

복원력 있는 이더넷 프로토콜 검토

목차

[소개](#)

[지원되는 플랫폼](#)

[배경 정보](#)

[왜 담당자인가?](#)

[혜택](#)

[제한 사항](#)

[프로토콜 작업](#)

[세그먼트](#)

[링크 상태 레이어](#)

[책임](#)

[포트 상태](#)

[패킷 세부사항](#)

[하드웨어 플러드 레이어\(HFL\)](#)

[BPA](#)

[고려 사항](#)

[BPA 동작](#)

[하드웨어 지원](#)

[환경 보호](#)

[세그먼트 통계](#)

[세그먼트 완료 조건 탐지](#)

[VLAN 로드 밸런싱 시작](#)

[PDU 형식](#)

[문제 해결](#)

[끊어진 링크 조사](#)

[대체\(ALT\) 포트](#)

[REP Adjacency 문제 해결](#)

[디버그](#)

[유용한 디버그](#)

[덜 유용한 디버그](#)

[관련 정보](#)

소개

이 문서에서는 REP(Resilient Ethernet Protocol)의 개요를 설명합니다.

지원되는 플랫폼

- DSBU(Desktop Switching Business Unit) Metro Switch(3750ME 및 ME3400) 릴리스 12.2(40)SE 이상
- Cisco Catalyst 4500 Series Switch Release 12.2(44)SG 이상

- Whitney2에서 시작하는 Cisco Catalyst 6500 Series 스위치(12.2SXI)
- Cisco Catalyst 7600 Series 라우터(12.2SRC) 시작

배경 정보

왜 담당자인가?

REP는 일부 특정 레이어 2 네트워크 설계에서 STP(Spanning Tree Protocol)를 대체하기 위해 사용되는 프로토콜입니다. 최신 STP 사양은 802.1Q-2005에 정의된 MST(Multiple Spanning Tree)입니다. MST에 대한 대안을 원하는 사용자에게는 다음과 같은 합법적 문제가 있습니다.

- STP는 브리지 도메인을 전체적으로 간주합니다. 그 결과 임의 원격 링크의 상태를 변경하면 로컬 장애가 복구됩니다. STP의 명백한 예측 불가능성은 브리지된 도메인을 작고 독립적인 조각으로 분할하는 경우에만 완화됩니다. 불행히도, 이것은 (모든 시나리오의 루프 방지와 같은) 스패닝 트리에서 일부 주요 기능을 제거하지 않고서는 달성하는 데 복잡하지만 불가능하지는 않습니다.
- STP 통합은 회선 스위칭 기술에서 흔히 사용되는 50ms(밀리초)의 복구 시간을 예상하는 통신 사업자에게 느린 것처럼 보일 수 있습니다. 이러한 느낌은 프로토콜 자체에서 발생하지 않습니다. 플랫폼에서 STP를 보다 효율적인 방법으로 실행하려면 최적화가 필요합니다. 그 동안 플랫폼 제약을 해결할 새로운 솔루션이 필요합니다.
- MST 로드 밸런싱 컨피그레이션은 유연하지 않습니다. MST가 인스턴스 로드 밸런싱을 달성하려면 모든 브리지가 동일한 영역에 속해야 합니다. 영역은 사용자 컨피그레이션에 따라 정의되며, 네트워크에 일부 재컨버전스를 도입하지 않으면 스위치에서 MST 컨피그레이션을 수정할 방법이 없습니다. 이는 신중한 사전 컨피그레이션을 통해 해결할 수 있으며, 제한된 범위에서는 VTP(VLAN Trunk Protocol) v3와 같은 다른 프로토콜을 사용할 수 있습니다.

혜택

REP 가 누릴 수 있는 몇 가지 이점은 다음과 같습니다.

- REP 는 다음과 같은 통합 시간을 제공합니다.
 - 3750ME는 20ms에서 79ms 사이의 수렴을 나타냅니다.
 - ME3400은 40ms에서 70ms 사이에서 수렴합니다.
- 현재 하드웨어에서 작동
- 예측 가능하고 차단된 포트
- 손쉬운 구성

제한 사항

다음은 REP 의 몇 가지 제한 사항입니다.

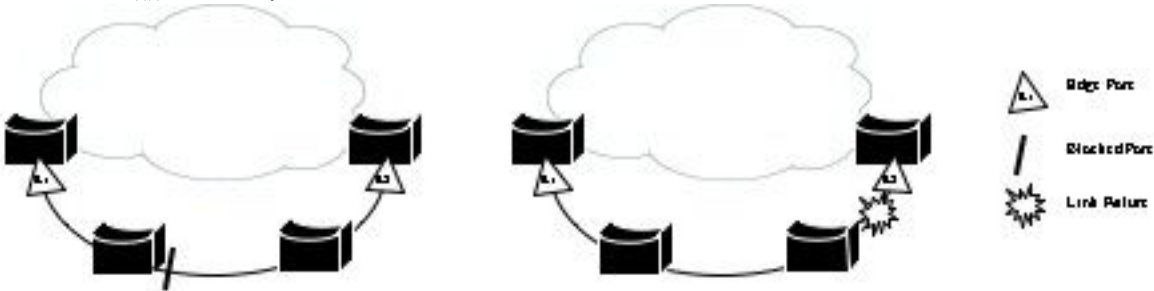
- 플러그 앤 플레이 없음
- 컨피그레이션 오류 방지(루프 생성이 용이함)
- 제한된 양의 리던던시(단 하나의 링크 장애만 견딜 수 있음)
- 전역 토폴로지를 검색할 수 없습니다(세그먼트 토폴로지만).

프로토콜 작업

세그먼트

REP 는 세그먼트를 최소 네트워크 구성 요소로 사용합니다. 세그먼트는 단순히 함께 연결된 포트의 모음입니다. 브리지에서 지정된 세그먼트에 속한 포트는 2개뿐이며 각 세그먼트 포트는 최대 1개의 외부 인접 디바이스를 가질 수 있습니다. 세그먼트의 정의는 전적으로 사용자 컨피그레이션에 의해 이루어집니다. 세그먼트는 사용자에게 의해 또한 결정된 2개의 에지 포트에 의해 종료된다. 세그먼트에서 실행되는 REP 프로토콜은 가능한 최소한이며 다음 속성만 보장합니다.

- 세그먼트의 모든 포트가 온라인 상태이고 작동 중인 경우, 그중 하나가 각 VLAN에 대한 트래픽을 논리적으로 차단합니다.
- 세그먼트의 하나 이상의 포트가 어떤 이유로든 작동하지 않을 경우 나머지 모든 작동 포트는 모든 VLAN을 위해 전달됩니다.
- 링크 장애 시 남아 있는 모든 운영 포트의 차단을 최대한 빨리 해제할 수 있습니다. 마찬가지로, 마지막으로 장애가 발생한 포트가 다시 작동하게 되면, VLAN당 논리적으로 차단된 포트가 하나 선택되었을 때 네트워크의 중단을 최대한 적게 유발해야 합니다.



구성 요소로서의 세그먼트

그림 1. 단순한

그림 1은 4개의 브리지에 분산된 6개의 포트를 포함하는 세그먼트의 예를 보여줍니다. 구성된 에지 포트 E1 및 E2는 다이어그램에서 삼각형으로 표시되고, 논리적으로 차단된 포트는 막대로 표시됩니다. 왼쪽 그림과 같이 모든 포트가 작동 중이면 단일 포트가 차단됩니다. 네트워크에 장애가 발생하면 오른쪽 다이어그램에 표시된 것처럼 논리적으로 차단된 포트가 전달 상태로 돌아갑니다.

그림 1과 같이 세그먼트가 열려 있으면 두 에지 포트 간에 연결이 제공되지 않습니다. REP 에지 스위치 간의 연결은 세그먼트 외부에 있는 것으로 간주됩니다(STP를 통해). 컨버전스 속도를 높이기 위해 REP 세그먼트에 장애가 발생할 경우 선택적인 컨피그레이션으로 STP TCN(Topology Change Notification)이 생성됩니다.

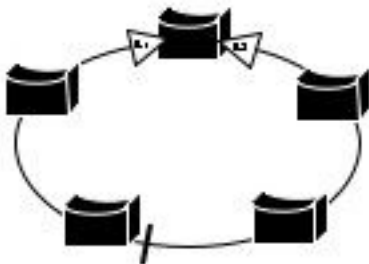


그림 2. 세그먼트를 링으로 래핑할 수 있습니다.

그림 2와 같이 두 에지 포트가 동일한 스위치에 있으면 세그먼트가 링으로 감깁니다. 이 경우 세그먼트를 통해 에지 포트 간에 연결이 이루어집니다. 실제로 이 컨피그레이션을 사용하면 세그먼트의 두 스위치 간에 이중 연결을 생성할 수 있습니다.

그림 1과 그림 2에서 보여주는 것처럼 열린 세그먼트와 닫힌 세그먼트를 조합하여 사용하면 다양한 네트워크 설계를 구현할 수 있습니다.

링크 상태 레이어

책임

- 고유한 네이버와의 연결을 설정합니다.
- 주기적으로 네이버와의 연결 무결성을 확인합니다.
- 상위 계층 상태 시스템에 대한 메시지 송수신
- 네이버에서 수신된 데이터를 승인합니다.
- PDU(Protocol Data Unit)의 속도 제한

포트 상태

포트가 REP에 대해 구성되면 다음과 같은 상태가 됩니다.

- 실패 상태(차단)
 - 네이버 관계 형성됨:
 - 대체 포트(차단 및 작동 중)
 - AP(액세스 포인트) 선택 실패:
 - 열린 포트(다른 포트에서 'AP'를 선택한 경우)
- 다음과 같은 상황에서는 포트가 작동하지 않습니다.
- 포트에서 감지된 인접 디바이스 없음
 - 포트에서 둘 이상의 인접 디바이스가 탐지됨
 - 인접 디바이스가 메시지를 승인(ACK)하지 않음

패킷 세부사항

기본적으로 REP 는 기능을 실행하지 않는 디바이스에 의해 삭제되도록 네이티브 VLAN(태그가 지정되지 않음)의 BPDU(Bridge Protocol Data Unit) 클래스 MAC 주소로 hello 패킷을 전송합니다. 각 LSL(Link Status Layer) PDU에는 전송된 PDU의 시퀀스 번호와 마지막으로 수신된 PDU의 원격 시퀀스 번호가 모두 포함됩니다. 이를 통해 포트 간의 안정적인 전송이 보장됩니다. 각 인접 디바이스는 ACK가 수신될 때까지 전송된 각 PDU의 사본을 보관합니다. ACK가 수신되지 않으면 타이머가 만료된 후 재전송됩니다.

실제 LSL PDU에는 다음이 포함되어 있습니다.

- ProtocolVersion(현재 0)
- 세그먼트 ID
- 원격 포트 ID
- 로컬 포트 ID
- 로컬 시퀀스 번호
- 원격 시퀀스 번호
- 상위 레이어 TLV

LSL 패킷은 각 hello 간격으로 또는 상위 레이어 프로토콜에서 요청할 때 전송됩니다. LSL PDU를 빌드하면 먼저 SegmentID 및 LocalPortID와 같은 자체 필드가 채워집니다. 그런 다음 BPA(Block

Port Advertisement) 또는 EPA(End Port Advertisement)와 같은 상위 레이어 프로토콜 대기열을 검사하여 추가 데이터를 대기열에 추가해야 하는지 확인합니다.

하드웨어 플러드 레이어(HFL)

HFL은 링크 장애 후 신속한 컨버전스를 지원하는 REP 모듈입니다. LSL과 같은 BPDU MAC 주소로 PDU를 보내지 않고 오히려 REP 관리 VLAN의 특수 MAC 주소(0100.0ccc.ccce)로 멀티캐스트 PDU를 보냅니다. 이렇게 하면 하드웨어로 세그먼트의 모든 스위치에 플러딩됩니다.

HFL 패킷 형식은 간단합니다.

- 프로토콜 버전(계속 0)
- 세그먼트 ID
- TLV(Higher Layer Type Length) 값

이때 HFL을 통해 전송되는 TLV는 BPA가 유일하다.

BPA

BPA는 포트 우선순위와 함께 차단되는 VLAN을 알리기 위해 AP에 의해 전송됩니다. 이렇게 하면 세그먼트에 링크 오류를 알리고 VLAN당 세그먼트당 단일 AP만 있음을 확인할 수 있습니다. 이것은 달성하기가 쉽지 않다.

고려 사항

안정적인 토폴로지에서는 AP 선택이 간단합니다. 온라인 상태의 포트는 모든 VLAN의 AP로 시작됩니다(차단). 우선순위가 더 높은 다른 포트에서 BPA를 수신하면 안전하게 차단을 해제할 수 있음을 알고 있습니다. 세그먼트의 포트에 장애가 발생하면 다른 포트의 차단을 해제하기 위해 동일한 프로세스가 사용됩니다. 장애가 발생한 모든 포트는 현재 AP보다 높은 포트 우선순위(우선순위에 실패한 비트가 있는)를 생성하며, 이로 인해 현재 AP의 차단이 해제됩니다.

그러나 이 링크가 다시 작동하면 문제가 발생합니다. 이 경우 우선 순위에서 실패한 비트가 지워지고 우선 순위가 정상으로 돌아갑니다. 이 포트가 새 우선 순위를 알고 있더라도 세그먼트의 다른 부분은 이 포트의 오래된 BPA 정보를 가질 수 있습니다. 이 다이어그램은 다음 시나리오를 보여줍니다.

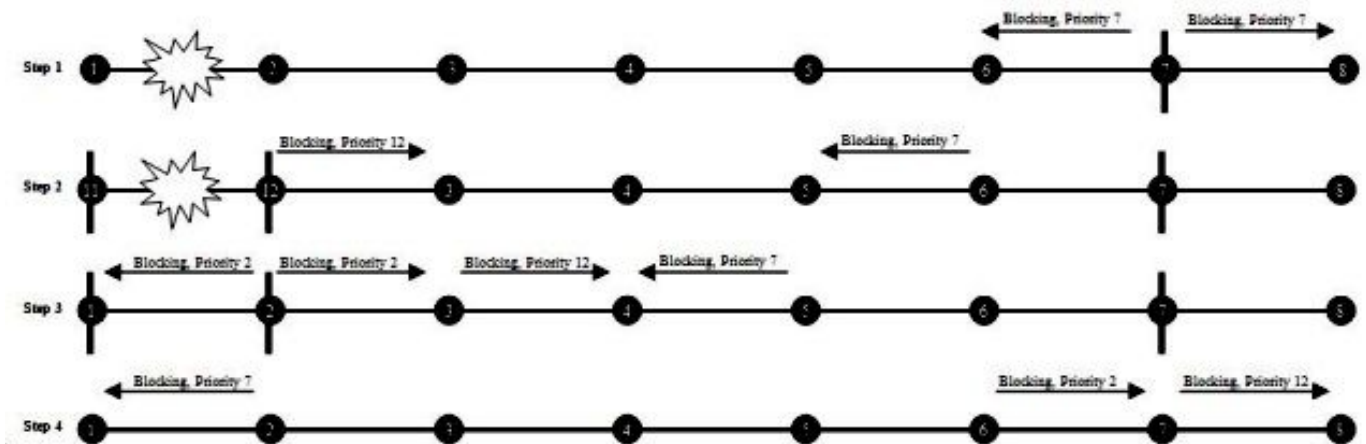


그림 3. 세그먼트를 여는 부실 정보

이 시나리오의 초기에는 포트 7이 차단을 수행하고 우선 순위를 7로 광고합니다. 다음으로, 11과 12 사이의 링크가 중단되고, 12가 12의 우선 순위를 가지고 차단 중임을 나타내는 BPA를 전송합니다.

이러한 차단 포트가 다른 포트의 BPA를 수신하기 전에 포트 12가 다시 작동하여 작동합니다. 곧 이어 포트 12가 우선 순위 7의 포트 7의 BPA를 수신하므로 차단이 해제됩니다. 그러면 포트 7이 우선 순위 12를 사용하여 포트 12에서 오래된 BPA를 가져오므로 차단이 해제됩니다. 이로 인해 루프가 발생합니다. 이 경합 상태는 BPA가 키를 사용하는 이유이다.

BPA 동작

각 포트는 다음 정보를 사용하여 포트 우선순위를 계산합니다.

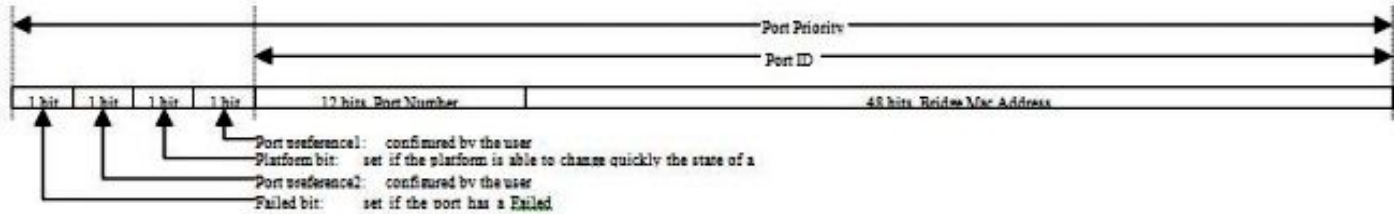


그림 4. 포트 우선 순위

이제는 실패한 포트가 항상 세그먼트에서 AP로 선택되는 이유가 분명합니다. 포트가 Failed(실패)에서 Alternate(대체)로 이동하면 해당 포트 ID 및 난수를 기반으로 고유한 키를 생성하고 해당 포트 ID와 함께 광고합니다. AP는 로컬 키를 포함하는 차단된 포트에서 메시지를 수신하는 경우에만 차단을 해제합니다. 이 메커니즘은 이전 섹션에서 설명한 경합 상태 시나리오를 방지하는 데 도움이 됩니다. 다음은 포트가 작동 및 작동 중지될 때 발생하는 상황을 보여 주는 다이어그램입니다.

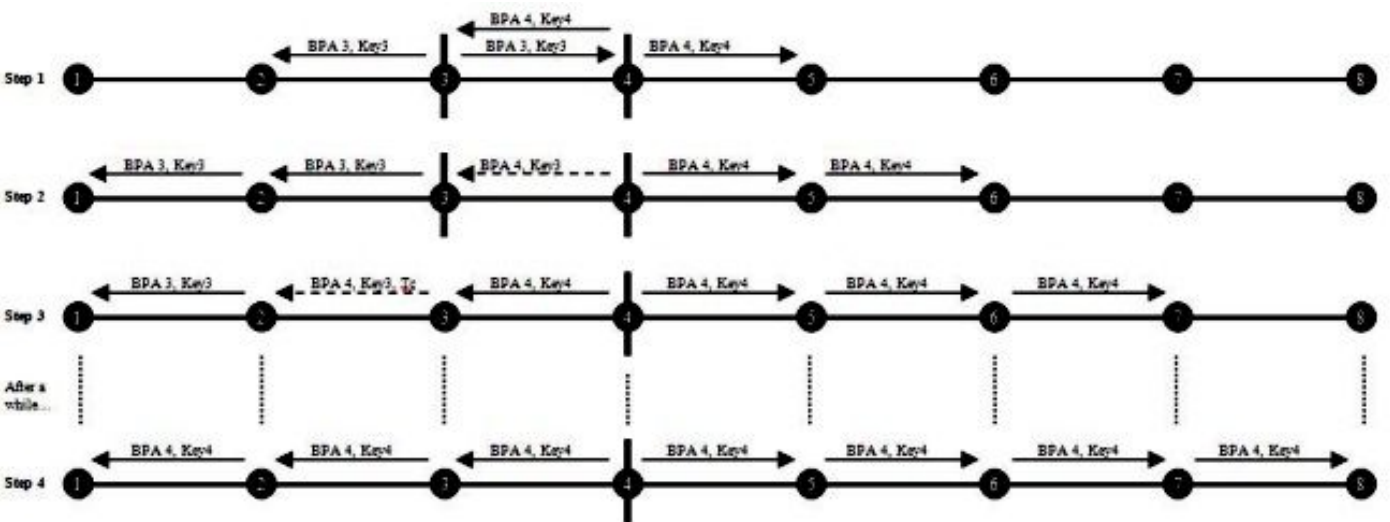


그림 5. 링크 업 시 BPA 운영

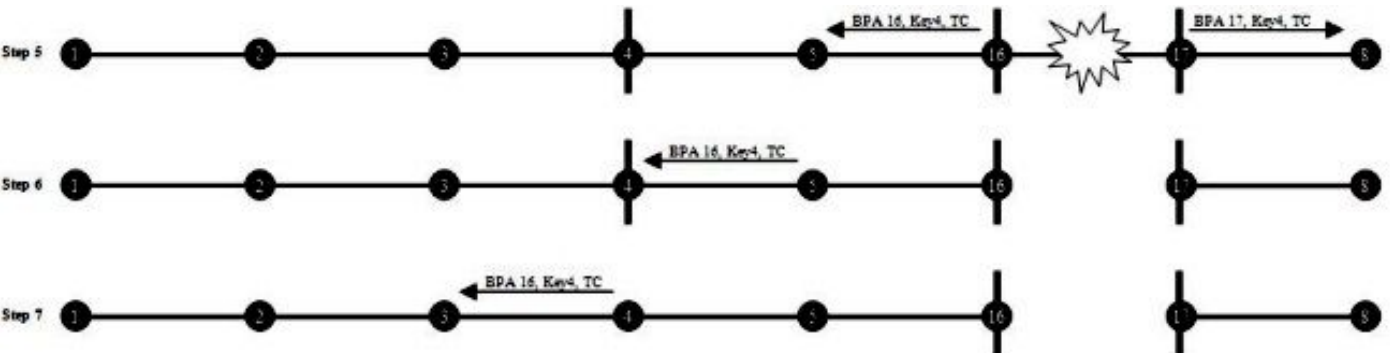


그림 6. 링크 실패 후 BPA 작업

하드웨어 지원

세그먼트에서 링크 실패가 발생하면 HFL을 통해 BPA가 나머지 세그먼트에 플러딩됩니다. 이를 완

전히 적용하려면 관리 VLAN이 모든 세그먼트 포트에서 전송되어야 하며 세그먼트 외부의 에지 포트 간에 전송되어야 합니다. HFL이 신뢰할 수 있는 수송을 보장할 수 없기 때문에 BPA도 LSL을 통해 이 정보를 보낸다. HFL 딜리버리에 문제가 있는 경우 LSL은 재컨버전스가 발생하도록 합니다.

환경 보호

엔드 포트는 에지 포트 또는 장애가 발생한 포트입니다. 세그먼트가 에지 포트에 의해 양쪽에서 종료될 경우 완전하다고 간주되며 VLAN 로드 밸런싱이 가능합니다. 장애가 발생한 포트가 세그먼트가 종료되면 모든 포트가 열려 있기 때문에 로드 밸런싱이 가능하지 않습니다.

엔드 포트는 LSL을 통해 릴레이되는 EPA를 주기적으로 전송합니다. 이러한 메시지는 다음과 같습니다.

- 세그먼트에 대한 통계 전파
- 세그먼트 완료 조건 탐지
- VLAN 로드 밸런싱 시작

세그먼트 통계

각 엔드 포트는 LSL을 통해 자신에 대한 정보를 포함하는 주기적인 EPA를 보냅니다. 각 중간 포트는 자체 정보를 추가하고 EPA를 릴레이합니다. 이러한 메시지는 양방향으로 이동하므로 REP에 참여하는 각 스위치에서 전체 REP 세그먼트에 대해 알고 있어야 합니다. EPA에 포함된 정보는 다음을 포함합니다.

- 브리지 ID
- 두 REP 참여 포트의 포트 ID 및 상태

세그먼트 완료 조건 탐지

각 에지 포트는 자체 에지 우선 순위 및 특수 키(BPA 키와 관련이 없음)가 포함된 특수 선택 EPA 메시지를 전송합니다. 이를 수신할 첫 번째 포트는 이 메시지에 자체 포트 우선순위를 두고 다음 스위치로 릴레이합니다. 경로에 있는 각 스위치는 자체 포트 우선순위를 EPA에 있는 것과 비교하고, 우선순위가 더 높은 경우 자체 포트에 대체합니다. 에지 포트가 EPA를 받으면 에지 우선 순위와 자체 우선순위를 비교합니다. 수신된 EPA에 우선순위가 더 높은 경우, 에지 포트는 키와 함께 다음 EPA 메시지를 기본 에지로 전송합니다. 이 메커니즘은 다음 두 가지를 달성하는 데 도움이 됩니다.

- 세그먼트가 완전한지 확인합니다.
- 두 에지 포트에 모두 가장 높은 우선순위의 중간 포트에 대한 지식 제공

VLAN 로드 밸런싱 시작

VLAN 로드 밸런싱은 서로 다른 VLAN을 차단하는 서로 다른 두 AP를 통해 이루어집니다. 기본 에지는 VLAN의 하위 집합 이상에서 AP를 담당하며, AP는 우선순위가 가장 높은 포트가 나머지를 차단하도록 지시하는 EPA 메시지를 전송합니다. 우선순위가 가장 높은 중간 포트에 대한 정보는 EPA 선택 메시지와 함께 이미 가져왔습니다. 이를 위해 생성되는 메시지 유형은 우선 순위가 가장 높은 포트가 차단해야 하는 VLAN의 비트맵이 포함된 EPA 명령 TLV입니다.

PDU 형식

EPA 헤더:

- 유형=EPA
- 인스턴스 번호
- 선택적 TLV

선거 TLV:

- edge우선순위
- 에지 키
- 최상의 포트 우선 순위

명령 TLV:

- 선택된포트우선순위
- 선택된 VLAN

정보 TLV:

- 브리지 ID
- 포트 ID 2개
- 포트 역할

문제 해결

끊어진 링크 조사

다음은 좋은 토폴로지의 예입니다.

```
SwitchA#show rep topology
REP Segment 1
BridgeName PortName edge Role
-----
SwitchA Fa0/2 Pri Alt
SwitchC Fa1/0/23 Open
SwitchC Fa1/0/2 Open
SwitchD Fa0/23 Open
SwitchD Fa0/2 Open
SwitchB Fa1/0/23 Sec Open
```

다음은 뭔가가 깨진 예입니다.

```
SwitchA#show rep topology
REP Segment 1
Warning: REP detects a segment failure, topology may be incomplete
BridgeName PortName edge Role
-----
SwitchA Fa0/2 Sec Open
SwitchC Fa1/0/23 Open
SwitchC Fa1/0/2 Fail
```

예전의 모습은 다음과 같습니다.

```
SwitchA#show rep topology archive
REP Segment 1
```



```

BridgeName PortName edge Role
-----
SwitchA Fa0/2 Pri Open
SwitchC Fa1/0/23 Open
SwitchC Fa1/0/2 Open
SwitchD Fa0/23 Open
SwitchD Fa0/2 Open
SwitchB Fa1/0/23 Sec Alt

```

실패한 SwitchC와 SwitchD 간의 링크에 대한 자세한 내용을 보려면 다음 명령을 입력합니다.

```

SwitchA#show rep topology archive detail
REP Segment 1
<snip>
SwitchC, Fa1/0/2 (Intermediate)
Open Port, all vlans forwarding
Bridge MAC: 0017.5959.c680
Port Number: 004
Port Priority: 010
Neighbor Number: 3 / [-4]
SwitchD, Fa0/23 (Intermediate)
Open Port, all vlans forwarding
Bridge MAC: 0019.e73c.6f00
Port Number: 019
Port Priority: 000
Neighbor Number: 4 / [-3]
<snip>

```

링크를 다시 불러온 후의 모습은 다음과 같습니다.

```

SwitchA#show rep topology
REP Segment 1
BridgeName PortName edge Role
-----
SwitchA Fa0/2 Pri Open
SwitchC Fa1/0/23 Open
SwitchC Fa1/0/2 Alt
SwitchD Fa0/23 Open
SwitchD Fa0/2 Open
SwitchB Fa1/0/23 Sec Open

```

이전에 장애가 발생한 포트는 AP로 남아 계속 차단됩니다. AP 선택은 차단된 포트 간에만 발생하기 때문입니다. 이 링크가 실패하면 토폴로지의 다른 모든 포트가 열립니다. 링크가 작동하면 SwitchC와 SwitchD 모두 우선 순위가 지정된 BPA를 전송했습니다. SwitchC Fa1/0/2가 우선순위가 더 높아서 AP가 되었습니다. 이는 토폴로지의 다른 포트가 실패할 때까지 또는 선점이 수행될 때까지 유지됩니다.

대체(ALT) 포트

ALT 포트는 일부 또는 모든 VLAN을 차단합니다. REP 세그먼트에 오류가 있으면 ALT 포트가 없고 모든 포트가 열려 있습니다. 이것이 REP 가 장애가 발생할 때 데이터 트래픽에 대한 활성 경로를 제공할 수 있는 방법입니다.

완전한 REP 세그먼트(장애가 없는 경우)에는 ALT 포트가 하나 있거나 ALT 포트가 두 개 있습니다. VLAN 로드 밸런싱이 활성화된 경우 세그먼트에 두 개의 ALT 포트가 있습니다. ALT 포트 중 하나는 지정된 VLAN 집합을 차단하고 다른 ALT 포트는 항상 기본 에지에 있으며 보완적인 VLAN 집합을 차단합니다. VLAN 로드 밸런싱이 활성화되어 있지 않으면 세그먼트에 단일 ALT 포트가 있어 모든 VLAN이 차단됩니다.

포트가 온라인 상태가 되는 순서와 내장된 포트 우선 순위에 따라 세그먼트의 어떤 포트가 ALT 포트가 되는지 결정됩니다. 특정 포트를 ALT 포트에 설정하려면 preferred 키워드로 구성합니다. 예를 들면 다음과 같습니다.

```
interface gig3/10
rep segment 3 edge preferred
```

gig3/1이 기본 에지이고 VLAN 로드 밸런싱을 구성한다고 가정해 보겠습니다.

```
interface gig3/1
rep segment 3 edge primary
rep block port preferred vlan 1-150
```

이러한 구성에서 선점 후 포트 gig3/10은 VLAN 1부터 150까지 차단하는 ALT 포트이고, 포트 gig3/1은 VLAN 151부터 4094까지 차단하는 ALT 포트입니다.

선점은 rep preempt segment 3 명령을 사용하여 수동으로 수행하거나, 기본 에지 포트 아래에서 rep preempt delay <seconds>를 구성한 경우 자동으로 수행됩니다.

링크 장애 후 세그먼트가 복구되면 장애에 인접한 두 포트 중 하나가 ALT 포트에 나타납니다. 선점 후 ALT 포트의 위치는 컨피그레이션에 지정된 대로 변경됩니다.

REP Adjacency 문제 해결

인접성이 있는지 확인하려면 다음 명령을 입력합니다.

```
SwitchC#show interface fa1/0/23 rep
Interface Seg-id Type LinkOp Role
```

```
-----
FastEthernet1/0/23 1 TWO_WAY Open
```

자세한 내용을 보려면 다음 명령을 입력합니다.

```
SwitchC#show interface fa1/0/23 rep detail
```

```
FastEthernet1/0/23 REP enabled
Segment-id: 1 (Segment)
PortID: 001900175959C680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 000400175959C6808335
Port Role: Open
Blocked VLAN: <empty>
Admin-vlan: 1
Preempt Delay Timer: disabled
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 255547, tx: 184557
HFL PDU rx: 3, tx: 2
BPA TLV rx: 176096, tx: 2649
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 870, tx: 109
EPA-COMMAND TLV rx: 2, tx: 2
EPA-INFO TLV rx: 45732, tx: 45733
```

디버그

대부분의 디버그는 너무 많은 출력을 인쇄하여 유용하게 사용할 수 없습니다. 전체 목록은 다음과 같습니다(일부는 서비스 내부에서만 사용 가능).

```
SwitchB#debug rep ?
all all debug options
bpa-event bpa events
bpasm BPA state machine
database protocol database
epasm EPA state machine
error protocol errors
failure-recovery switchover events
lsism LSL state machine
misc miscellaneous
packet protocol PDU
prsm Port Role state machine
showcli show debug info
```

유용한 디버그

몇 가지 유용한 디버그는 다음과 같습니다.

- **debug rep showcli** (needs service internal) - 이 debug는 일반 show rep 명령을 입력할 때 많은 추가 정보를 출력합니다.
- **debug rep error** - 이 디버그는 매우 유용할 가능성이 있습니다.
- **debug rep failure-recovery** - 이 debug는 링크가 끊어질 때 이동하는 메시지를 인쇄합니다.

```
*Mar 5 05:01:11.530: REP LSL-OP Rx EXT Local (Fa0/23 seg:1, tc:1, frs:0) prio:
*Mar 5 05:01:11.530: 0x80 0x00 0x19 0x00 0x17 0x59 0x59 0xC6
*Mar 5 05:01:11.530: 0x80
*Mar 5 05:01:11.530: REP Flush from Fa0/23 to REP, sending msg
*Mar 5 05:01:11.530: REP LSL-OP Rx INT Local (Fa0/2 seg:1, tc:1, frs:0) prio:
*Mar 5 05:01:11.530: 0x80 0x00 0x19 0x00 0x17 0x59 0x59 0xC6
*Mar 5 05:01:11.530: 0x80
*Mar 5 05:01:11.530: REP Flush from Fa0/2 to REP, sending msg
```

- **debug rep prsm** - 이 디버그는 생성되지 않는 인접성을 트러블슈팅하는 데 적합합니다. 링크 업/다운에서 발생하는 상황에 대한 Play-by-Play를 제공합니다.

```
4d05h: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state to up
4d05h: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2,
changed state to up
```

```
*Mar 5 05:06:19.098: rep_pr Fa0/2 - pr: during state FAILED_PORT,
got event 5(no_ext_neighbor)
*Mar 5 05:06:19.098: @@@ rep_pr Fa0/2 - pr: FAILED_PORT ->
FAILED_PORT_NO_EXT_NEIGHBOR[Fa0/2]rep_pr_act_no_ext_neighbor@272:
PRSM->fp_no_ext_neighbor state
[Fa0/2]rep_pr_lsl_event_handler@448: REP_MSG_EXT_PEER_GONE rcvd
```

```
4d05h: %REP-4-LINKSTATUS: FastEthernet0/2 (segment 1) is operational
```

```
*Mar 5 05:06:22.236: rep_pr Fa0/2 - pr: during state FAILED_PORT_NO_EXT_
NEIGHBOR, got event 0(link_op)
*Mar 5 05:06:22.236: @@@ rep_pr Fa0/2 - pr:
FAILED_PORT_NO_EXT_NEIGHBOR ->
ALTERNATE_PORT[Fa0/2]rep_pr_act_ap@162: PRSM->alternate state
[Fa0/2]rep_pr_lsl_event_handler@431: REP_MSG_LINKOP_TRUE rcvd
```

```
*Mar 5 05:06:23.125: rep_pr Fa0/2 - pr: during state ALTERNATE_PORT,
```

```
got event 2(pre_empty_ind)
*Mar 5 05:06:23.133: @@@ rep_pr Fa0/2 - pr: ALTERNATE_PORT -> UNBLOCK_VLANS_ACT
*Mar 5 05:06:23.133: rep_pr Fa0/2 - pr: during state UNBLOCK_VLANS_ACT,
got event 3(no_local_block_vlan)
*Mar 5 05:06:23.133: @@@ rep_pr Fa0/2 - pr: UNBLOCK_VLANS_ACT ->
OPEN_PORT[Fa0/2]rep_pr_act_op@252: PRSM->active state
[Fa0/2]rep_pr_act_uva@222: PRSM unblock vlans
[Fa0/2]rep_pr_sm_preempt_ind@374: Posting pre empty indication
```

- **debug rep epasm** - 이 디버그는 토폴로지 변경 시 유용한 정보를 제공합니다. 세그먼트가 안정적이면 아무것도 인쇄되지 않습니다.

포트가 오프라인 상태가 되면 다음과 같이 출력됩니다.

```
*Mar 5 04:48:31.463: rep_epa_non_edge Fa0/2 - epa-non-edge: during state
INTERMEDIATE_PORT, got event 1(lr_eq_fp)*Mar 5 04:48:31.463: @@@ rep_epa_non_
edge Fa0/2 - epa-non-edge: INTERMEDIATE_PORT -> FAILED_PORT[Fa0/2]rep_epa_non_
edge_act_failed_port@164: Trigger archiving
[Fa0/23]rep_epa_set_peer_archive_flag@1084: set arch flag
[Fa0/2]rep_epa_non_edge_act_failed_port@171: no edge, failed port
*Mar 5 04:48:35.473: rep_epa_non_edge Fa0/2 - epa-non-edge: during state
FAILED_PORT, got event 0(epa_hello_tmo)
*Mar 5 04:48:35.473: @@@ rep_epa_non_edge Fa0/2 - epa-non-edge: FAILED_PORT ->
FAILED_PORT[Fa0/2]rep_epa_non_edge_act_periodic_tx@90: archiving on port down
[Fa0/2]rep_epa_copy_topology@913: deip=0x3396F18,pe=0,se=1,fp=0,ap=0,op=2
[Fa0/23]rep_epa_non_edge_handle_info_tlv@1560: archiving on internal flag
[Fa0/23]rep_epa_copy_topology@913: deip=0x33961F0,pe=1,se=0,fp=0,ap=1,op=3
[Fa0/2]rep_epa_non_edge_act_periodic_tx@102: epa non edge, send info tlv
[Fa0/23]rep_epa_set_peer_archive_flag@1084: set arch flag
[Fa0/2]rep_epa_non_edge_handle_election_tlv@325: archiving on seg cfg change
[Fa0/2]rep_epa_copy_topology@913: deip=0x3396F18,pe=0,se=1,fp=0,ap=0,op=2
[Fa0/2]rep_epa_set_peer_archive_flag@1084: set arch flag
[Fa0/23]rep_epa_non_edge_handle_election_tlv@325: archiving on seg cfg change
[Fa0/23]rep_epa_copy_topology@913: deip=0x33961F0,pe=1,se=0,fp=0,ap=1,op=3
[Fa0/2]rep_epa_non_edge_handle_info_tlv@1560: archiving on internal flag
[Fa0/2]rep_epa_copy_topology@913: deip=0x3396F18,pe=0,se=1,fp=0,ap=0,op=2
```

포트가 온라인 상태가 될 때의 출력은 다음과 같습니다.

```
*Mar 5 04:49:39.982: rep_epa_non_edge Fa0/2 - epa-non-edge: during state FAILED_PORT,
got event 2(lr_neq_fp)
*Mar 5 04:49:39.982: @@@ rep_epa_non_edge Fa0/2 - epa-non-edge: FAILED_PORT ->
INTERMEDIATE_PORT[Fa0/2]rep_epa_non_edge_stop_timer@123: epa non edge, stop the timer
[Fa0/2]rep_epa_copy_topology@913: deip=0x32E2FA4,pe=0,se=1,fp=0,ap=1,op=1
[Fa0/2]rep_epa_copy_to_stable_topology@1040: copy to stbl
[Fa0/23]rep_epa_copy_topology@913: deip=0x3ACFFB8,pe=1,se=0,fp=0,ap=0,op=4
[Fa0/23]rep_epa_copy_to_stable_topology@1040: copy to stbl
```

덜 유용한 디버그

- **debug rep bpa-event** - 이 debug는 BPA를 언제 받고 무엇을 하는지 알려줍니다. 초당 4개의 줄이 있어요.

```
[Fa0/23]: BPA: Sending ext pak to bparx
[Fa0/2]: BPA: Enqueued internal pak
[Fa0/2]: BPA: Sending int msg to bparx
[Fa0/2]: BPA: Relay pak
[Fa0/2]: BPA: Enqueue ext pak
```

- **debug rep bpsasm** - 이 디버그는 BPA를 수신할 때마다 BPA 상태 머신이 수행하는 작업을 알려줍니다. 1초에 세줄이 있어요

```
*Mar 5 04:44:23.857:      rep_bpa_rx BPA RX sm: during state BPA_RX_IDLE,
got event 0(bpa_rx_bpa_msg)
*Mar 5 04:44:23.857: @@@ rep_bpa_rx BPA RX sm: BPA_RX_IDLE -> BPA_RX_IDLE
[Fa0/23]: BPA Rx sm: Received bpa: <internal> 0, <vlan_detail> 0
[Fa0/23]: BPA Rx sm: Role 2: TC 0; Internal 0; Frm Remote Segment 0
```

```
*Mar 5 04:44:23.857:      rep_bpa_rx BPA RX sm: during state BPA_RX_IDLE,
got event 0 (bpa_rx_bpa_msg)
*Mar 5 04:44:23.857: @@@ rep_bpa_rx BPA RX sm: BPA_RX_IDLE -> BPA_RX_IDLE
[Fa0/2]: BPA Rx sm: Received bpa: <internal> 1, <vlan_detail> 0
[Fa0/2]: BPA Rx sm: Role 2: TC 0; Internal 1; Frm Remote Segment 0
```

- **debug rep lsism - 이 디버그는 하위 레벨 LSL 메시지 처리를 덤프합니다.**

```
*Mar 5 05:03:10.564: REP Fa0/23 seq:4411 ACK'ed (ref: 1)
*Mar 5 05:03:10.564: REP Fa0/23 seq:4412 ACK'ed (ref: 1)
*Mar 5 05:03:10.564: REP LSL: Fa0/23 rx expected seq# (4744),
process it (TLV: 0).
*Mar 5 05:03:10.782: REP Fa0/2 seq:440 ACK'ed (ref: 1)
```

관련 정보

- [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.