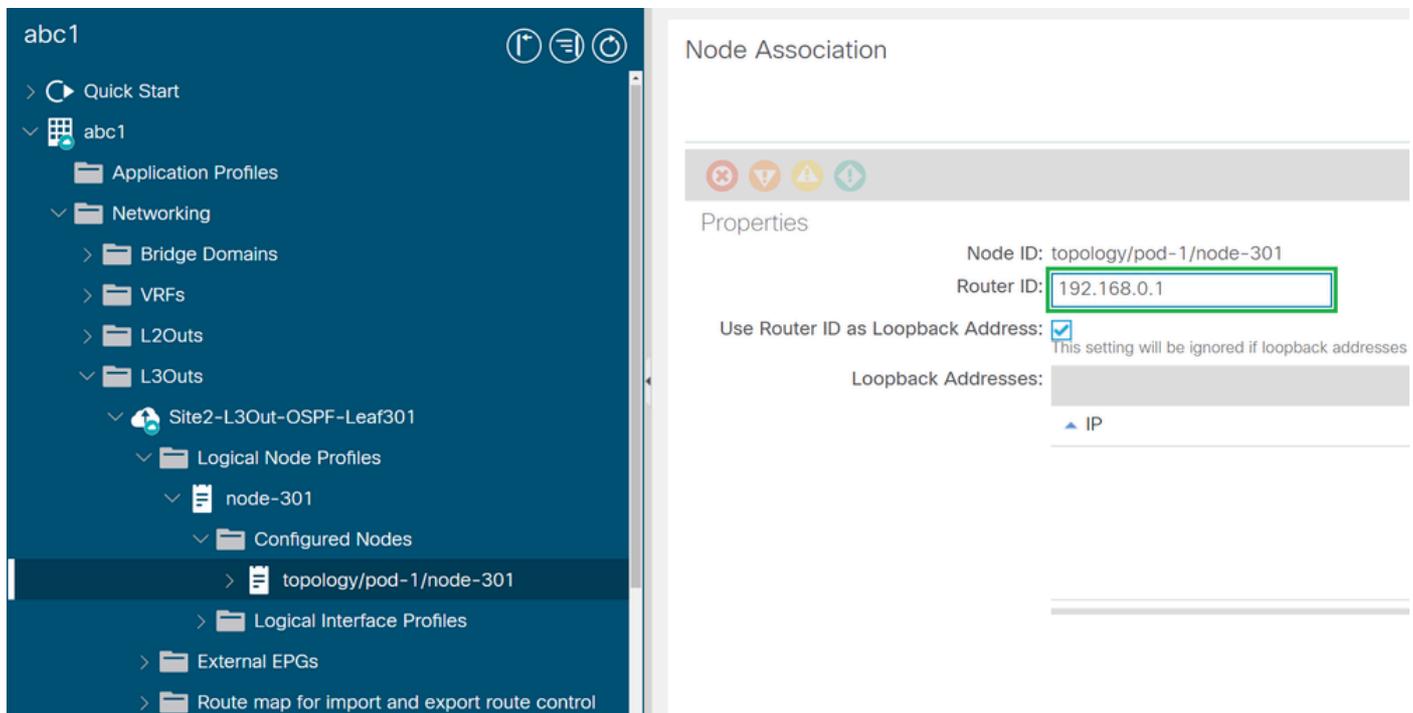
172.16.0.1

, Backbone Area, Authentication Type: none (0)
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1

외부 장치에서

```
NX-OS# show logging log | tail -n 100 | grep ospf-bootcamp  
2024 Jan 4 13:55:36 NX-OS %OSPF-4-DUPRID: ospf-bootcamp [22263] (301-13-abc1) Router 192.0.2.1 on int
```

해결 방법: 두 디바이스에서 서로 다른 라우터 ID를 사용합니다.



두 디바이스 모두에서 서로 다른 라우터 ID 사용

OSPF 인접성 문제 해결: MTU 불일치

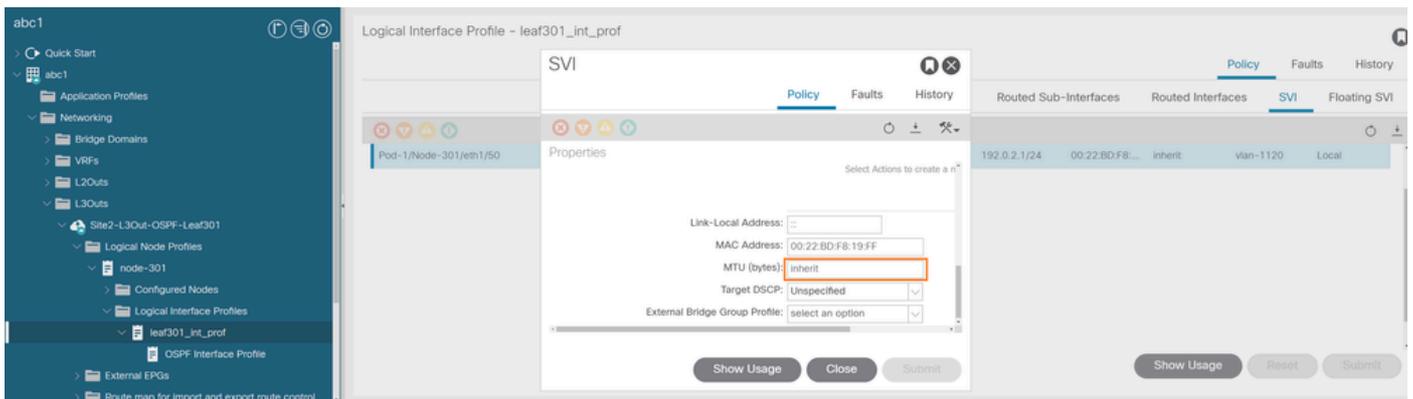
두 개의 OSPF 인접 라우터가 양방향 통신을 설정하고 브로드캐스트 네트워크에서 DR/BDR 선택을 완료하면 라우터가 Exstart 상태로

전환됩니다. 이 상태에서 인접 라우터는 활성/대기 관계를 설정하고 DBD 패킷을 교환하는 동안 사용할 초기 DBD(Database Descriptor) 시퀀스 번호를 결정합니다.

활성/대기 관계가 협상되면(라우터 ID가 가장 높은 라우터가 활성 상태가 됨) 인접 라우터가 교환 상태로 전환됩니다. 이 상태에서 라우터는 전체 링크 상태 데이터베이스를 설명하는 DBD 패킷을 교환합니다. 라우터는 또한 링크 상태 요청 패킷을 전송하며, 인접 디바이스에서 보다 최근의 LSA(링크 상태 알림)를 요청합니다.

인접 라우터 인터페이스에 대한 MTU 설정이 일치하지 않으면 라우터는 Exstart/Exchange 상태에 머물러 있습니다. MTU가 높은 라우터는 인접 라우터에 설정된 MTU보다 큰 패킷을 전송하므로, 인접 라우터는 패킷을 무시합니다.

기본 상속 컨피그레이션이 있는 APIC GUI 컨피그레이션에서 로 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile 이동합니다.



기본적으로 ACI 패브릭은 레이어 3 인터페이스 MTU를 1500이 아닌 9000으로 설정합니다

기본적으로 ACI 패브릭은 레이어 3 인터페이스 MTU를 1500이 아닌 9000으로 설정합니다. ACI는 MTU가 더 높기 때문에 외부 라우터에서 DBD 패킷을 계속 수락하고 이를 승인하려고 시도합니다.

외부 라우터의 MTU가 더 낮거나 더 높은 경우 ACI의 ACK와 함께 DBD 패킷을 무시하고 초기 DBD 패킷을 계속 재전송하며 Exstart/Exchange 상태를 유지합니다.

테두리 나뭇잎에서:

<#root>

[+]From the border Leaf we can identify the state of the neighborhood relation

```
BL-301# show ip ospf neighbors vrf abc1:vrf-1
```

```
OSPF Process ID default VRF abc1:vrf-1
```

```
Total number of neighbors: 1
```

```
Neighbor ID  Pri State      Up Time  Address      Interface
172.16.0.1   1
```

EXCHANGE

```
/ -          01:10:05 192.0.2.50
```

Vlan1

[+] You can check the associated faults to the Tenant:VRF / OSPF interface
BL-301# moquery -c faultInst -x 'query-target-filter=wcard(faultInst,dn,"

abc1:vrf-1

\if-\[

vlan1

\])' | egrep "code|rule|dn|descr|lastTransition"

code :

F1385

descr :

OSPF adjacency is not full, current state Exchange

dn : topology/pod-1/node-301/sys/ospf/inst-default/dom-abc1:vrf-1/if-[vlan1]/adj-172.16.0

lastTransition : 2023-12-28T12:26:23.369-05:00

rule : ospf-adj-ep-failed

title : OSPF Adjacency Down

code :

F3592

descr :

OSPF interface vlan1 mtu is different than neighbor mtu

dn : topology/pod-1/node-301/sys/ospf/inst-default/dom-abc1:vrf-1/if-[vlan1]/fault-F3592

lastTransition : 2023-12-28T12:26:23.369-05:00

rule : ospf-if-mtu-config-mismatch-err

[+] Identify the MTU applied on the OSPF interface

BL-301# show int vlan 1 | egrep "MTU"

MTU

9000

bytes, BW 10000000 Kbit, DLY 1 usec

[+] If the default configuration is on place there will be a missmatch with the 1500 default

BL-301# show ip ospf event-history adjacency | grep "neighbor mtu"

2023-12-28T12:24:31.986149000-05:00 ospf default [20751]: TID 21885:ospfv2_check_ddesc_for_nbr_state:49

neighbor mtu [

1500] is smaller than if mtu 9000

[+] Or if the locally configured MTU is lower than external router

[2023-12-28T14:05:48.495659000-05:00:T:ospfv2_check_ddesc_for_nbr_state:478] abc1:vrf-1DBD from 192.0.2

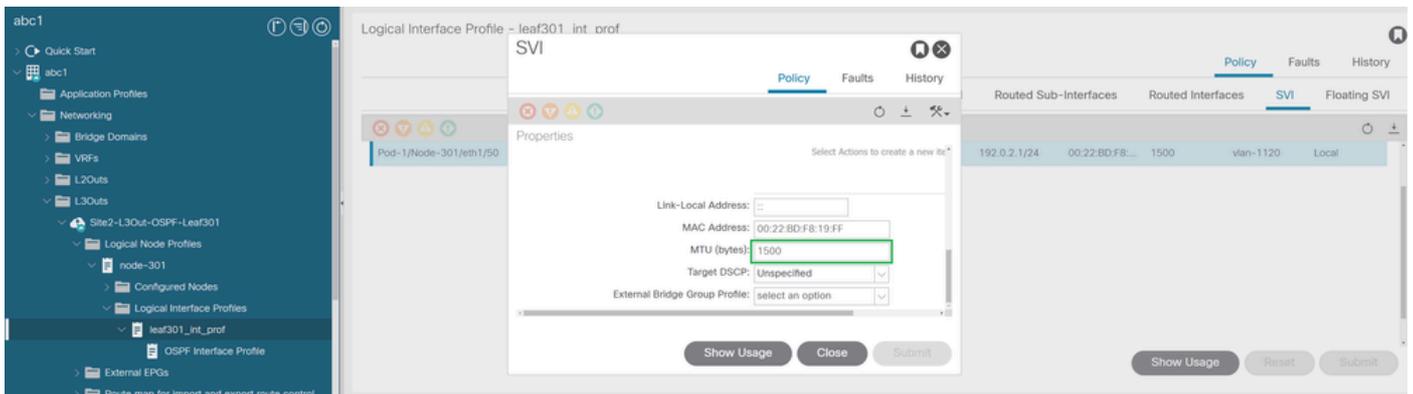
mtu [1500] is large than if mtu 1200

가능한 솔루션:

- 두 디바이스의 MTU 일치

MTU가 어느 한쪽에서 변경되어도 멤버십이 이미 설정되어 있으므로 다음 협상까지 그대로 유지되며 여러 가지 이유로 트리거될 수 있습니다. 예를 들어 물리적 인터페이스 중단, 정책 재구축, 리프 다시 로드, 업그레이드 등이 있습니다.

이미지에 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile 표시된 대로 이동합니다.



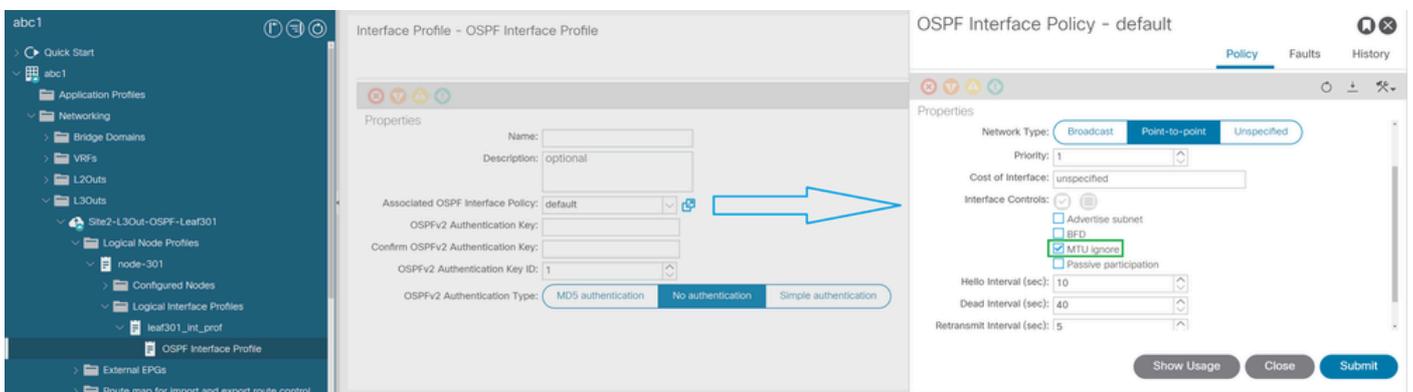
MTU를 1500으로 구성

- 연결된 OSPF 인터페이스 정책에서 MTU가 무시되면 연결이 재설정됩니다.

MTU 무시 문제는 OSPF 데이터베이스가 증가할 때 나타날 수 있습니다. MTU가 몇 바이트만 다른 경우, DBD를 생성하거나 올바른 크기의 패킷을 업데이트하는 올바른 LSA 조합을 발견하기 전까지 설정이 오랫동안 작동할 수 있습니다.

소규모 랩에서의 테스트는 정상적으로 작동하지만 프로덕션 네트워크에서 예기치 않은 동작이 발생할 수 있습니다.

이미지에 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile > Associated OSPF Interface Policy 표시된 대로 이동합니다.



MTU 무시 컨피그레이션

OSPF 인접성 문제 해결: 인증 불일치

라우팅 업데이트 정보를 안전하게 교환하기 위해 OSPF에서 인증을 활성화할 수 있습니다. OSPF 인증은 none(또는 null), simple 또는

MD5일 수 있습니다. 인증 방법 'none'은 OSPF에 대해 어떤 인증도 사용되지 않으며 기본 방법입니다. 간단한 인증으로 비밀번호는 네트워크를 통해 일반 텍스트로 전달됩니다. MD5 인증에서는 암호가 네트워크를 통과하지 않습니다.

OSPF에서 지원하는 세 가지 인증 유형입니다.

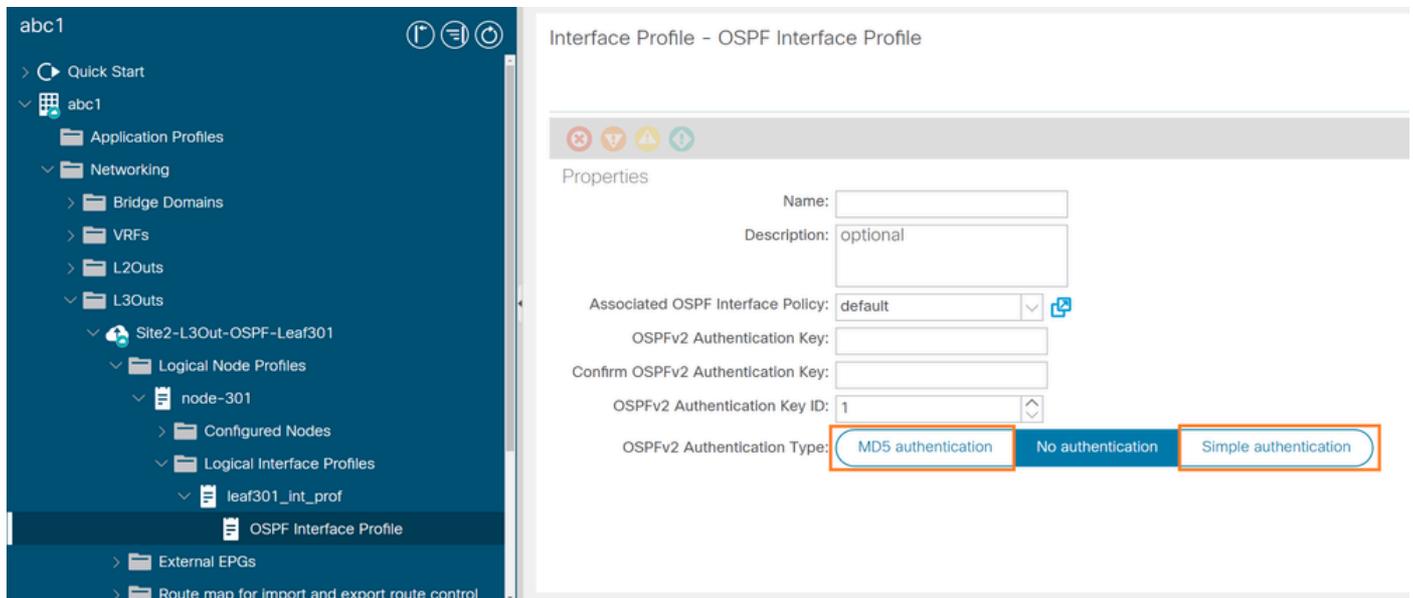
Null 인증 — 이는 유형 0이라고도 하며 패킷 헤더에 인증 정보가 포함되어 있지 않음을 의미합니다. 기본값입니다.

단순 인증 — 유형 1이라고도 하며 간단한 일반 텍스트 비밀번호를 사용합니다.

MD5 인증 — 유형 2라고도 하며 MD5 암호화 비밀번호를 사용합니다.

인증을 설정할 필요가 없습니다. 그러나 이 설정이 적용되면 동일한 세그먼트의 모든 피어 라우터에는 동일한 비밀번호와 인증 방법이 있어야 합니다.

ACI GUI에서 그림과 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile 같이 로 이동합니다.



MD5 또는 단순 인증이 구성됨

CLI에서:

```
<#root>
```

```
[+] Check Authentication type configured
```

```
APIC# moquery -c ospfIfP -x 'query-target-filter=wcard(ospfIfP.dn,"tn-abc1\out-Site2-L3Out-OSPF-BL-301
authType      :
```

```
simple
```

```
[+] Capture a single packet TCPDUMP for local BL OSPF IP
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 192.168.0.1, Backbone Area, Authentication Type:
```

```
simple (
```

```
1)
```

```
Simple text password: cisco
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

```
or
```

```
[+] Check Authentication type configured
```

```
APIC# moquery -c ospfIfP -x 'query-target-filter=wcard(ospfIfP.dn,"tn-abc1/out-Site2-L3Out-OSPF-BL-301")'
authType      :
```

```
md5
```

```
[+] Capture a single packet TCPDUMP for local BL OSPF IP
```

```
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
```

```
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 192.168.0.1, Backbone Area, Authentication Type:
```

MD5 (2)

```
Key-ID: 1, Auth-Length: 16, Crypto Sequence Number: 0x026c0a34
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

```
[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1
192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 48
Router-ID 172.16.0.1, Backbone Area, Authentication Type:
```

none (0)

```
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1
```

```
[+] Live OSPF trace Decode for VRF
BL-301# log_trace_bl_print_tool /var/sysmgr/tmp_logs/ospfv2_1_trace.bl | tail -n 250 | grep abc1:vrf-1
[2024-01-04T16:23:29.650806000-05:00:T:ospfv2_set_authentication:70] abc1:vrf-1out pkt on Vlan1:
```

```
auth simple text: key cisco
```

or

```
[2024-01-04T16:24:22.794682000-05:00:T:ospfv2_set_authentication:96] abc1:vrf-1out pkt on Vlan1:
```

auth md5: key cisco

, key id 1 Seq 40635829 (time 1704403462)

외부 장치에서:

```
NX-OS# show logging log | tail -n 100 | grep ospf-bootcamp
2024 Jan  4 16:55:01 NX-OS %OSPF-4-AUTH_ERR: ospf-bootcamp [22263] (301-13-abc1) Received packet from
or
2024 Jan  4 16:55:20 NX-OS %OSPF-4-AUTH_ERR: ospf-bootcamp [22263] (301-13-abc1) Received packet from
```

해결 방법: 인증과 일치시킵니다.

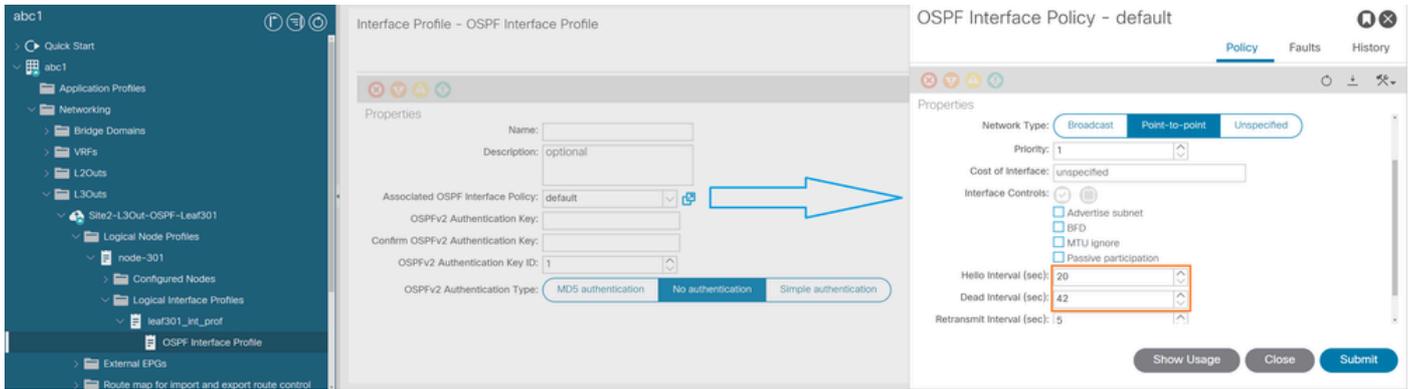
OSPF 인접성 문제 해결: Hello/Dead 타이머 불일치

OSPF hello 패킷은 OSPF 프로세스에서 해당 인접 디바이스와의 연결을 유지하기 위해 OSPF 인접 디바이스로 전송하는 패킷입니다. hello 패킷은 구성 가능한 간격(초)으로 전송됩니다. 이더넷 링크의 기본값은 10초입니다(P2P 및 브로드캐스트 네트워크 유형의 경우). Hello 패킷에는 Dead 간격 내에 수신된 Hello 패킷에 대한 모든 네이버 목록이 포함됩니다. Dead 간격도 구성 가능하며(초) 기본값은 hello 간격 값의 4배입니다. 모든 hello 간격의 값은 네트워크 내에서 동일해야 합니다. 마찬가지로, 모든 dead 간격의 값은 네트워크 내에서 동일해야 합니다.

링크가 작동 중임을 표시하여 연결을 유지하기 위해 이 두 간격이 함께 작동합니다. 라우터가 데드 간격 내에 인접 디바이스로부터 hello 패킷을 수신하지 못하면 해당 인접 디바이스가 다운된 것으로 선언됩니다.

기본 OSPF hello 및 dead 타이머가 ACI 패브릭에서 수정된 경우 외부 라우터와 일치해야 합니다.

ACI GUI에서 그림과 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile > Associated OSPF Interface Policy 같이 로 이동합니다.



사용자 지정 *Hello/Dead* 타이머

테두리 나뉘어:

<#root>

[+] Check OSPF interface configuration
 BL-301# show ip ospf interface vlan 1 | egrep "Timer|Network"
 State P2P, Network type P2P, cost 4
 Timer intervals: Hello

20

, Dead

42

, Wait 42, Retransmit 5

Or

BL-301# moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.id,"vlan1")' | egrep "deadIntvl|helloIntvl"

deadIntvl : 42

helloIntvl : 20

nwT : p2p

Or

```
APIC# moquery -c ospfRsIfPol -x 'query-target-filter=wcard(ospfIfP.dn,"abc1\out-Site2-L3Out-OSPF-BL-30
tnOspfIfPolName : Custom_OSPF_Interface_Policy
```

```
APIC# moquery -c ospfIfPol -x 'query-target-filter=wcard(ospfIfPol.name,"Custom_OSPF_Interface_Policy")
deadIntvl :
```

42

helloIntvl :

2

0

nwT : p2p

```
[+] Capture a single packet TCPDUMP for local BL OSPF IP
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 44
    Router-ID 192.168.0.1, Backbone Area, Authentication Type: none (0)
    Options [External]
```

Hello Timer 20s

,

Dead Timer 42s

, Mask 255.255.255.0, Priority 1

```
[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1
192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 44
    Router-ID 172.16.0.1, Backbone Area, Authentication Type: none (0)
    Options [External]
```

Hello Timer 10s

,

Dead Timer 40s

, Mask 255.255.255.0, Priority 1

외부 장치에서:

<#root>

[+] Check OSPF interfaces con vrf

NX-OS# show ip int bri vrf 301-13-abc1

IP Interface Status for VRF "301-13-abc1"(21)

Interface	IP Address	Interface Status
Lo1001	110.1.0.1	protocol-up/link-up/admin-up
Eth1/2.1120	192.0.2.50	protocol-up/link-up/admin-up

[+] Check OSPF configuration by default Dead timer on NX-OS devices is 4 times hello interval

NX-OS# show run ospf all | section Ethernet1/2.1120 | grep hello

ip ospf hello-interval

10

[+] Check OSPF interface advertized parameters

NX-OS# show ip ospf interface Ethernet1/2.1120 | grep Timer
Timer intervals:

Hello

10

,

Dead

, Wait 40, Retransmit 5

해결 방법: Match OSPF Timers.

OSPF 인접성 문제 해결: 인터페이스 유형 불일치

이 섹션에서는 Broadcast 또는 unspecified가 ACI에 구성되어 있고 외부 디바이스가 P2P인 경우의 트러블슈팅에 대해 설명합니다.

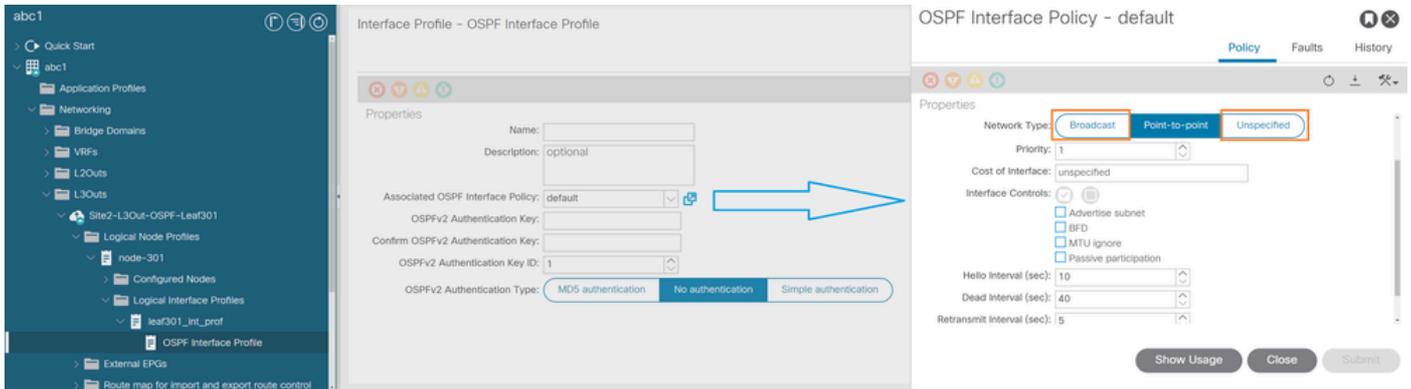
브로드캐스트

- Broadcast 네트워크 유형은 OSPF 지원 이더넷 인터페이스의 기본값입니다
- Broadcast 네트워크 유형에는 링크가 레이어 2 브로드캐스트 기능을 지원해야 합니다
- 브로드캐스트 네트워크 유형에는 10초 hello 및 40초 데드 타이머(P2P와 동일)가 있습니다
- OSPF 브로드캐스트 네트워크 유형에는 DR/BDR을 사용해야 합니다.

포인트 투 포인트

- P2P OSPF 네트워크 유형은 DR/BDR 관계를 유지하지 않습니다
- P2P 네트워크 유형에는 10초 hello 및 40초 데드 타이머가 있습니다
- P2P 네트워크 유형은 직접 연결된 두 라우터 간에 사용하도록 설계되었습니다

ACI GUI에서 그림과 Fabric > Tenants > Networking > L3Outs > "L3outName" > "Node-X" > Logical Interface Profiles > OSPF Interface Profile > Associated OSPF Interface Policy 같이 로 이동합니다.



브로드캐스트 또는 지정되지 않은 네트워크 유형이 구성됨

테두리 나뉠이어서:

<#root>

[+] Check OSPF neighborhood relation

```
BL-301# show ip ospf neighbors vrf abc1:vrf-1
```

```
OSPF Process ID default VRF abc1:vrf-1
```

```
Total number of neighbors: 1
```

Neighbor ID	Pri	State	Up Time	Address	Interface
172.16.0.1	1				

INITIALIZING/DROTHER

```
00:06:42 192.0.2.50 Vlan1
```

[+] Check OSPF interface configuration

```
BL-301# moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.id,"vlan1")' | egrep "deadIntvl|helloIntvl"
```

```
deadIntvl : 40
```

```
helloIntvl : 10
```

```
nwT :
```

bcast

or

```
BL-301# moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.id,"vlan1")' | egrep "deadIntvl|helloIntvl"
```

```
deadIntvl : 40
```

```
helloIntvl    : 10
nwT           :
```

unspecified

Or

```
APIC# moquery -c ospfRsIfPol -x 'query-target-filter=wcard(ospfIfP.dn,"abc1\out-Site2-L3Out-OSPF-BL-30
tnOspfIfPolName : Custom_OSPF_Interface_Policy
```

```
APIC# moquery -c ospfIfPol -x 'query-target-filter=wcard(ospfIfPol.name,"Custom_OSPF_Interface_Policy")
deadIntvl     : 40
helloIntvl    : 10
nwT           :
```

bcast

```
APIC# moquery -c ospfIfPol -x 'query-target-filter=wcard(ospfIfPol.name,"Custom_OSPF_Interface_Policy")
deadIntvl     : 40
helloIntvl    : 10
nwT           :
```

unspecified

[+] Whether it is bcast or unspecified the interface will show as Broadcast
BL-301# show ip ospf interface vlan 1 | egrep "Timer|Network"
State DR, Network type

BROADCAST

, cost 4
Timer intervals: Hello 10, Dead 40, Wait 40, Retransmit 5

[+] Capture a single packet TCPDUMP for local BL OSPF IP
BL-301# tcpdump src host 192.0.2.1 -vv -e -i kpm_inb -c 1
192.0.2.1 > ospf-all.mcast.net: OSPFv2, Hello, length 48
Router-ID 192.168.0.1, Backbone Area, Authentication Type: none (0)
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1

Designated Router

192.0.2.1

Neighbor List:
172.16.0.1

[+] Capture a single OSPF hello packet using TCPDUMP coming from external device OSPF IP
BL-301# tcpdump src host 192.0.2.50 -vv -e -i kpm_inb -c 1
192.0.2.50 > ospf-all.mcast.net: OSPFv2, Hello, length 44
Router-ID 172.16.0.1, Backbone Area, Authentication Type: none (0)
Options [External]
Hello Timer 10s, Dead Timer 40s, Mask 255.255.255.0, Priority 1

외부 장치에서:

<#root>

[+] Check OSPF interfaces con vrf
NX-OS# show ip int bri vrf 301-13-abc1
IP Interface Status for VRF "301-13-abc1"(21)
Interface IP Address Interface Status
Lo1001 110.1.0.1 protocol-up/link-up/admin-up
Eth1/2.1120 192.0.2.50 protocol-up/link-up/admin-up

[+] Check OSPF configuration by default Dead timer on NX-OS devices is 4 times hello interval
NX-OS# show run ospf all | section Ethernet1/2 | grep network
ip ospf network

point-to-point

```
[+] Check OSPF interface advertized parameters
NX-OS# show ip ospf interface Ethernet1/2 | grep type
      State P2P, Network type
```

P2P

, cost 1

확인 명령 검사표

이러한 명령은 다양한 시나리오를 트러블슈팅하기 위해 이 문서 전체에서 참조되었습니다.

노드	명령	목적
ACI 스위치	show ip ospf neighbors vrf <<TNT:VRF>>	VRF에서 인접 관계 확인
	show ip ospf interface brief vrf <<TNT:VRF>>	VRF와 연결된 OSPF 인터페이스 확인
	moquery -c faultInst -x 'query-target-filter=wcard(faultInst.dn,"<<TNT:VRF>>")'	VRF에 연결된 결함을 확인할 수 있습니다
	moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.dn,"<<TNT:VRF>>")'	VRF와 연결된 모든 OSPF 인터페이스 세부사항 확인
	moquery -c ospfIf -x 'query-target-filter=wcard(ospfIf.id,"<<vlanX>>")'	OSPF 인터페이스 컨피그레이션 확인

	<pre>moquery -c arpAdjEp -x 'query-target-filter=wcard(arpAdjEp.ifId,"<<vlanX>>") grep ip</pre>	<p>인터페이스에 연결된 ARP에서 외부 디바이스에 적용된 IP를 확인합니다</p>
	<pre>log_trace_bl_print_tool /var/sysmgr/tmp_logs/ospfv2_1_trace.bl tail -n 250 grep <<TNT:VRF>></pre>	<p>VRF용 라이브 OSPF 추적 디코딩</p>
	<pre>tcpdump -i kpm_inb proto ospf -vv -e -w - tee /data/techsupport/Node- XXX_OSPF.pcap tcpdump -r - host any</pre>	<p>Wireshark에서 분석할 OSPF 트래픽 캡처</p>
	<pre>tcpdump -i kpm_inb proto ospf -vv -e -w - tee /data/techsupport/Node- XXX_OSPF_HOST.pcap tcpdump -r - host <<X.X.X.X>></pre>	<p>Wireshark에서 분석하기 위해 HOST의 특정 트래픽 캡처</p>
	<pre>tcpdump -xxxvi kpm_inb 'proto ospf and (host <<X.X.X.X>> or host <<Y.Y.Y.Y>>)' - w /data/techsupport/Node-XXX_OSPF_HOST.pcap</pre>	<p>Wireshark에서 분석하기 위해 HOST의 SRC 및 DST별 트래픽 캡처</p>
	<pre>tcpdump src host <<X.X.X.X>> -vv -e -i kpm_inb -c 1</pre>	<p>및 특정 호스트에 대한 단일 인밴드 컨트롤 플레인 캡처</p>
ACI APIC	<pre>moquery -c ospfIfP -x 'query-target-filter=wcard(ospfIfP.dn,"<<TNT>>\/out- <<L3outName>>")'</pre>	<p>구성된 인증 유형 확인</p>
	<pre>moquery -c l3ext.RsPathL3OutAtt -x 'query-target- filter=wcard(l3extRsPathL3OutAtt.dn,"<<TNT>>\/out-<<L3outName>>")'</pre>	<p>L3out 경로 컨피그레이션 확인</p>
	<pre>moquery -c faultRecord -f 'fault.Inst.code=="F1385"' -x 'query-target- filter=wcard(faultRecord.dn,"<<TNT:VRF>>")'</pre>	<p>결함 F1385 protocol-ospf-adjacency-down에 대한 결함 기록 레코드 확인</p>
	<pre>moquery -c ospfRsIfPol -x 'query-target-filter=wcard(ospfIfP.dn,"<<TNT>>\/out- <<L3outName>>")' grep tnOspfIfPolName</pre>	<p>L3out에서 사용자 지정 관련 OSPF 인터페이스 정책 확인</p>

	moquery -c ospfIfPol -x 'query-target-filter=wcard(ospfIfPol.name,"Custom_OSPF_Interface_Policy")'	사용자 지정 관련 OSPF 인터페이스 정책 세부 정보 확인
NXOS 스위치	show ip int bri vrf <<VRF>>	OSPF 인터페이스 con vrf 확인
	show run ospf all section EthernetX/Y	OSPF 컨피그레이션 확인
	show ip ospf interface EthernetX/Y	OSPF 인터페이스 알림 매개 변수를 확인합니다.

관련 정보

- [ACI 외부 포워딩 문제 해결](#)
- [Cisco 기술 지원 및 다운로드](#)

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.