

Catalyst 스위치에서 백본 빠른 구성 및 이해

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[BPDU 및 비교 방법](#)

[간접 링크 오류에서 STP를 복구하는 방법](#)

[표준 STP의 백본 빠른 향상](#)

[간접 링크 실패 탐지](#)

[간접 링크 장애에 대응](#)

[루트 링크 쿼리 PDU](#)

[백본 고속 기능이 활성화된 시나리오 예](#)

[CatOS 및 Cisco IOS용 백본 고속 구성](#)

[CatOS 구성](#)

[Cisco IOS용 컨피그레이션](#)

[관련 정보](#)

소개

이 문서에서는 백본을 빠르게 구성하는 방법에 대해 설명합니다. 백본 fast는 브리지 네트워크의 모든 스위치에서 활성화되면 간접 링크 장애로부터 복구할 때 스위치를 최대 20초(max_age)까지 저장할 수 있는 Cisco 독점 기능입니다. 일부 STP(Spanning-Tree Protocol) 기본 사항을 신속하게 검토한 후, 백본이 빠르게 적용되는 정확한 실패 시나리오와 CatOS 및 Cisco IOS® 소프트웨어를 모두 실행하는 Catalyst 스위치에 대해 이를 구성하는 방법을 확인할 수 있습니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- Cisco IOS Software 릴리스 12.1(6)EA2 이상을 실행하는 Catalyst 2950 Series 스위치
- Cisco IOS Software 릴리스 12.1(4)EA1 이상을 실행하는 Catalyst 3550 Series 스위치
- CatOS 5.1(1a) 이상을 실행하는 Catalyst 4000 Series 스위치
- Cisco IOS Software 릴리스 12.1(8a)EW 이상을 실행하는 Catalyst 4500/4000 Series 스위치
- CatOS 버전 4.1(1) 이상을 실행하는 Catalyst 5500/5000 Series 스위치
- CatOS 버전 5.1(1)CSX 이상을 실행하는 Catalyst 6500/6000 Series 스위치
- Cisco IOS Software 릴리스 12.0-7XE 이상을 실행하는 Catalyst 6500/6000 Series 스위치

BPDU 및 비교 방법

BPDU(Bridge Protocol Data Units)는 담당 필드에 따라 엄격하게 분류할 수 있습니다. 이러한 필드 중에는 루트 브리지 ID, 루트에 대한 경로 비용 및 발신자 브리지 ID가 있습니다. BPDU는 다음과 같은 이유로 다른 BPDU보다 더 나은 것으로 간주됩니다.

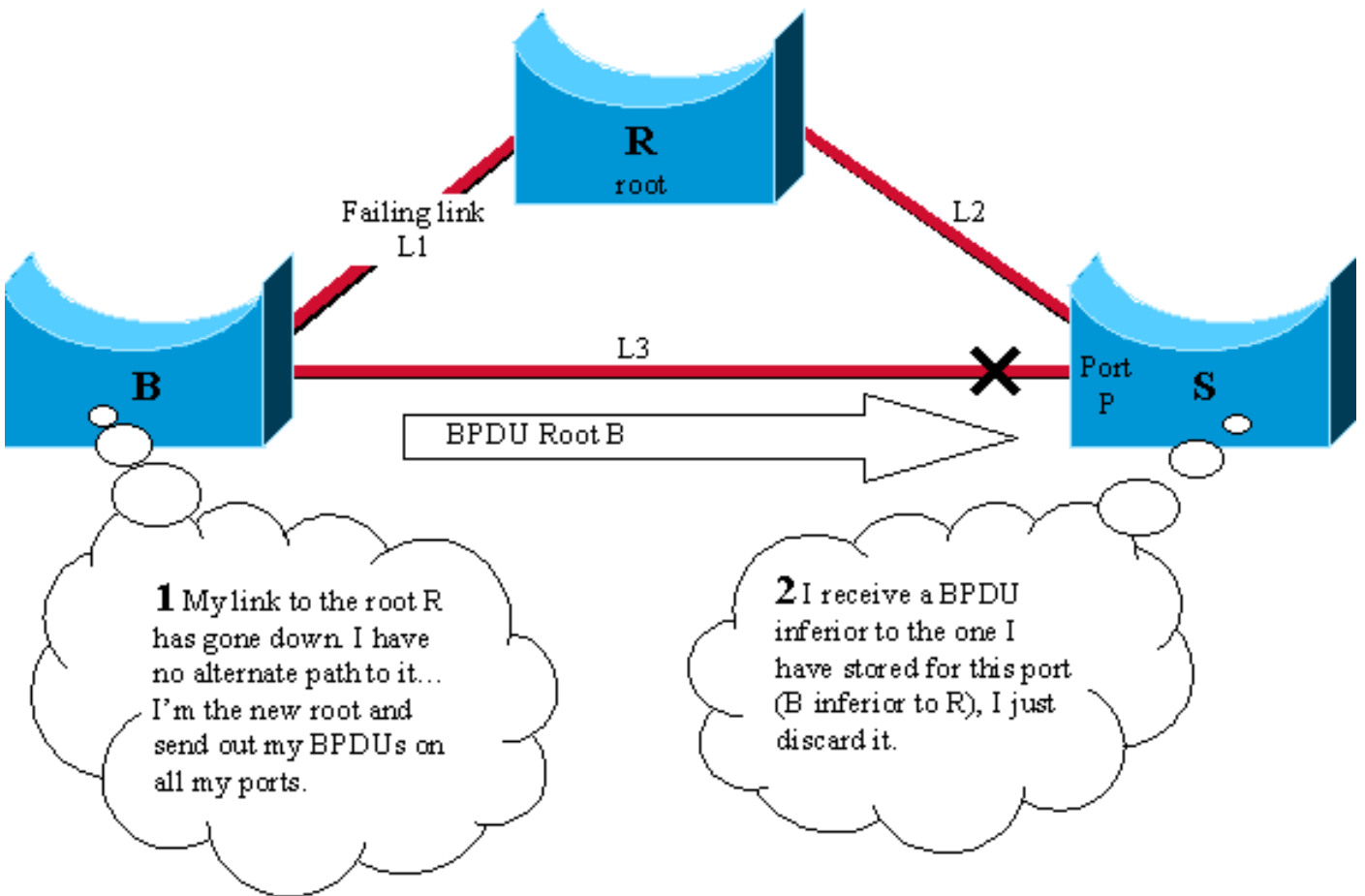
- 한 BPDU가 다른 BPDU보다 더 나은 루트 브리지 ID를 전달하는 경우 값이 낮을수록 우수합니다.
- 루트 브리지 ID 값이 같으면 루트에 대한 경로 비용이 가장 낮은 BPDU가 더 좋습니다.
- 루트 브리지 ID 값이 동일하고 루트에 대한 비용이 동일할 경우 발신자 브리지 ID가 더 나은 BPDU가 더 좋습니다. 값이 낮을수록 우수합니다.

그러면 동점으로 작용할 수 있는 다른 변수가 있습니다. 그러나 BPDU가 높을수록 최상의 루트 브리지에 대한 액세스 권한이 향상됩니다.

보내는 포트보다 더 나은 포트에서 BPDU를 수신하는 브리지는 루트 포트가 아닌 경우 이 포트를 차단 모드로 전환합니다. 즉, 이 포트에 연결된 세그먼트에 지정된 브리지인 다른 브리지가 있습니다. 브리지는 BPDU의 값을 현재 지정된 브리지에서 보낸 포트에 저장합니다.

간접 링크 오류에서 STP를 복구하는 방법

이는 간접 링크 실패 후 다시 계산해야 할 때 STP가 어떻게 동작하는지, 즉, 직접 연결되지 않은 링크의 장애로 인해 브리지가 일부 포트의 상태를 변경해야 할 때 어떻게 동작하는지를 보여줍니다.



풀 메시 토폴로지에 3개의 스위치 R, B, S가 포함된 이 다이어그램을 생각해 보십시오. R이 루트 브리지이고 B가 백업 루트 브리지인 것으로 가정합니다. S는 포트 P를 차단하고 B는 링크 L3에 대해

지정된 브리지입니다.

1. 링크 L1이 다운되면 스위치 B가 즉시 장애를 감지하여 루트로 간주합니다. BPDU를 S에 전송하기 시작하며 새로운 루트가 될 것이라고 주장합니다.
2. S는 B에서 이 새 BPDU를 수신하면 포트 P에 저장된 BPDU보다 낫다는 것을 인식하고 무시합니다.
3. max_age 타이머가 만료되면(기본적으로 20초), 포트 P에 대해 S에 저장된 BPDU는 시간 초과됩니다. 포트가 즉시 수신 대기로 이동하고 S가 더 나은 BPDU를 B로 보내기 시작합니다.
4. B가 S에서 BPDU를 수신하면 BPDU 전송이 중지됩니다.
5. 포트 P는 수신 및 학습 상태를 통해 전달 상태로 이동합니다. 이 작업에는 fw_delay 값의 2배, 추가로 30초가 소요됩니다. 전체 연결이 복원됩니다.

이 간접 링크 오류에서 복구하는 데 max_age 값(20초)과 fw_delay 값(2x15초)의 두 배가 걸렸습니다. 기본 매개변수를 사용하는 시간은 50초입니다. 백본 빠른 기능은 max_age(20초)를 저장하도록 제안합니다. 이를 위해 포트가 하위 BPDU를 수신한 직후 타임아웃됩니다.

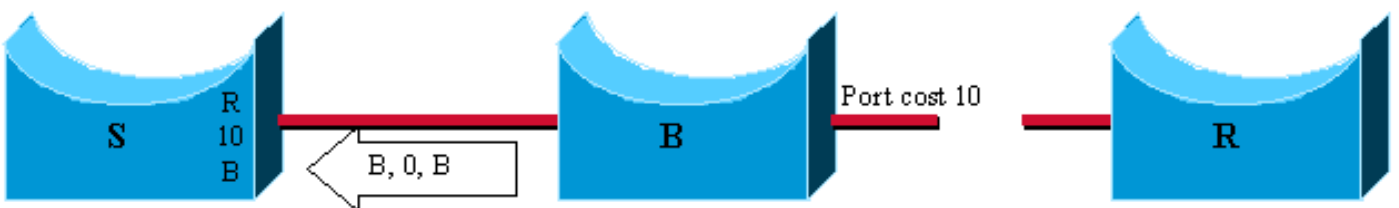
표준 STP의 백본 빠른 향상

앞의 예제에서 STP는 간접 링크 장애로 인해 잘못된 정보가 되는 정보를 무효화합니다. 이를 위해 max_age를 수동적으로 기다립니다. 이 max_age 지연을 제거하기 위해 백본 fast는 두 가지 향상된 기능을 제공합니다.

- 가능한 한 빨리 간접 링크 장애를 탐지하는 기능. 이는 지정된 브리지가 직접 링크 장애가 발생할 때 전송하는 하위 BPDU를 추적하여 실현됩니다.
- 포트에 저장된 BPDU 정보가 여전히 유효한지 즉시 확인할 수 있는 메커니즘. 이는 새로운 PDU(Protocol Data Unit) 및 루트 링크 쿼리(이 문서에서 RLQ PDU라고 함)를 사용하여 구현됩니다.

간접 링크 실패 탐지

지정된 브리지에서 포트에 하위 BPDU를 수신하면 이 브리지는 루트가 손실되어 더 높은 브리지 ID를 가진 루트를 광고하기 시작합니다. 이 루트는 우리보다 더 나쁜 루트입니다.



In this case, B lost the root and sends a BPDU with root id B, path cost 0 and bridge id B. It is inferior to the one that S had stored, because R is a better root than B.

IEEE(Institute of Electrical and Electronics Engineers) 사양에 대한 일반적인 동작은 낮은 BPDU를 무시하기만 하는 것입니다. 백본은 빠른 속도로 사용합니다. 수신되는 즉시 루트의 경로에서 오류가 발생했으며 적어도 하나의 포트를 에이징해야 하기 때문입니다.

참고: 네트워크에서 하위 BPDU 생성 없이 간접 링크 장애가 발생할 수 있습니다. 이전 다이어그램에서 허브를 추가하면 됩니다.

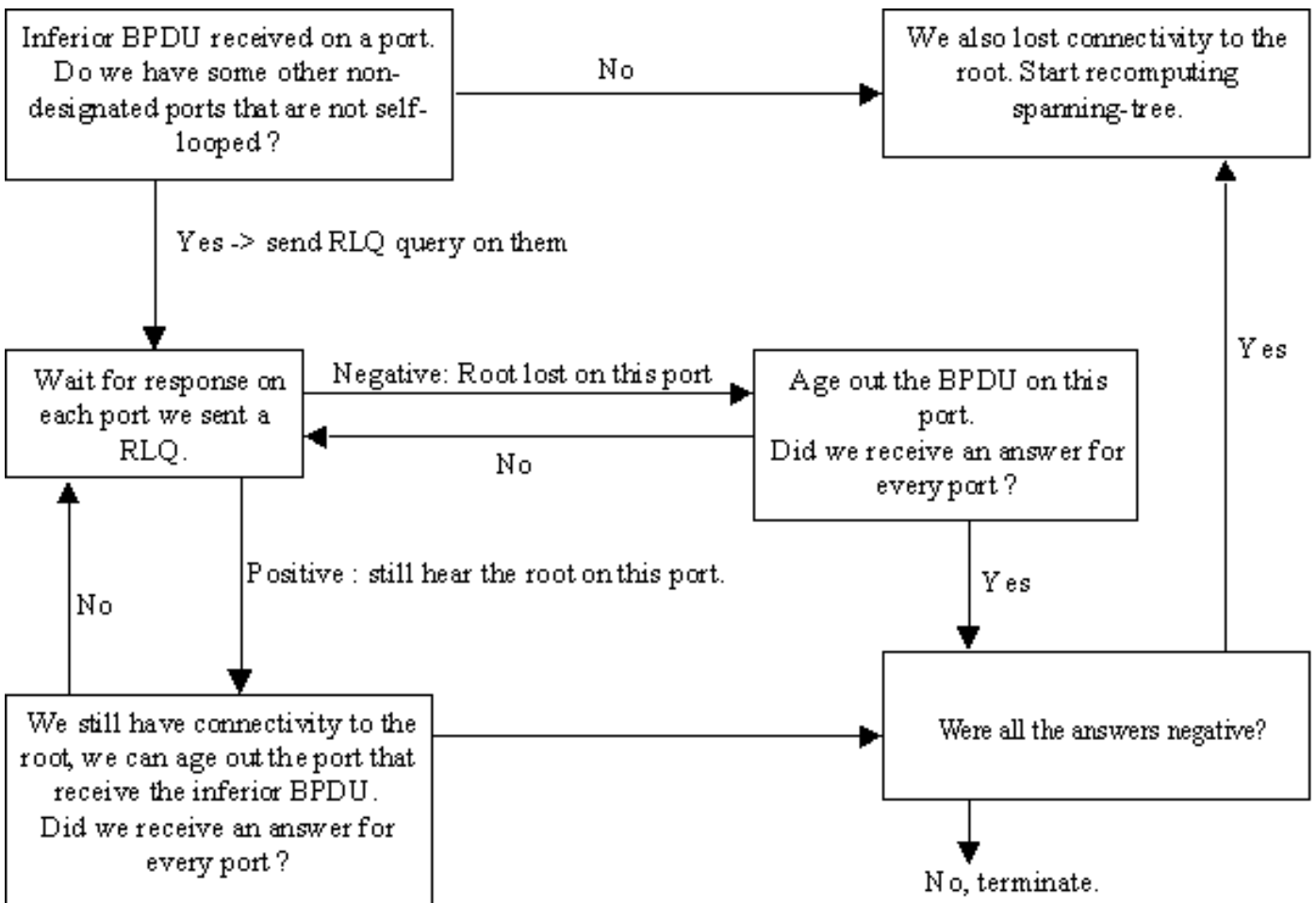


루트 브리지 R과 허브 간에 링크 오류가 발생합니다. B는 링크가 중단되었음을 감지하지 못하고 max_age가 새 루트임을 주장하기 전에 기다립니다. 브리지가 직접 링크 오류를 탐지하는 경우에만 메커니즘이 작동합니다.

지정된 브리지에서 전송하는 하위 BPDU를 추적하기만 하면 됩니다. 이는 포트에 저장되는 BPDU입니다. 예를 들어 새로 삽입된 브리지가 하위 BPDU를 전송하기 시작하면 백본 고속 기능이 시작되지 않습니다.

간접 링크 장애에 대응

비지정 포트에서 하위 BPDU가 탐지되면 백본의 두 번째 단계가 빠르게 트리거됩니다. 장애 발생 시 영향을 받을 수 있는 포트의 에이징을 max_age로 수동적으로 기다리는 대신 RLQ PDU를 통해 즉시 테스트할 수 있는 사전 대응적 방법이 도입되었습니다. RLQ는 지정되지 않은 포트에서 루트에 대해 일종의 ping을 수행하는 데 사용되며, 포트에 저장된 BPDU가 여전히 유효한지 또는 폐기되어야 하는지 신속하게 확인할 수 있습니다.



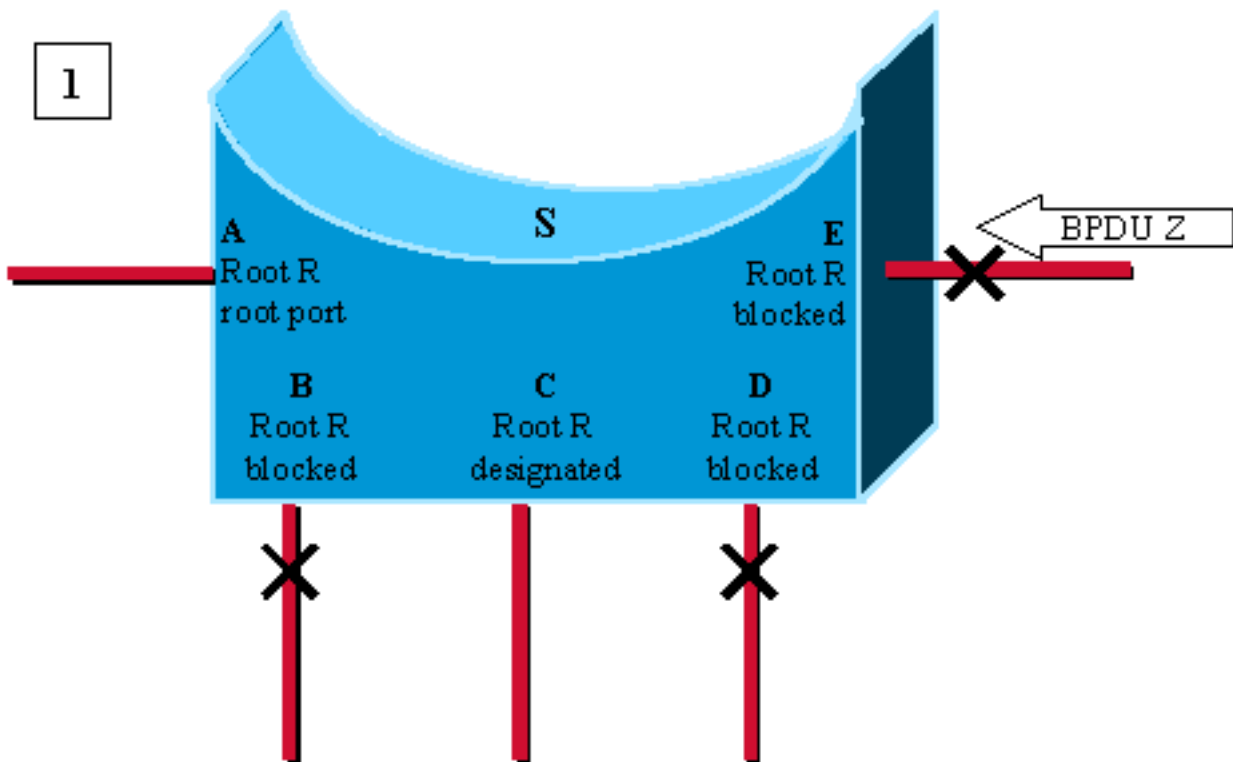
지정된 브리지에서 하위 BPDU를 수신하면 하위 BPDU와 자체 루프가 있는 포트를 제외한 모든 비지정 포트에 RLQ PDU를 전송합니다. 이는 BPDU를 수신하는 데 사용되는 포트의 루트에서 계속 수신되는지 확인하기 위한 것입니다. 하위 BPDU를 수신한 포트는 장애가 발생했음을 이미 알고 있기 때문에 제외되며, 자체 루프가 있고 지정된 포트는 루트로 연결되지 않으므로 유용하지 않습니다.

포트에서 RLQ 응답을 수신하면 응답이 음수이면 포트가 루트에 대한 연결이 끊기고 해당 BPDU를 사용할 수 있습니다. 또한, 지정되지 않은 다른 모든 포트가 이미 부정적인 답변을 받은 경우, 전체 브리지는 루트가 손실되고 STP 계산을 처음부터 시작할 수 있습니다.

이 포트를 통해 루트 브리지에 계속 액세스할 수 있다는 응답이 확인되면 처음에 하위 BPDU를 수신한 포트를 즉시 사용할 수 있습니다.

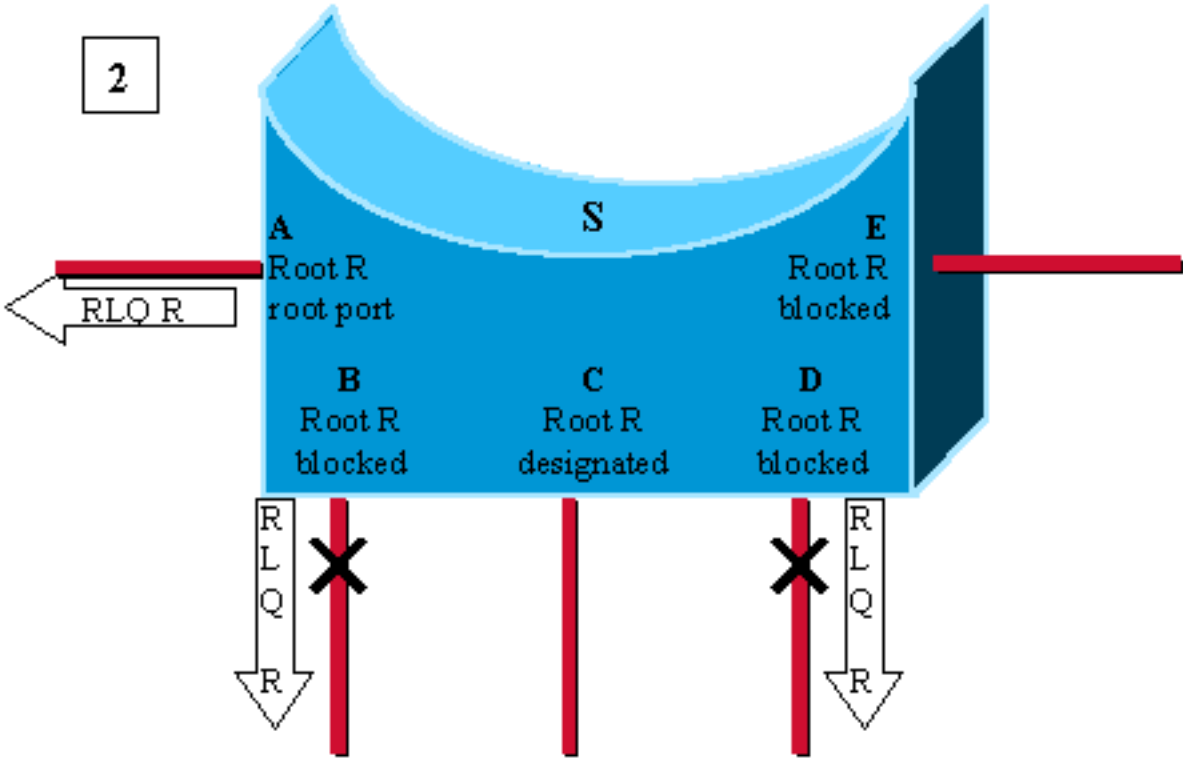
이 예에서 포트 A, B, D 및 E는 스위치 S에 대해 지정되지 않은 포트입니다. A는 루트 포트이고 나머지는 차단 중입니다. E가 하위 BPDU(1)를 수신하면 백본이 빠르게 작동하여 STP 재계산 속도를 높일 수 있습니다.

E(2)를 제외한 모든 비 지정 포트에서 루트 R을 찾는 RLQ 요청을 보냅니다. 회신은 이러한 포트를 통해 액세스할 수 있는 루트를 지정합니다. D가 수신하는 RLQ 응답은 D가 루트 R에 대한 경로를 손실했음을 나타내며 BPDU를 즉시 사용하지 않도록 지정합니다(3). 포트 A와 B는 여전히 R(4)에 대한 경로가 있음을 확인합니다. 따라서 스위치 S는 루트에 계속 연결되어 있으므로 포트 E를 즉시 종료하고 일반 STP 규칙(5)을 계속 사용합니다.



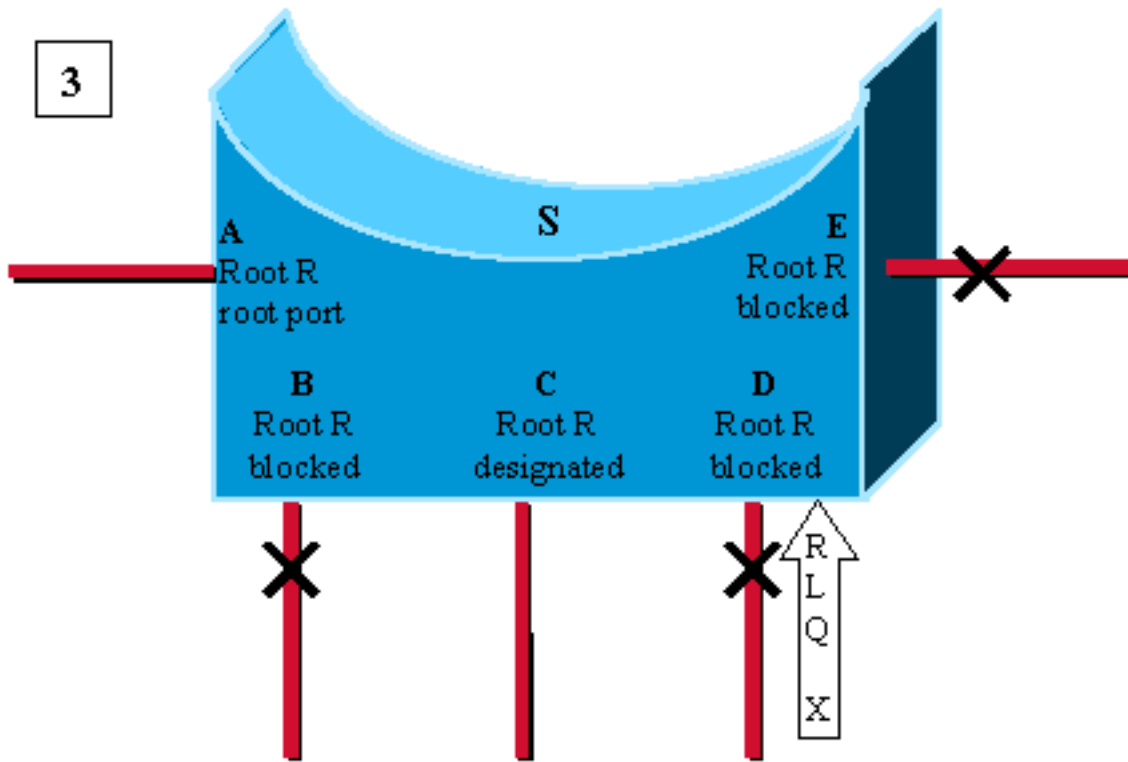
Port E receives an inferior BPDU, advertising root Z instead of root R stored on the different ports.

2



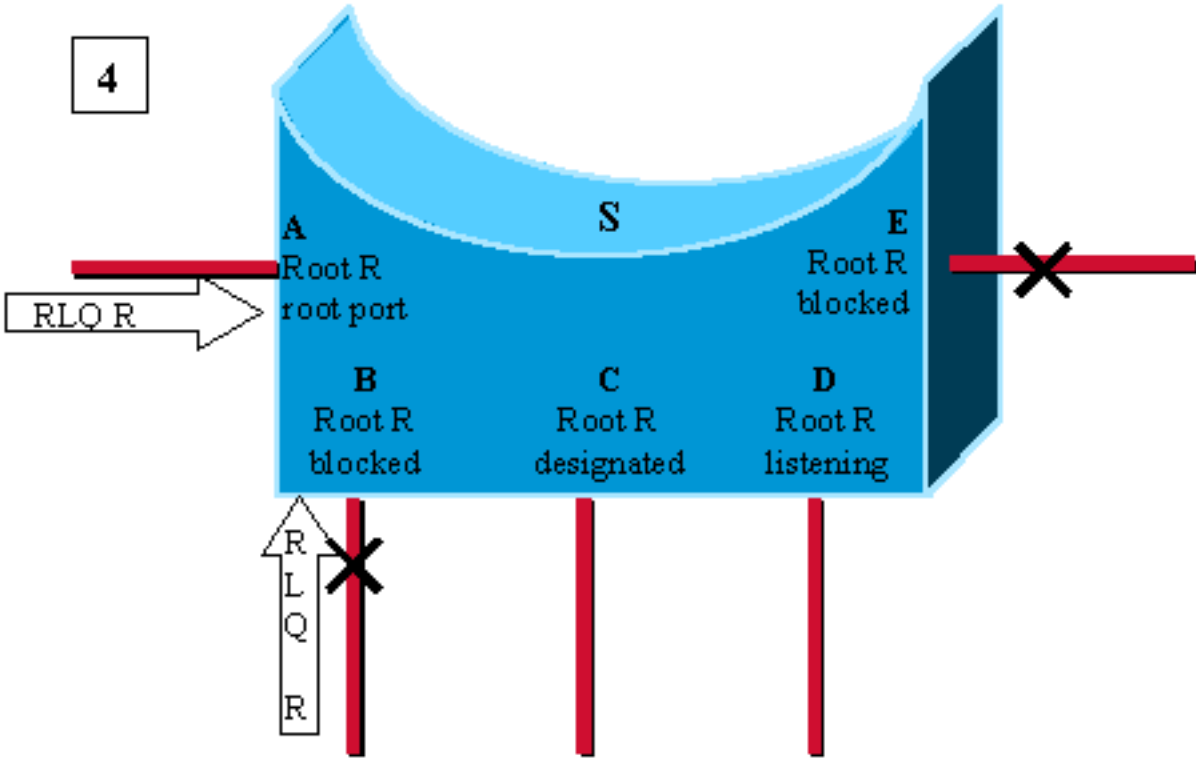
Switch S needs to recheck all its other non-designated ports. It sends out a RLQ request for root R on ports A,B and D.

3



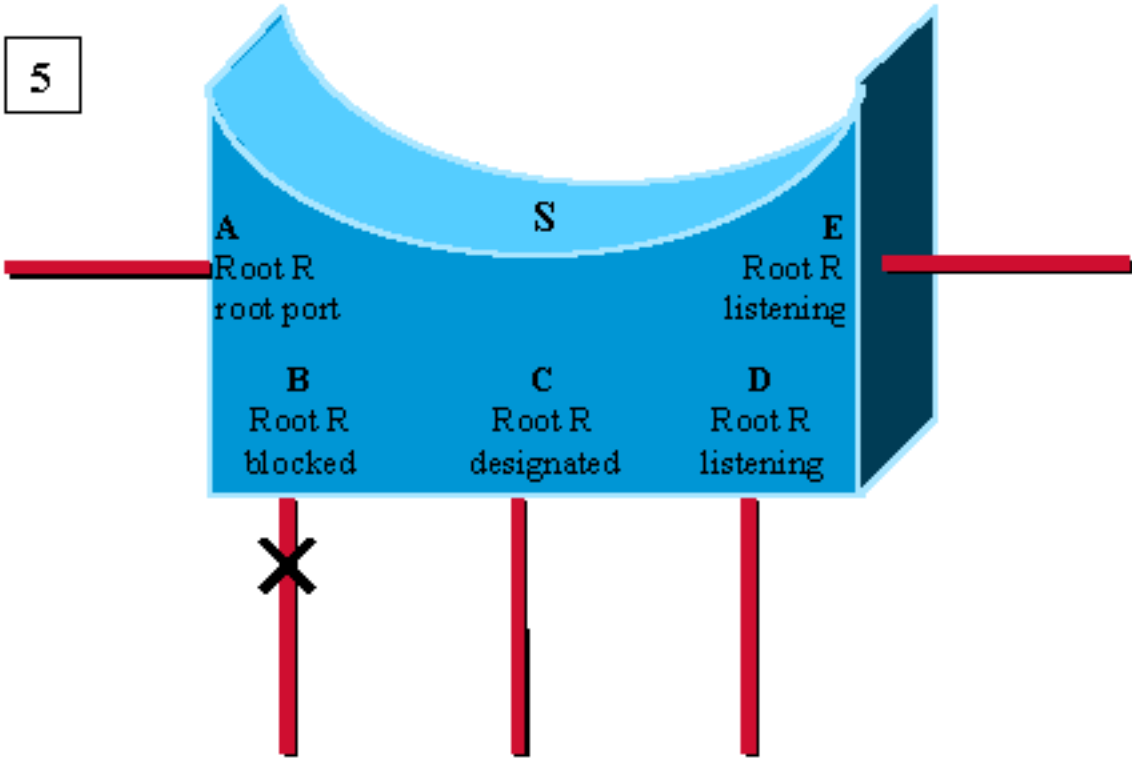
Port D is the first to receive and RLQ response from bridge X claiming to be the root. It is a negative response: D has lost connectivity to the root R. We age out immediately the BPDU on port D and go to listening. As we don't know if we still have connectivity to the root R, we don't age out port E yet.

4



Here, A and B receive a RLQ response confirming R as being the root. As switch S still has connectivity to the root, we can age out immediately the BPDUs stored on port E.

5



Port E transitions to listening, without waiting for max_age. Usual spanning-tree rules then apply to determine whether E and D will eventually go to blocking or forwarding.

스위치에서 루트가 R과 다른 응답만 수신한 경우, 루트는 손실된 것으로 간주하여 즉시 STP 계산을 다시 시작합니다. 이 경우는 브리지의 지정되지 않은(및 자체 루프가 없는) 유일한 포트가 루트 포트이고 이 포트에서 하위 BPDU를 수신할 때도 발생합니다.

루트 링크 쿼리 PDU

RLQ의 두 가지 형식은 RLQ 요청 및 RLQ 응답입니다.

RLQ 요청은 일반적으로 BPDU를 수신하는 포트에서 전송되므로 이 포트를 통해 루트에 계속 연결되어 있는지 확인합니다. 요청에서 어떤 브리지가 루트이고 RLQ 응답이 이 포트를 통해 액세스할 수 있는 루트 브리지와 함께 다시 오도록 지정합니다. 두 루트가 동일하면 연결이 여전히 활성 상태이고 그렇지 않으면 연결이 끊깁니다.

RLQ 요청을 수신하는 브리지는 RLQ 쿼리에 지정된 루트 브리지와 다른 루트 브리지가 있고 루트 인지 여부 때문에 쿼리한 루트에 대한 연결이 손실된 것을 알고 있으면 즉시 응답합니다.

그렇지 않으면 루트 포트를 통해 쿼리를 루트로 전달합니다.

지정된 포트에서 RLQ 응답이 플래딩됩니다. RLQ 요청의 발신자는 브리지 ID를 PDU에 넣습니다. 이는 자체 쿼리에 대한 회신을 다시 받을 때 지정된 포트에서 응답을 플래딩하지 않도록 하기 위한 것입니다.

RLQ PDU는 일반 STP BPDU와 동일한 패킷 구조를 갖습니다. 유일한 차이점은 두 개의 서로 다른

Cisco 특정 SNAP 주소가 사용된다는 것입니다.요청과 응답에 대해 하나씩 입력합니다.

표준 BPDU 형식은 다음과 같습니다.

DA SA 길이 DSAP SSAP CNTL 스냅 PDU

PDU 필드는 다음과 같습니다.

프로토콜 식별자	버전	메시지 유형	플래그	루트 ID	루트 경로 비용
발신자 ID	포트 ID	메시지 기간	최대 사용 기간	hello 시간	전달 지연

PDU에 사용되는 메시지 유형은 표준 BPDU와 다릅니다.

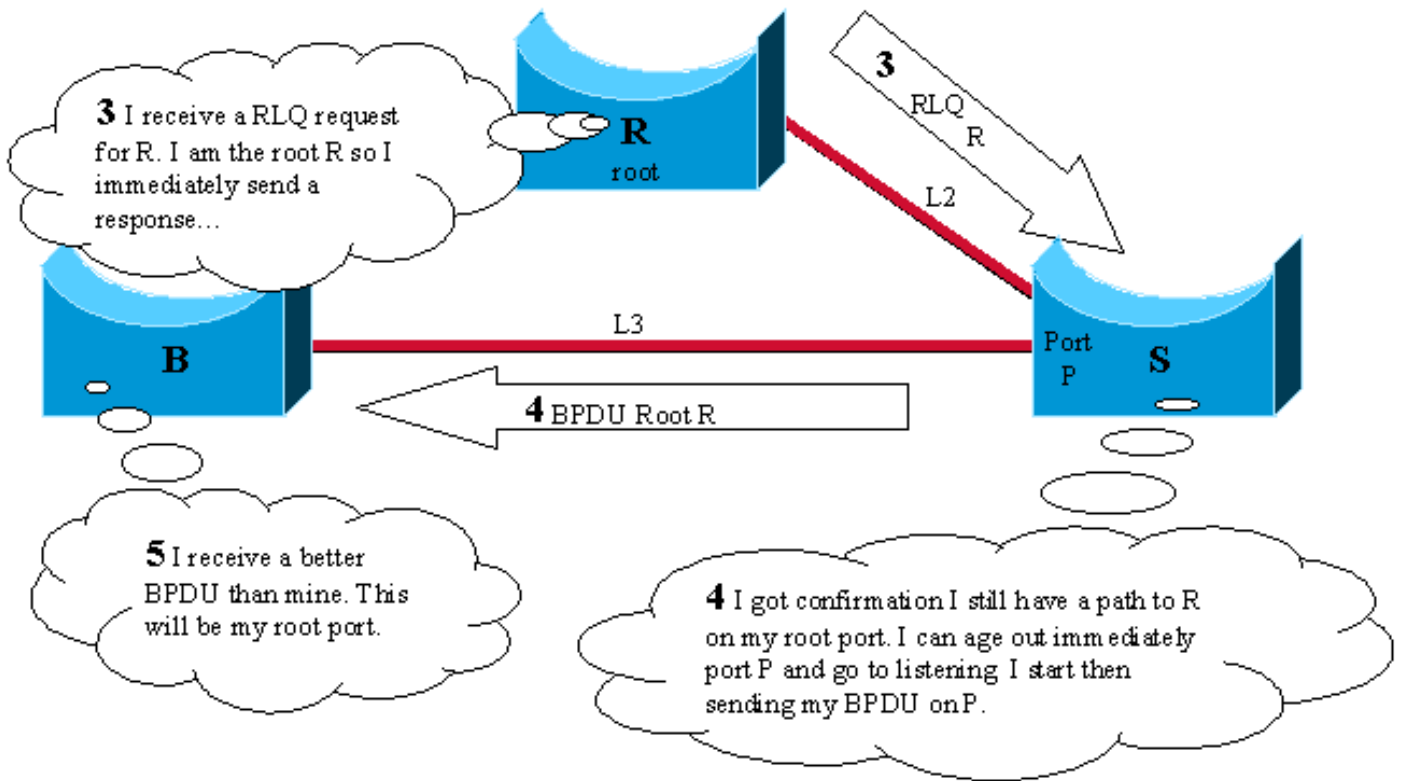
사용되는 필드는 루트 ID와 발신자 브리지 ID뿐입니다.

이러한 PDU를 처리하려면 네트워크의 모든 스위치에서 이 Cisco 고유의 기능을 구성해야 합니다.

백본 고속 기능이 활성화된 시나리오 예

이 시나리오는 첫 번째 예를 기반으로 하지만 이번에는 세 스위치에서 백본이 빠르게 활성화되었습니다.

1. 첫 번째 단계는 앞서 설명한 것과 정확히 동일합니다.
2. S가 B에서 하위 BPDU를 수신하면 max_age를 기다리지 않고 지정되지 않은 포트를 재확인하기 시작합니다.루트 브리지 R을 위해 루트 포트에 RLQ 쿼리를 보냅니다.
3. 루트 브리지 R은 쿼리를 수신하고 RLQ 응답으로 즉시 응답합니다. 이 응답에서는 해당 방향으로 루트 R이 여전히 있음을 지정합니다.
4. 이제 는 지정되지 않은 모든 포트를 확인했으며 루트에 여전히 연결되어 있습니다.그런 다음 포트 P. P에 저장된 정보를 수신으로 전환하고 BPDU를 보내기 시작합니다.이 단계에서는 이미 max_age seconds를 저장했고 표준 STA(Spanning-Tree Algorithm)가 적용됩니다.
5. B는 S(R은 B보다 루트)에서 더 나은 BPDU를 수신하고 이제 L3로 연결되는 포트를 루트 포트로 간주합니다.



CatOS 및 Cisco IOS용 백본 고속 구성

루트 경로 안정성을 스위치에 알려려면 백본 빠른 속도로 RLQ 요청 및 회신 메커니즘을 사용해야 하므로, 백본 빠른 사용을 네트워크의 모든 스위치에서 활성화해야 합니다. RLQ 프로토콜은 스위치에서 백본 fast가 활성화된 경우에만 활성화됩니다. 또한 모든 스위치에서 백본 fast가 활성화되지 않은 경우 네트워크에서 RLQ 플러딩 문제가 발생할 수도 있습니다. 기본적으로 백본 fast는 비활성화되어 있습니다.

Catalyst 2900XL 및 3500XL 스위치에서는 백본 빠른 속도가 지원되지 않습니다. 일반적으로 스위치 도메인에 지원되는 다른 Catalyst 스위치 외에 이러한 스위치가 포함된 경우 백본을 빠르게 활성화해야 합니다. 엄격한 토폴로지에서 XL 스위치가 있는 환경에서 백본을 빠르게 구현할 경우 XL 스위치가 마지막 스위치이며 두 위치에서 코어에만 연결된 기능을 활성화할 수 있습니다. XL 스위치의 아키텍처가 데이지 체인 방식으로 되어 있는 경우에는 이 기능을 구현하지 마십시오.

RSTP 또는 IEEE 802.1w를 사용하여 백본을 빠르게 구성할 필요가 없습니다. 메커니즘은 기본적으로 포함되며 RSTP에서 자동으로 활성화됩니다. RSTP 또는 IEEE 802.1w에 대한 자세한 내용은 PVST+에서 [Rapid-PVST Migration Configuration Example](#)으로 [스패닝 트리를](#) 참조하십시오.

CatOS 구성

CatOS를 실행하는 Catalyst 4000, 5000 및 6000 시리즈 스위치의 경우, 이러한 명령을 사용하여 모든 포트에서 백본을 신속하게 지원하고 구성을 확인합니다.

```
Console> (enable) set spantree backbonefast enable
Backbonefast enabled for all VLANs
Console> (enable) show spantree backbonefast
! This command show that the backbonefast feature is enabled. Backbonefast is enabled. Console>
(enable)
```

백본 빠른 통계를 표시하려면

```

Console> (enable) show spantree summary
Summary of connected spanning tree ports by vlan
Uplinkfast disabled for bridge.
Backbonefast enabled for bridge.
Vlan  Blocking Listening Learning Forwarding STP Active
-----
1      0      0      0      1      1

      Blocking Listening Learning Forwarding STP Active
-----
Total  0      0      0      1      1

```

BackboneFast statistics

```

! The show spantree summary command displays all backbonefast statistics. -----
- Number of inferior BPDUs received (all VLANs): 0 Number of RLQ req PDUs received (all VLANs):
0 Number of RLQ res PDUs received (all VLANs): 0 Number of RLQ req PDUs transmitted (all VLANs):
0 Number of RLQ res PDUs transmitted (all VLANs): 0 Console> (enable)

```

Cisco IOS용 컨피그레이션

Cisco IOS 소프트웨어와 함께 실행되는 Catalyst 스위치의 경우 모든 인터페이스에 대해 백본을 빠르게 전역적으로 활성화하려면 이 명령을 사용합니다.

```

CAT-IOS# configure terminal
CAT-IOS(config)# spanning-tree backbonefast
CAT-IOS(config)# end
CAT-IOS#

```

백본 속도가 활성화되었는지 확인하고 통계를 표시하려면 다음을 수행합니다.

```

CAT-IOS# show spanning-tree backbonefast

```

BackboneFast is enabled

BackboneFast statistics

```

-----
Number of transition via backboneFast (all VLANs)           : 0
Number of inferior BPDUs received (all VLANs)              : 0
Number of RLQ request PDUs received (all VLANs)           : 0
Number of RLQ response PDUs received (all VLANs)          : 0
Number of RLQ request PDUs sent (all VLANs)                : 0
Number of RLQ response PDUs sent (all VLANs)              : 0
CAT-IOS#

```

관련 정보

- [PortFast 및 기타 명령을 사용하여 워크스테이션 시작 연결 지연 수정](#)
- [Cisco UplinkFast 기능 이해 및 구성](#)
- [Loop Guard 및 BPDU Skew Detection 기능을 사용한 스페닝 트리 프로토콜 개선 사항](#)
- [스패닝 트리 PortFast BPDU 가드 개선](#)
- [PVST+에서 Rapid-PVST 마이그레이션 컨피그레이션 예](#)
- [스패닝 트리 프로토콜 - 기술 지원](#)
- [스위치 - 제품 지원](#)
- [LAN 스위칭 - 기술 지원](#)

- [기술 지원 및 문서 - Cisco Systems](#)