

Comprensione e configurazione di Backbone Fast sugli switch Catalyst

Sommario

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[BPDU e modalità di confronto](#)

[Ripristino di STP in seguito a un errore di collegamento indiretto](#)

[Miglioramenti Backbone Fast per STP standard](#)

[Rileva errori di collegamento indiretti](#)

[Reazione a errori di collegamento indiretti](#)

[PDU query collegamento radice](#)

[Scenario di esempio con la funzione Backbone Fast abilitata](#)

[Configurazione di Backbone Fast per CatOS e Cisco IOS](#)

[Configurazione per CatOS](#)

[Configurazione per Cisco IOS](#)

[Informazioni correlate](#)

Introduzione

In questo documento viene descritto come configurare rapidamente la backbone. Backbone fast è una funzione proprietaria di Cisco che, una volta abilitata su tutti gli switch di una rete bridge, può salvare uno switch fino a 20 secondi (max_age) quando viene ripristinato da un errore di collegamento indiretto. Dopo una breve analisi di alcuni elementi base del protocollo Spanning-Tree Protocol (STP), è possibile verificare lo scenario di errore esatto a cui la backbone viene applicata rapidamente e come configurarla per gli switch Catalyst con software CatOS e Cisco IOS®.

Prerequisiti

Requisiti

Nessun requisito specifico previsto per questo documento.

Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Switch Catalyst serie 2950 con software Cisco IOS versione 12.1(6)EA2 e successive
- Switch Catalyst serie 3550 con software Cisco IOS versione 12.1(4)EA1 e successive

- Switch Catalyst serie 4000 con CatOS 5.1(1a) e versioni successive
- Switch Catalyst serie 4500/4000 con software Cisco IOS versione 12.1(8a)EW e successive
- Switch Catalyst serie 5500/5000 con CatOS versione 4.1(1) e successive
- Switch Catalyst serie 6500/6000 con CatOS versione 5.1(1)CSX e successive
- Switch Catalyst serie 6500/6000 con software Cisco IOS versione 12.0-7XE e successive

BPDU e modalità di confronto

Le BPDU (Bridge Protocol Data Unit) possono essere classificate in base ai campi che contengono. Tra questi campi vi sono l'ID del bridge radice, il costo del percorso alla radice e l'ID del bridge mittente. Una BPDU è considerata migliore di un'altra BPDU per queste ragioni:

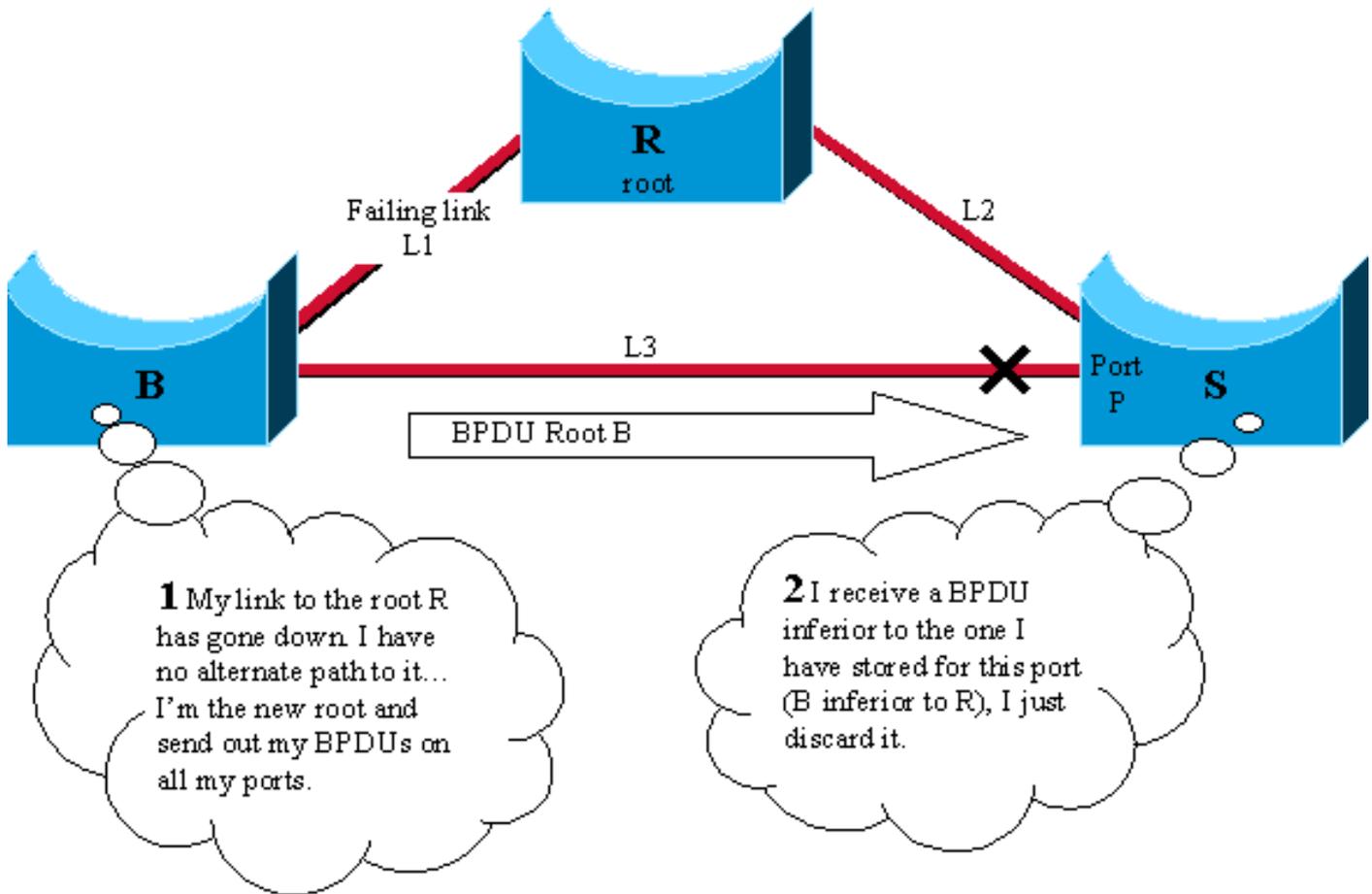
- Quando un BPDU ha un ID di bridge radice migliore di un altro. Più basso è il valore, meglio è.
- Quando i valori ID del bridge radice sono uguali, la BPDU con il costo del percorso più basso per la radice è migliore.
- Quando i valori ID del bridge radice sono uguali e i costi per la radice sono gli stessi, la BPDU con l'ID del bridge mittente migliore è migliore. Più basso è il valore, meglio è.

Ci sono altre variabili che possono agire come tie-breaker. Tuttavia, migliore è una BPDU, migliore è l'accesso al miglior root bridge.

Un bridge che riceve una BPDU su una porta migliore di quella che invia, mette questa porta in modalità di blocco a meno che non sia la porta radice. Ciò significa che sul segmento connesso a questa porta, c'è un altro ponte che è un ponte designato. Un bridge memorizza il valore della BPDU su una porta inviata dal bridge designato corrente.

Ripristino di STP in seguito a un errore di collegamento indiretto

Questo esempio mostra il comportamento di STP quando deve essere ricalcolato dopo un errore di collegamento indiretto, ovvero quando un bridge deve modificare lo stato di alcune delle proprie porte a causa di un errore su un collegamento non direttamente collegato.



Prendiamo in considerazione questo diagramma, che coinvolge tre switch R, B e S in una topologia a rete completa. Si supponga che R sia il bridge radice e B il bridge radice di backup. S blocca la porta P e B è il bridge designato per il collegamento L3.

1. Se il collegamento L1 si interrompe, lo switch B rileva immediatamente l'errore e presume che si tratti della radice. Inizia a inviare BPDU a S e dichiara di essere la nuova radice.
2. Quando S riceve questa nuova BPDU da B, si rende conto di essere inferiore a quella memorizzata per la porta P e la ignora.
3. Alla scadenza del timer di `max_age` (20 secondi per impostazione predefinita), la BPDU memorizzata su S per la porta P scade. La porta entra immediatamente in ascolto e S inizia a inviare la migliore BPDU a B.
4. Non appena B riceve la BPDU da S, interrompe l'invio della BPDU.
5. La porta P passa allo stato di inoltro tramite gli stati di ascolto e apprendimento. Questa operazione richiede il doppio del valore `fw_delay`, ovvero altri 30 secondi. Viene quindi ripristinata la connettività completa.

Il valore di `max_age` (20 secondi) più il doppio del valore di `fw_delay` (2×15 secondi) è stato necessario per risolvere questo errore di collegamento indiretto. In questo modo, si ottengono 50 secondi con i parametri predefiniti. La funzione di accelerazione della backbone propone di salvare `max_age` (20 secondi). A tal fine, si guasta immediatamente dopo che la porta ha ricevuto BPDU inferiori.

Miglioramenti Backbone Fast per STP standard

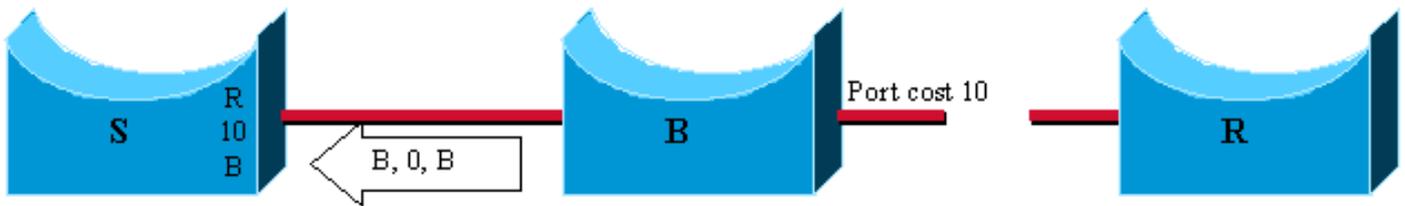
Nell'esempio precedente, il protocollo STP invalida le informazioni che diventano errate a causa di un errore di collegamento indiretto. Per fare questo, attende passivamente `max_age`. Per

eliminare questo ritardo `max_age`, la backbone fast introduce due miglioramenti:

- La capacità di rilevare un errore di collegamento indiretto il prima possibile. Ciò si ottiene tracciando le BPDU inferiori che un bridge designato invia quando rileva un errore di collegamento diretto.
- Meccanismo che consente un controllo immediato se le informazioni BPDU memorizzate su una porta sono ancora valide. Questa funzionalità viene implementata con una nuova PDU (Protocol Data Unit) e con la query di collegamento principale, definita in questo documento PDU RLQ.

Rileva errori di collegamento indiretti

Se si riceve una BPDU inferiore su una porta dal bridge designato, il bridge ha perso la radice e ha iniziato a pubblicizzare una radice con un ID del bridge superiore, una radice peggiore della nostra.



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.

Il comportamento abituale per quanto riguarda le specifiche IEEE (Institute of Electrical and Electronics Engineers) è quello di ignorare semplicemente eventuali BPDU inferiori. La backbone le utilizza rapidamente perché non appena ne viene ricevuta una, è certo che si è verificato un errore sul percorso della radice e che è necessario eseguire il timeout di almeno una porta.

Nota: Un errore di collegamento indiretto può verificarsi senza alcuna generazione BPDU inferiore nella rete. È sufficiente aggiungere un hub nel diagramma precedente:

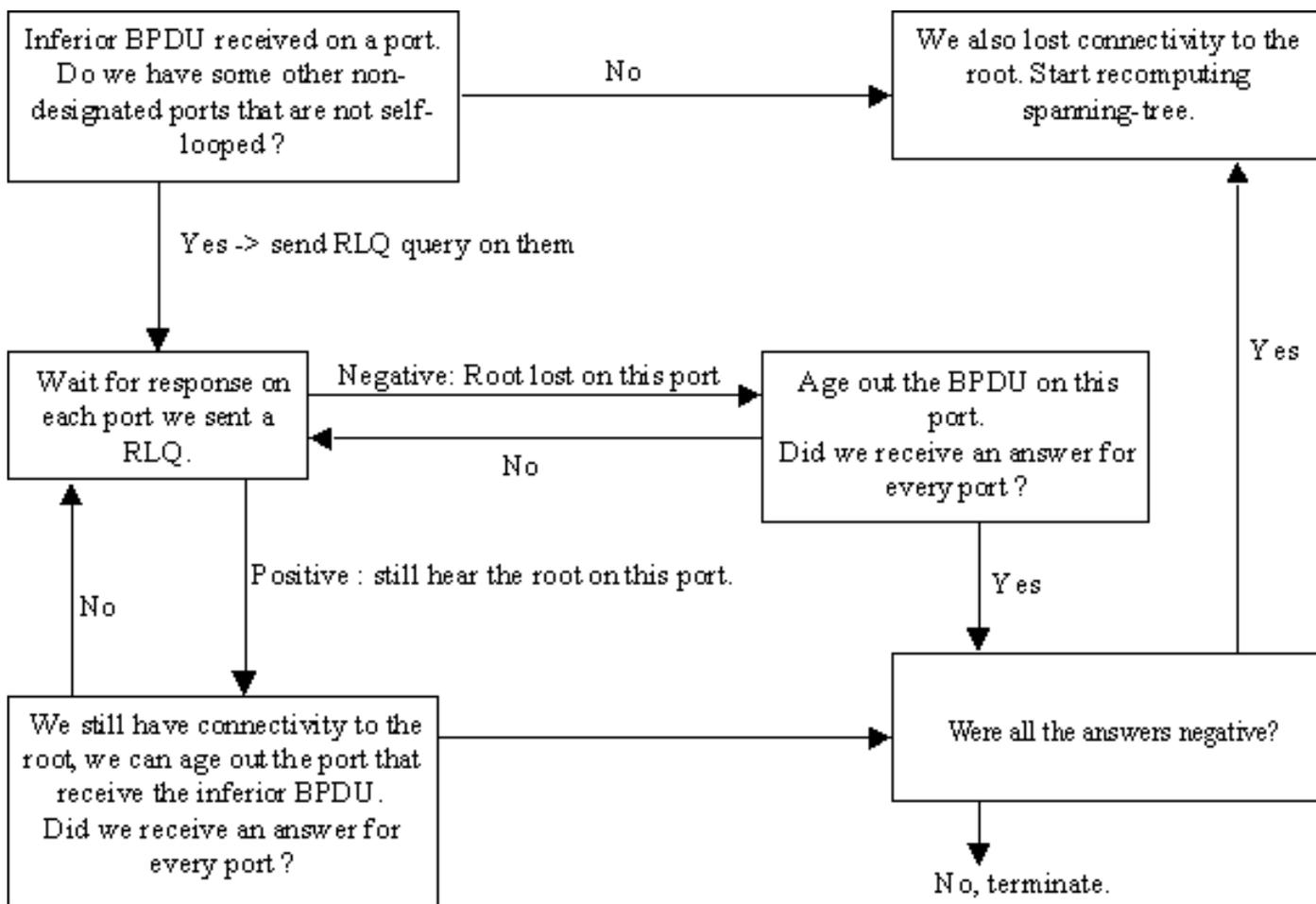


L'errore di collegamento si verifica tra il bridge radice R e l'hub. B non rileva che il collegamento si interrompe e attende `max_age` prima di dichiarare di essere la nuova radice. Tenete presente che il meccanismo funziona solo se un bridge rileva un errore di collegamento diretto.

Tenere traccia solo delle BPDU inferiori inviate dal bridge designato. Poiché si tratta della BPDU che è memorizzata sulla porta. Se, ad esempio, un bridge appena inserito inizia a inviare un pacchetto BPDU inferiore, non avvia la funzione di velocità della backbone.

Reazione a errori di collegamento indiretti

Quando viene rilevata una BPDU inferiore su una porta non designata, viene attivata la seconda fase della velocità della backbone. Anziché attendere passivamente max_age per l'esaurimento delle porte che possono essere interessate dal guasto, viene introdotto un modo proattivo per testarle immediatamente tramite la PDU RLQ. L'RLQ viene usato per ottenere una sorta di ping per la porta principale su una porta non designata e può verificare rapidamente se la BPDU memorizzata su una porta è ancora valida o deve essere eliminata.



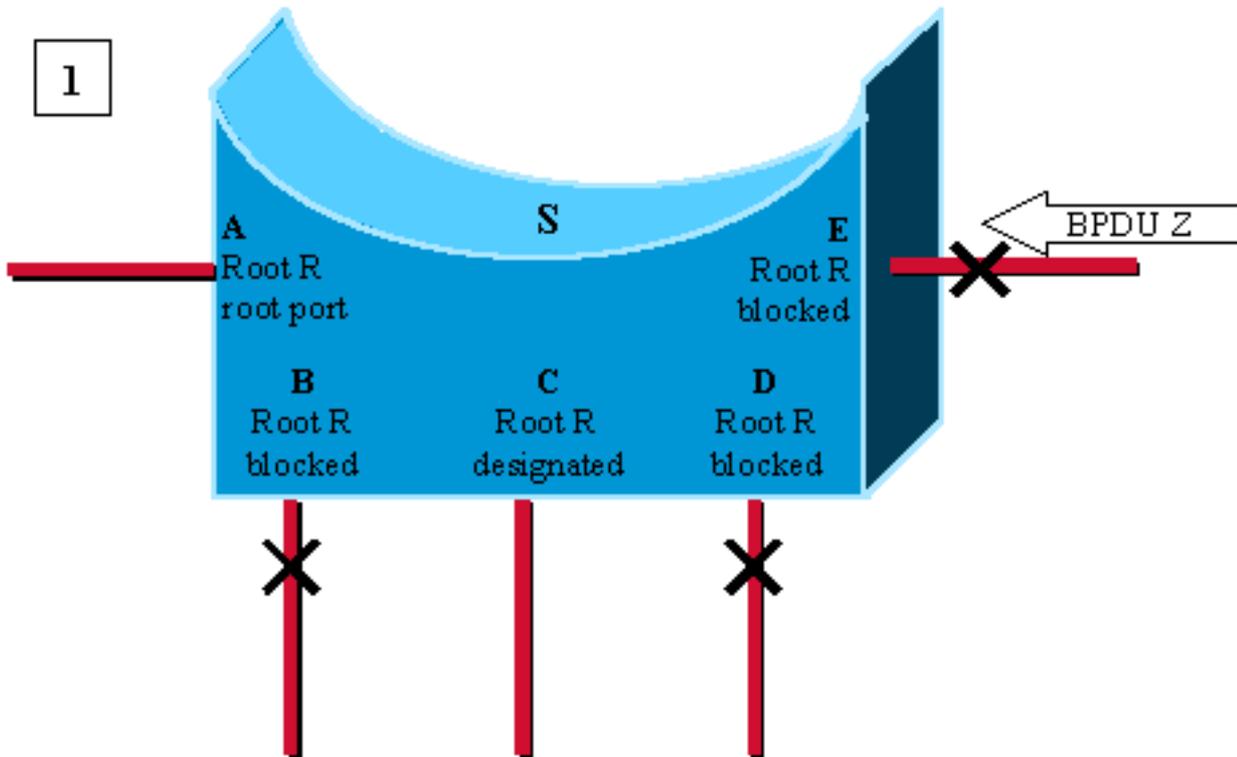
Alla ricezione di una BPDU inferiore da un bridge designato, inviare una PDU RLQ su tutte le porte non designate, ad eccezione della porta su cui si sono ricevute la BPDU inferiore e le porte con loop automatico. In questo modo, è possibile verificare che l'audio venga trasmesso ancora dalla directory principale sulle porte in cui si è abituati a ricevere le BPDU. La porta su cui si è ricevuta la BPDU inferiore viene esclusa in quanto si è già a conoscenza del fatto che si è verificato un errore. Le porte designate e con autolooping non sono utili in quanto non portano alla radice.

Alla ricezione di una risposta RLQ su una porta, se la risposta è negativa, la porta ha perso la connessione alla radice e si può verificare il timeout della BPDU. Inoltre, se tutte le altre porte non designate hanno già ricevuto una risposta negativa, l'intero bridge perde la radice e può iniziare il calcolo STP da zero.

Se la risposta conferma che è ancora possibile accedere al bridge radice tramite questa porta, è possibile escludere immediatamente la porta su cui è stata inizialmente ricevuta la BPDU inferiore.

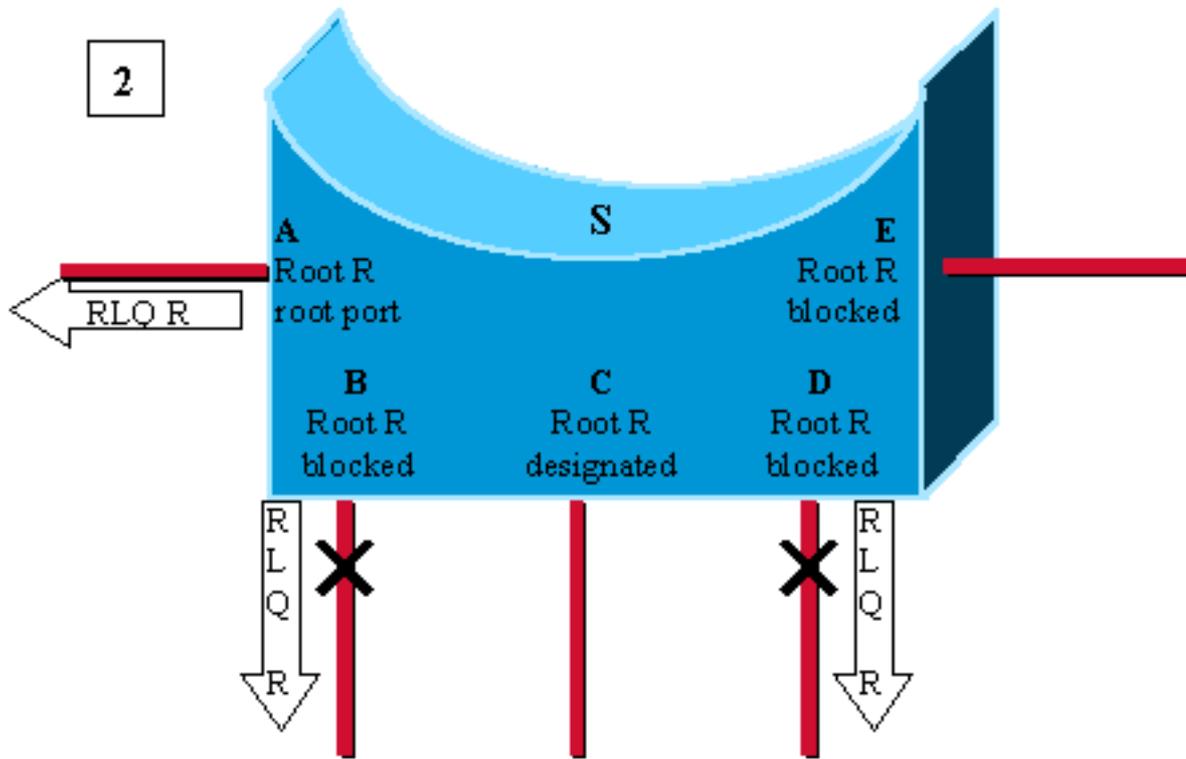
Nell'esempio, le porte A, B, D ed E non sono porte designate per lo switch S. A è la porta radice e le altre stanno bloccando. Quando E riceve una BPDU inferiore (1), la backbone si attiva rapidamente per velocizzare il ricalcolo della STP.

Inviare una richiesta RLQ, che cerca la radice R su tutte le porte non designate tranne E (2). Le risposte specificano quale directory principale è accessibile tramite queste porte. La risposta RLQ ricevuta da D specifica che D ha perso il percorso alla radice R. Età BPDU immediatamente (3). Le porte A e B ricevono conferma che hanno ancora un percorso verso R (4). Pertanto, poiché lo switch S ha ancora connettività con la radice, escludere immediatamente la porta E e continuare con le normali regole STP (5).



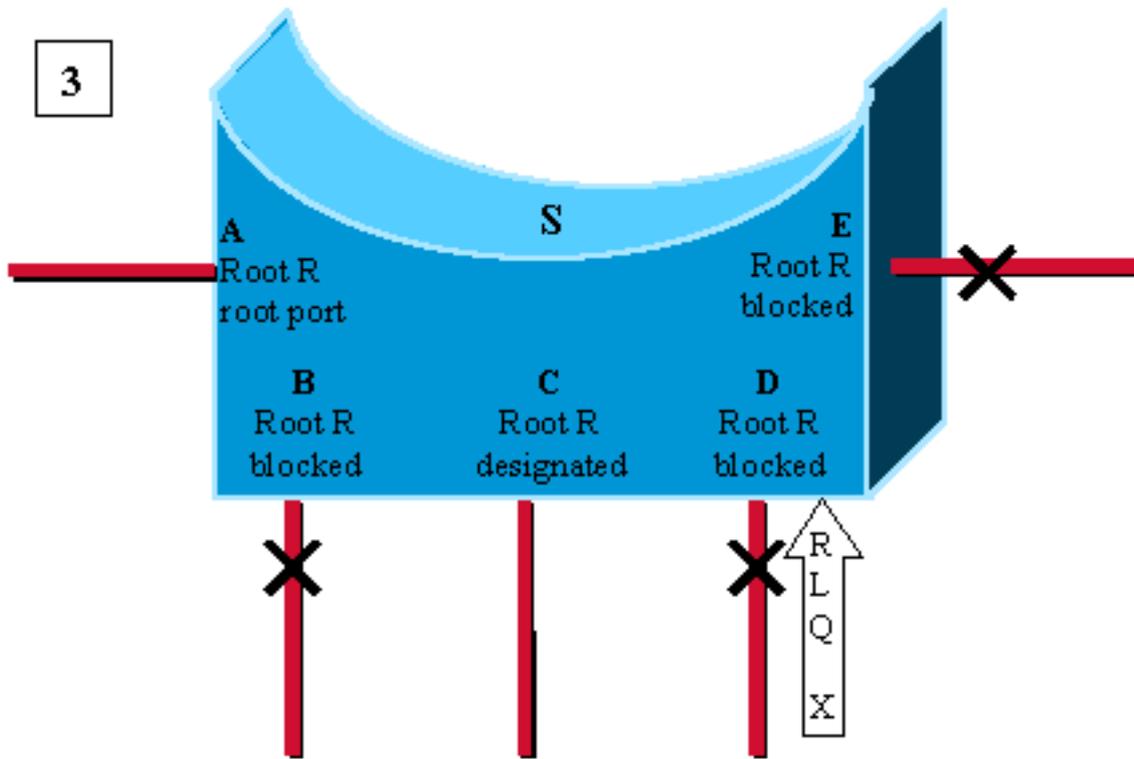
Port E receives an inferior BPDU, advertising root Z instead of root R stored on the different ports.

2



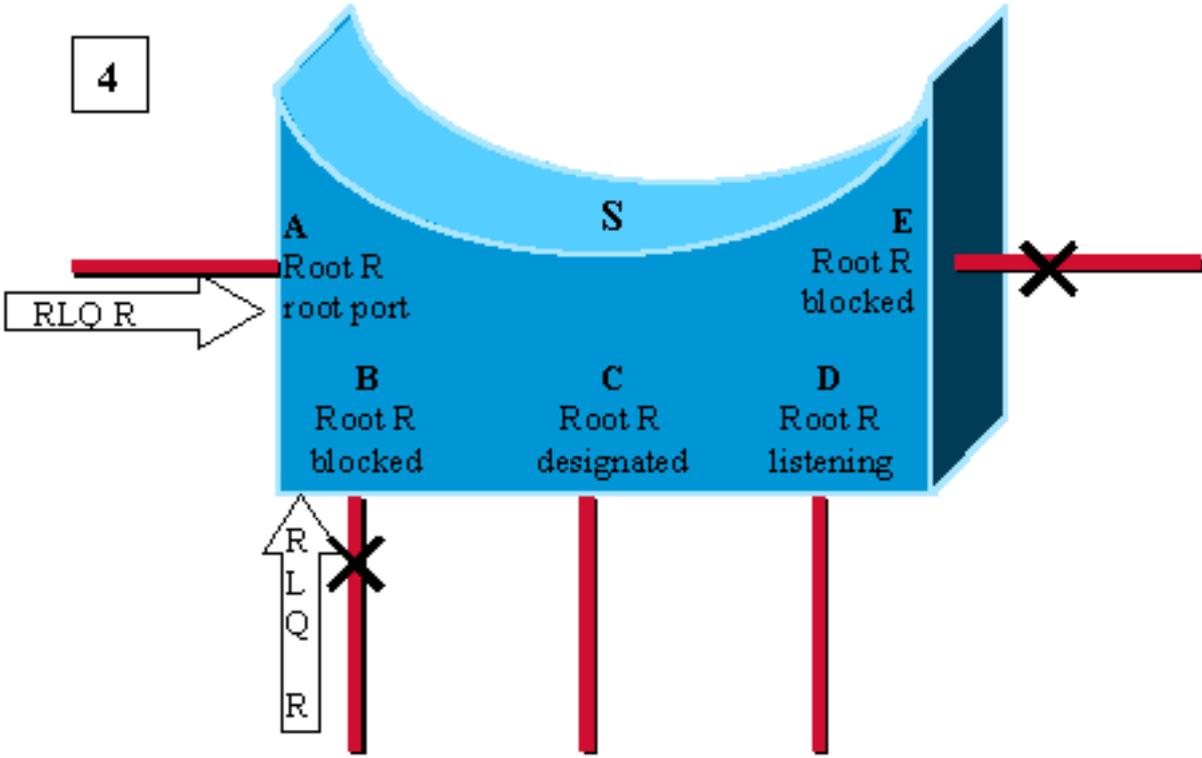
Switch S needs to recheck all its other non-designated ports. It sends out a RLQ request for root R on ports A,B and D.

3



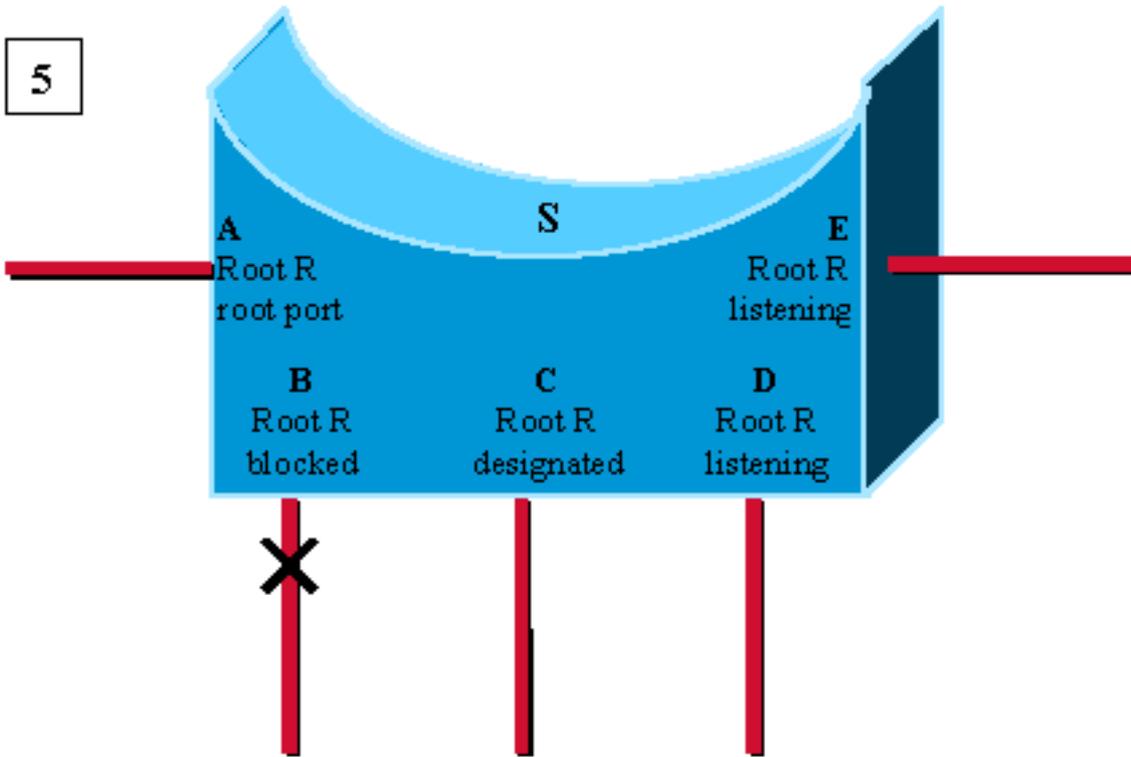
Port D is the first to receive and RLQ response from bridge X claiming to be the root. It is a negative response: D has lost connectivity to the root R. We age out immediately the BPDU on port D and go to listening. As we don't know if we still have connectivity to the root R, we don't age out port E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

Nel caso in cui lo switch abbia ricevuto solo risposte con una radice diversa da R, considerare la radice come persa e riavviare immediatamente il calcolo STP da zero. Notare che questo caso si verifica anche quando l'unica porta non designata (e non con loop automatico) sul bridge è la porta radice e si riceve un pacchetto BPDU inferiore su questa porta.

PDU query collegamento radice

Le due forme di richieste di offerta sono richieste di offerta e risposte.

La richiesta RLQ viene inviata su una porta dove in genere si ricevono i BPDU, per verificare che sia ancora disponibile la connettività alla radice tramite questa porta. Specificare nella richiesta quale bridge è la radice e la risposta RLQ restituirà un bridge radice accessibile tramite questa porta. Se le due radici sono le stesse, la connettività è ancora viva, altrimenti, è persa.

Un bridge che riceve una richiesta RLQ risponde immediatamente se sa di aver perso la connessione con la radice della query perché ha un bridge radice diverso da quello specificato nella query RLQ e se si tratta della radice.

In caso contrario, la query viene inoltrata verso la radice attraverso la relativa porta radice.

Le risposte RLQ vengono inondate nelle porte designate. Il mittente della richiesta RLQ inserisce il proprio ID bridge nella PDU. In questo modo, quando riceve una risposta alla propria richiesta, non inonda la risposta sulle porte designate.

La PDU RLQ ha la stessa struttura di pacchetto di una normale BPDU STP. L'unica differenza consiste nell'utilizzo di due diversi indirizzi SNAP specifici di Cisco: una per la richiesta e una per la risposta.

Questo è il formato BPDU standard:

DA SA Lunghezza DSAP SSAP CNTL AGGANCIO PDU

Il campo PDU è:

Identificatore protocollo	Version	Tipo di messaggio	Flag	ID radice	Costo percorso radice
ID mittente	ID porta	Età messaggi	Età massima	Frequenza di invio dei messaggi hello	Ritardo di inoltro

Anche il tipo di messaggio usato nella PDU è diverso dalla BPDU standard.

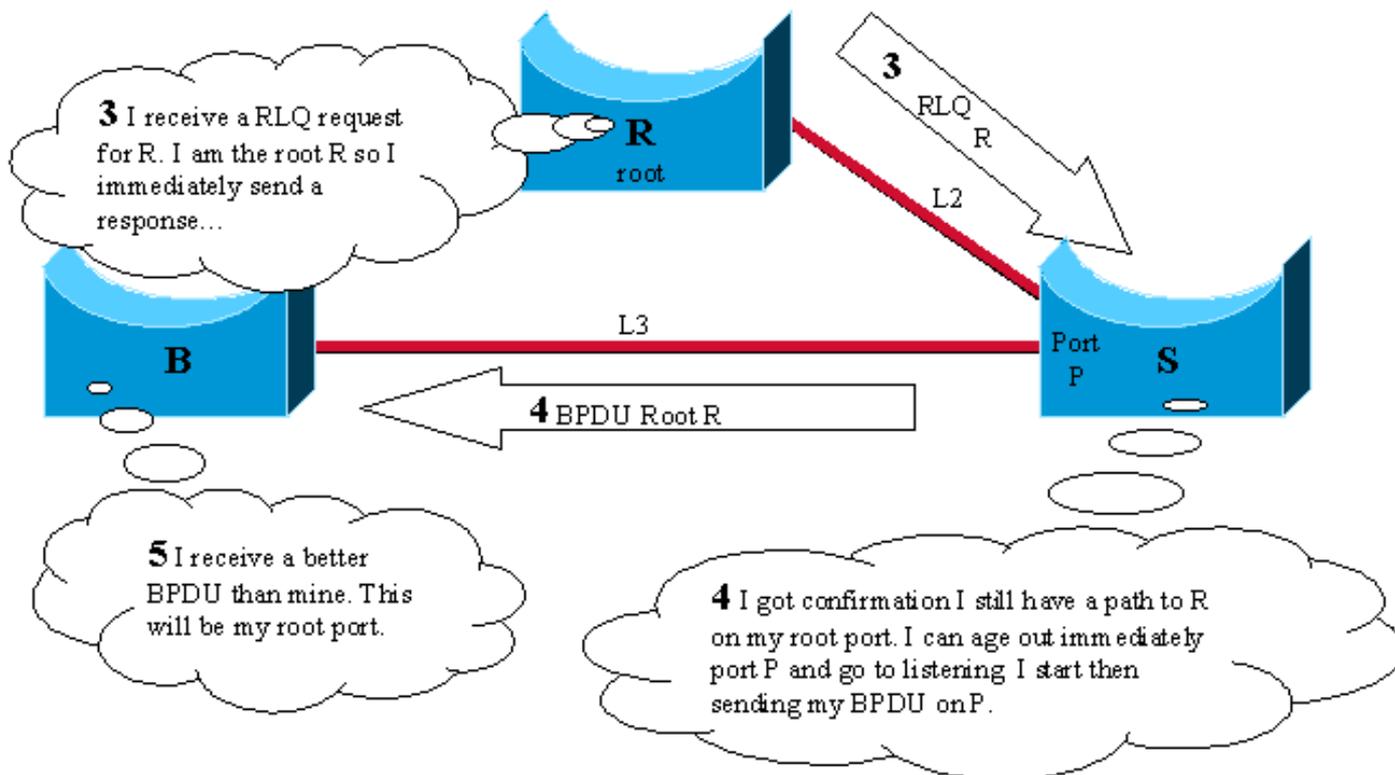
Gli unici campi utilizzati sono l'ID radice e l'ID bridge mittente.

Per elaborare le PDU, questa funzione specifica di Cisco deve essere configurata su tutti gli switch della rete.

Scenario di esempio con la funzione Backbone Fast abilitata

Questo scenario si basa sul primo esempio, ma questa volta con la backbone fast abilitata sui tre switch.

1. La prima fase è esattamente la stessa che è stata spiegata in precedenza.
2. Non appena S riceve la BPDU inferiore da B, inizia a riconfermare le porte non designate anziché attendere max_age. Invia una query RLQ sulla porta radice per il bridge radice R.
3. Il bridge radice R riceve la query e risponde immediatamente con una risposta RLQ che specifica che in quella direzione è ancora presente un R radice.
4. S ha ora controllato tutte le porte non designate e dispone ancora della connettività alla radice. È quindi possibile inviare immediatamente le informazioni memorizzate sulla porta P. P passa all'ascolto e inizia a inviare BPDU. In questa fase, sono già stati salvati i secondi max_age e quindi viene applicato lo Spanning-Tree Algorithm (STA) standard.
5. B riceve la BPDU migliore da S (R migliore della radice di B) e considera ora le porte che conducono a L3 come porta radice.



Configurazione di Backbone Fast per CatOS e Cisco IOS

Se usato, il comando `backbone fast` deve essere abilitato su tutti gli switch della rete perché il comando `backbone fast` richiede l'uso del meccanismo RLQ Request and Reply per informare gli switch della stabilità del percorso radice. Il protocollo RLQ è attivo solo quando la funzione `backbone fast` è abilitata su uno switch. Inoltre, la rete può anche avere problemi con il flooding RLQ, se la funzione `backbone fast` non è abilitata su tutti gli switch. Per impostazione predefinita, l'opzione `backbone fast` è disabilitata.

La funzionalità `Backbone Fast` non è supportata sugli switch Catalyst 2900XL e 3500XL. In generale, è necessario abilitare la `backbone fast` se il dominio dello switch contiene questi switch oltre ad altri switch Catalyst che li supportano. Quando si implementa la `backbone fast` in ambienti con switch XL, in topologie rigide è possibile abilitare la funzione in cui lo switch XL è l'ultimo switch in linea ed è connesso al core solo in due punti. Non implementare questa funzione se l'architettura degli switch XL è concatenata a margherita.

Non è necessario configurare la `backbone fast` con RSTP o IEEE 802.1w perché il meccanismo è incluso in modo nativo e abilitato automaticamente in RSTP. Per ulteriori informazioni su RSTP o IEEE 802.1w, fare riferimento all'[esempio di configurazione dello Spanning Tree da PVST+ a Rapid-PVST Migration](#).

Configurazione per CatOS

Per gli switch Catalyst serie 4000, 5000 e 6000 con CatOS, utilizzare questi comandi per abilitare la `backbone fast` a livello globale per tutte le porte e per verificare la configurazione.

```
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)

Per visualizzare le statistiche sulla velocità della backbone:

```
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)

Configurazione per Cisco IOS

Per gli switch Catalyst con software Cisco IOS, utilizzare questi comandi per abilitare la backbone fast a livello globale per tutte le interfacce.

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

Per verificare che la funzione backbone fast sia abilitata e per visualizzare le statistiche:

```
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#
```

Informazioni correlate

- [Utilizzo di PortFast e di altri comandi per correggere i ritardi di connettività all'avvio della postazione di lavoro](#)
- [Descrizione e configurazione della funzione Cisco UplinkFast](#)
- [Miglioramenti del protocollo Spanning Tree che utilizzano le funzioni Loop Guard e BPDU](#)

Skew Detection

- Miglioramenti della funzionalità Spanning Tree PortFast BPDU Guard
- Esempio di configurazione di Spanning Tree nella migrazione da PVST+ a Rapid-PVST
- Spanning Tree Protocol - Supporto tecnologico
- Switch - Supporto dei prodotti
- Switching per LAN - Supporto tecnologico
- Documentazione e supporto tecnico – Cisco Systems