

Comprenez et configurez Backbone Fast sur des Commutateurs de Catalyst

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[BPDU et comment les comparer](#)

[Comment STP récupère d'une défaillance de liaison indirecte](#)

[Améliorations de Backbone Fast sur STP standard](#)

[Déterminez les défaillances de liaison indirecte](#)

[Réagissez aux défaillances de liaison indirecte](#)

[PDU de requête de liaison racine](#)

[Exemple de scénario avec la fonction Backbone Fast activée](#)

[Configurez Backbone Fast pour CatOS et Cisco IOS](#)

[Configuration pour CatOS](#)

[Configuration pour Cisco IOS](#)

[Informations connexes](#)

Introduction

Ce document décrit comment configurer le circuit principal rapide. Backbone Fast est une fonctionnalité propriétaire de Cisco qui, une fois activée sur tous les commutateurs d'un réseau ponté, peut sauvegarder un commutateur jusqu'à 20 secondes (max_age) quand il récupère d'une défaillance de liaison indirecte. Après un examen rapide des bases du protocole STP (spanning-tree protocol), vous pouvez consulter le scénario de panne précis auquel Backbone Fast s'applique et savoir comment le configurer pour les commutateurs Catalyst qui exécutent les logiciels CatOS et Cisco IOS® à la fois.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

[Composants utilisés](#)

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Commutateurs de la gamme Catalyst 2950 qui exécutent le logiciel Cisco IOS version 12.1(6)EA2 et ultérieure

- Commutateurs de la gamme Catalyst 3550 qui exécutent le logiciel Cisco IOS version 12.1(4)EA1 et ultérieure
- Commutateurs de la gamme Catalyst 4000 qui exécutent CatOS version 5.1(1a) et ultérieure
- Commutateurs de la gamme Catalyst 4500/4000 qui exécutent le logiciel Cisco IOS version 12.1(8a)EW et ultérieure
- Commutateurs de la gamme Catalyst 5500/5000 qui exécutent CatOS version 4.1(1) et ultérieure
- Commutateurs de la gamme Catalyst 6500/6000 qui exécutent CatOS version 5.1(1)CSX et ultérieure
- Commutateurs de la gamme Catalyst 6500/6000 qui exécutent le logiciel Cisco IOS version 12.0-7XE et ultérieure

BPDU et comment les comparer

Les unités BPDU (Bridge Protocol Data Unit) peuvent être strictement classifiées par les champs qu'elles portent. Parmi ces champs il y a l'ID de pont racine, le coût du chemin à la racine et l'ID de pont expéditeur. Un BPDU est considéré préférable à un autre BPDU dans les cas suivants :

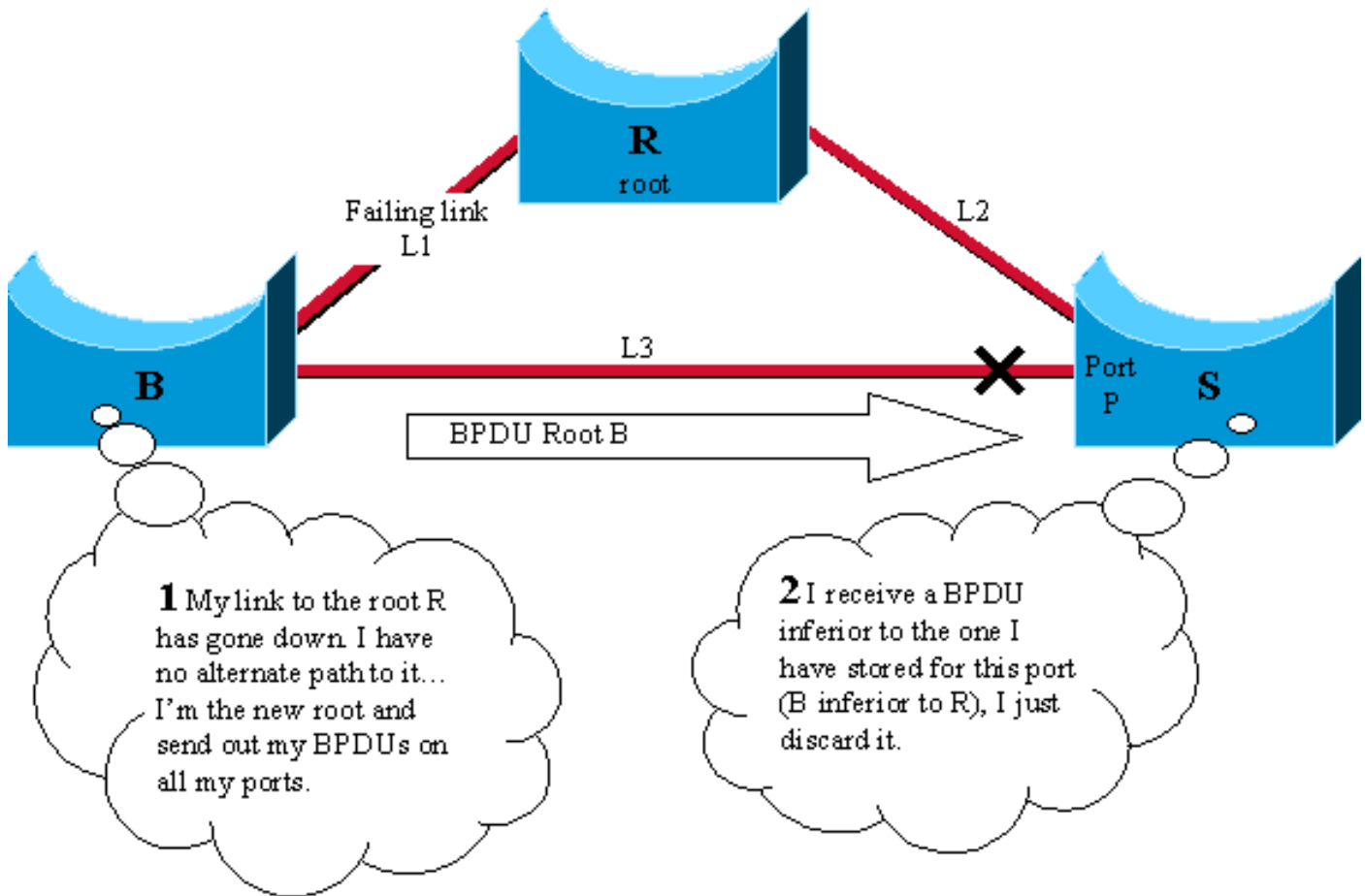
- Un BPDU porte un meilleur ID de pont racine qu'un autre. Plus la valeur est inférieure, mieux c'est.
- Quand les valeurs d'ID de pont racine sont égales, alors le BPDU avec le plus bas coût de chemin à la racine est préférable.
- Quand les valeurs d'ID de pont racine sont égales et les coûts à la racine sont identiques, alors le BPDU avec le meilleur ID de pont expéditeur convient mieux. Plus la valeur est inférieure, mieux c'est.

Il y a d'autres variables qui peuvent faire la différence. Cependant, plus un BPDU est performant, plus l'accès au meilleur pont racine sera performant.

Un pont qui reçoit un BPDU sur un meilleur port que le port d'envoi met ce port en mode de blocage, à moins que ce soit son port racine. Ceci signifie que sur le segment connecté à ce port, il y a un autre pont qui est un pont désigné. Un pont enregistre la valeur du BPDU sur un port envoyé par le pont désigné actuel.

Comment STP récupère d'une défaillance de liaison indirecte

Ceci illustre comment STP se comporte quand il doit procéder à un recalcul après une défaillance de liaison indirecte, c.-à-d., quand un pont doit modifier l'état de certains de ses ports en raison d'une panne sur une liaison qui n'est pas directement attachée.



Considérez ce schéma, qui implique trois commutateurs R, B et S dans une topologie entièrement engrenée. Supposez que R est le pont racine et B est le pont racine de secours. S bloque son port P et B est le pont désigné pour la liaison L3.

1. Si la liaison L1 descend, le commutateur B détecte immédiatement la panne et suppose que c'est la racine. Il commence à envoyer des BPDU à S et prétend être la nouvelle racine.
2. Quand S reçoit ce nouveau BPDU de B, il réalise qu'il est inférieur à celui qu'il avait enregistré pour le port P et l'ignore.
3. Lorsque le compteur de `max_age` expire (20 secondes par défaut), le BPDU enregistré sur S pour le port P expire. Le port passe immédiatement à l'écoute et S commence à envoyer son meilleur BPDU à B.
4. Dès que B recevra le BPDU de S, il cesse d'envoyer son BPDU.
5. Le port P passe à l'état de transmission via des états d'écoute et d'apprentissage. Cela prend deux fois la valeur `fw_delay`, soit 30 secondes supplémentaires. La connectivité totale est alors restaurée.

Elle a pris la valeur `max_age` (20 secondes) plus deux fois la valeur `fw_delay` (2x15 secondes) pour récupérer de cette défaillance de liaison indirecte. Soit, 50 secondes avec les paramètres par défaut. La fonctionnalité Backbone Fast propose de sauvegarder `max_age` (20 secondes). Pour cela, elle expire juste après que le port reçoit des BPDU inférieurs.

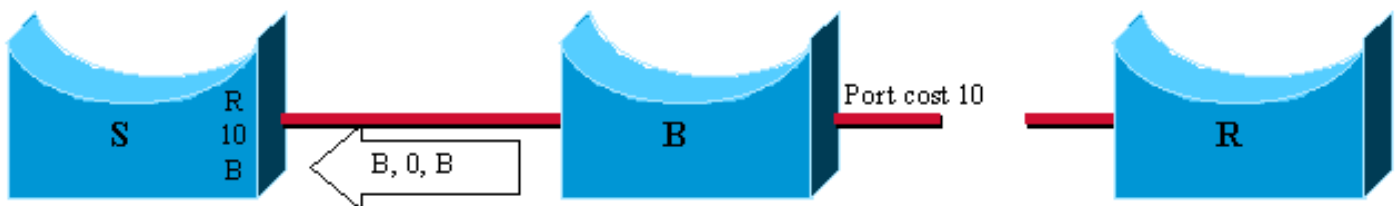
Améliorations de Backbone Fast sur STP standard

Avec l'exemple précédent, STP infirme les informations qui deviennent erronées en raison d'une défaillance de liaison indirecte. Pour cela, il attend passivement le `max_age`. Afin de se débarrasser de ce retard de `max_age`, Backbone Fast introduit deux améliorations :

- La capacité de détecter une défaillance de liaison indirecte dès que possible. Ceci est réalisé en suivant les BPDU inférieurs qu'un pont désigné envoie quand il connaît une panne de lien direct.
- Un mécanisme qui autorise un contrôle immédiat si l'information BPDU enregistrée sur un port est encore valide. Ceci est mis en application avec une nouvelle unité de données de protocole (PDU) et la requête de liaison racine (RQL), mentionnées dans ce document sous le nom de PDU RLQ.

Détectez les défaillances de liaison indirecte

Si un BPDU inférieur est reçu sur un port de notre pont désigné, alors cette passerelle a perdu la racine et les débuts pour annoncer une racine avec un ID plus élevé de passerelle, une plus mauvaise racine que le nôtre.



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.

Le comportement de routage habituel en ce qui concerne les caractéristiques IEEE est d'ignorer simplement tous les BPDU inférieurs. Backbone Fast les utilise parce que dès leur réception, il est certain qu'une panne s'est produite sur le chemin à la racine et que vous devez expirer au moins un port.

Remarque: Une défaillance de liaison indirecte peut se produire sans aucune génération de BPDU inférieur dans le réseau. Ajoutez simplement un concentrateur dans le schéma précédent :



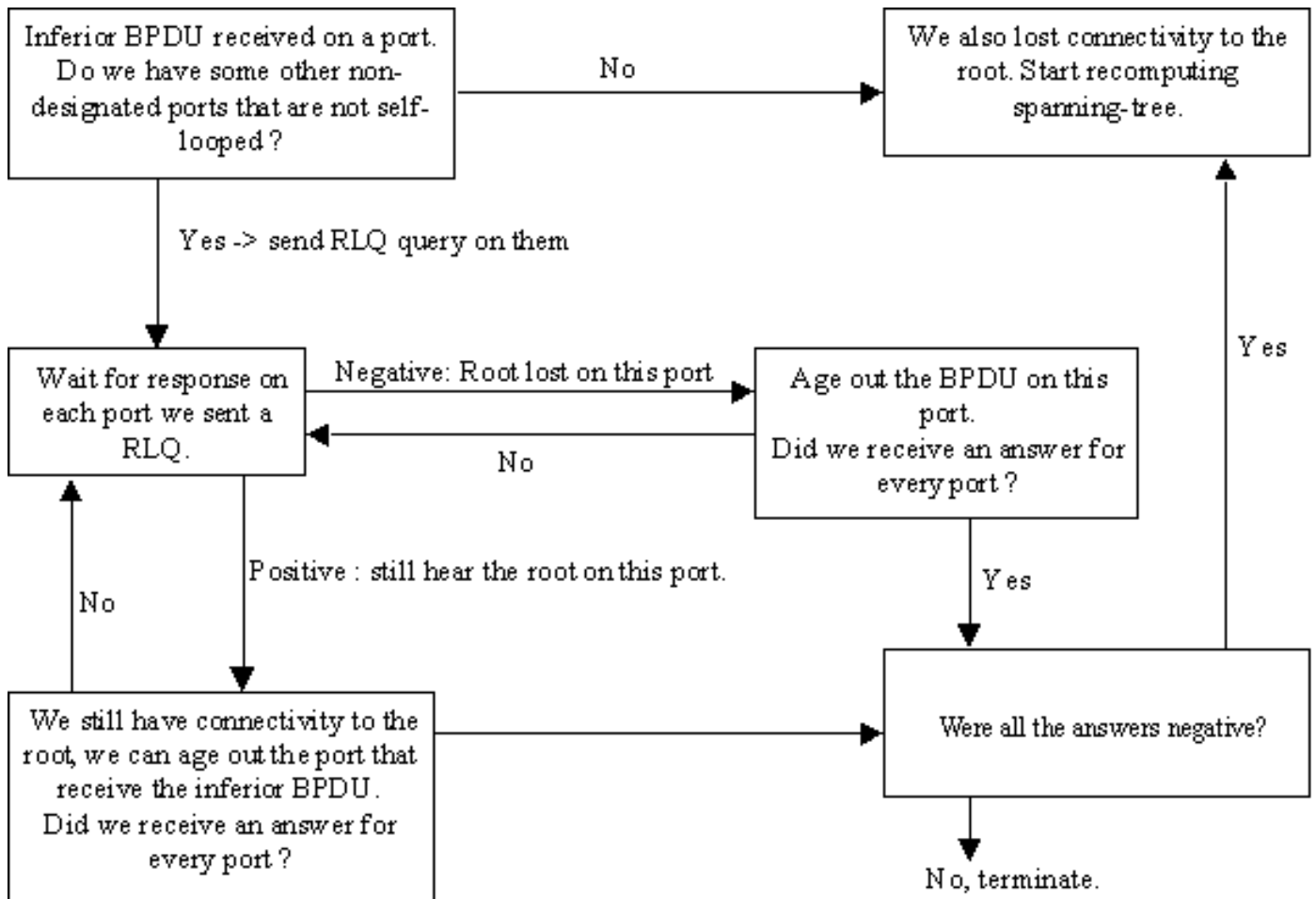
La défaillance de liaison se produit entre le pont racine R et le concentrateur. B ne détecte pas que la liaison descend et attend le max_age avant de prétendre être la nouvelle racine. Souvenez-vous que le mécanisme fonctionne seulement si un pont détecte une panne de lien direct.

Contrôlez seulement les BPDU inférieurs envoyés par le pont désigné. Puisque c'est le BPDU qui est enregistré sur le port. Si, par exemple, un pont qui vient d'être inséré commence à envoyer un BPDU inférieur, il ne démarre pas la fonctionnalité Backbone Fast.

Réagissez aux défaillances de liaison indirecte

Quand un BPDU inférieur est détecté sur un port non désigné, la deuxième étape de Backbone

fast est déclenchée. Au lieu d'attendre passivement le max_age pour expirer les ports qui peuvent être affectés par la panne, une façon proactive de les tester immédiatement est introduite à l'aide de PDU RLQ. Le RLQ est utilisé pour réaliser un genre de ping pour la racine sur un port non désigné et autorisé à confirmer rapidement si le BPDU enregistré sur un port est encore valide ou doit être rejeté.



À la réception d'un BPDU inférieur d'un pont désigné, envoyez un PDU RLQ sur tous les ports non désignés excepté le port où vous avez reçu le BPDU inférieur et les ports en boucle automatique. Cela vise à vérifier que vous entendez toujours la racine sur les ports où vous avez l'habitude de recevoir le BPDU. Le port où vous avez reçu le BPDU inférieur est exclu parce que vous savez déjà compte qu'il a subi une panne, les ports en boucle automatique et désignés ne sont pas utiles, car ils ne mènent pas à la racine.

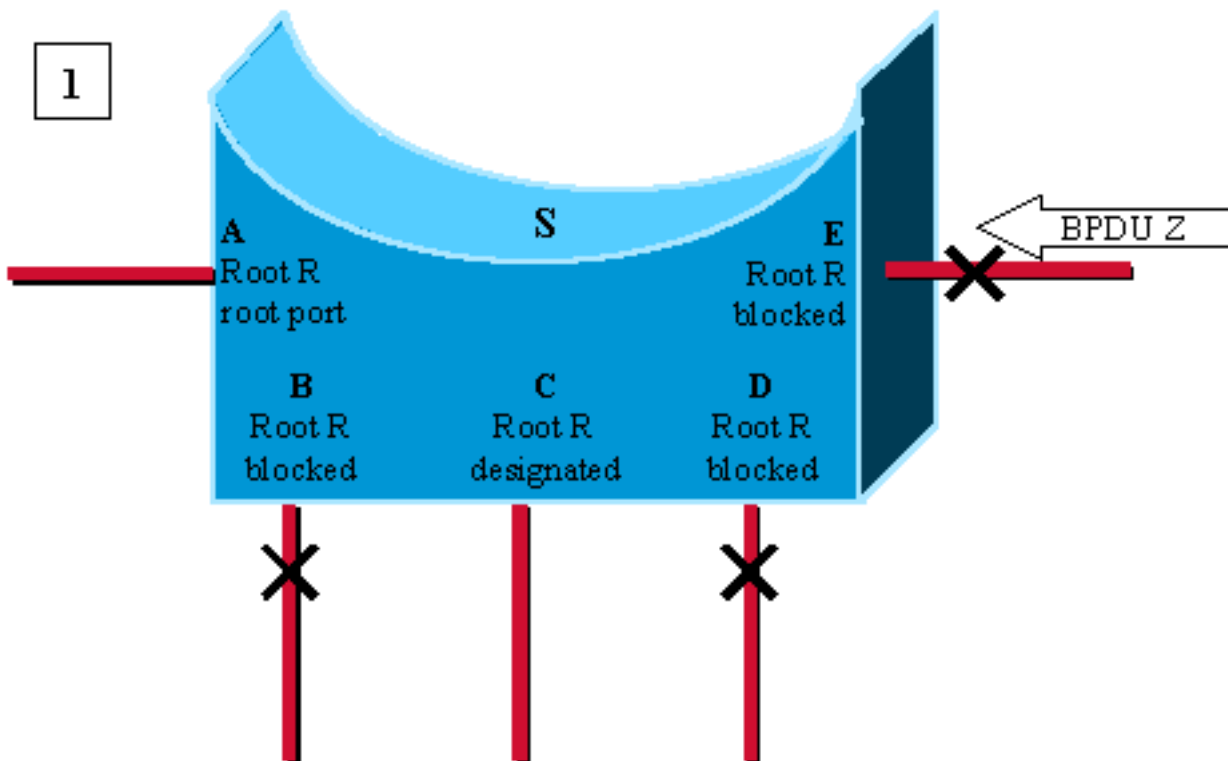
À la réception d'une réponse RLQ sur un port, si la réponse est négative, le port a perdu la connexion à la racine et vous pouvez expirer son BPDU. En outre, si tous les autres ports non désignés recevaient déjà une réponse négative, la totalité du pont perd la racine et peut commencer le calcul de STP à partir de zéro.

Si la réponse le confirme vous pouvez encore accéder au pont racine par l'intermédiaire de ce port, vous pouvez immédiatement expirer le port sur lequel nous avons reçu initialement le BPDU inférieur.

Dans cet exemple, les ports A, B, D et E sont des ports non désignés pour commuter S. A est le port racine et les autres se bloquent. Quand E reçoit un BPDU inférieur (1), Backbone Fast intervient pour accélérer le recalcul de STP.

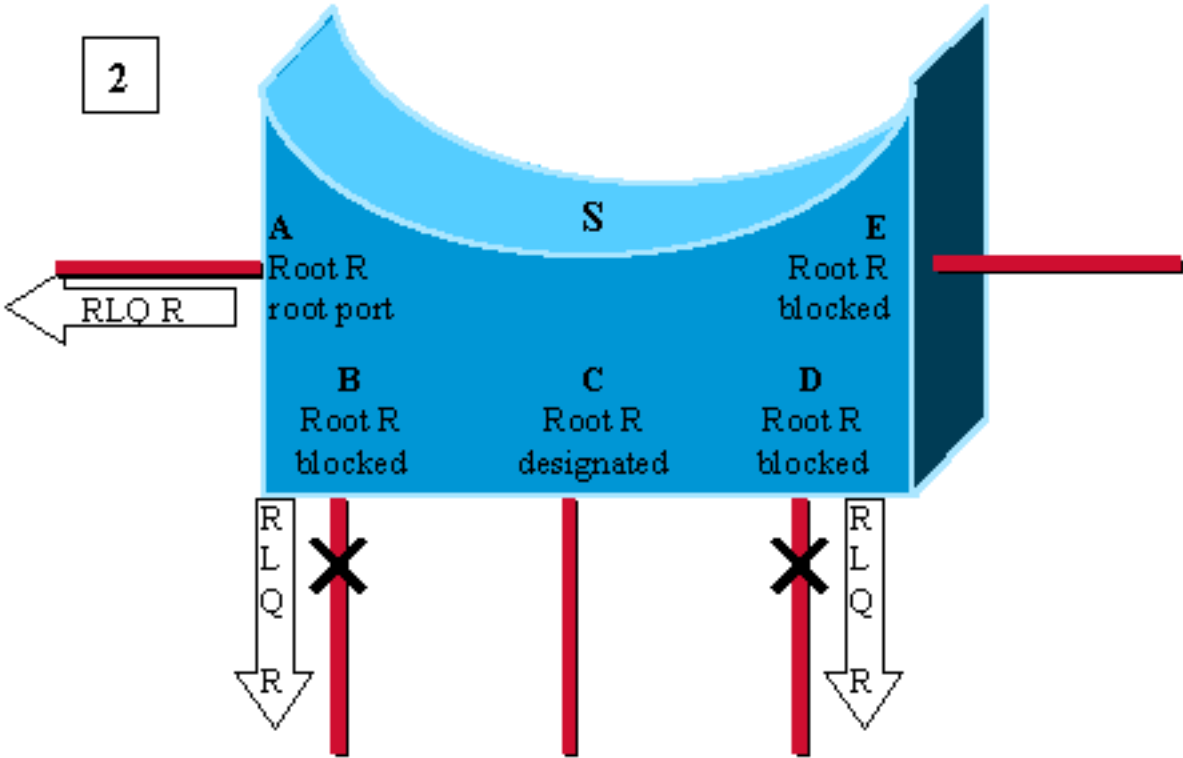
Envoyez une requête RLQ qui recherche la racine R sur tous les ports non désignés sauf E (2).

Les réponses spécifient quelle racine est accessible par l'intermédiaire de ces ports. La réponse RLQ que D reçoit spécifie que ce D a perdu son chemin à la racine R. Expirez immédiatement son BPDU (3). Les ports A et B reçoivent la confirmation qu'ils ont toujours un chemin de routage à R (4). Comme le commutateur S a toujours une connectivité à la racine, expirez immédiatement le port E et continuez avec les règles STP normales (5).



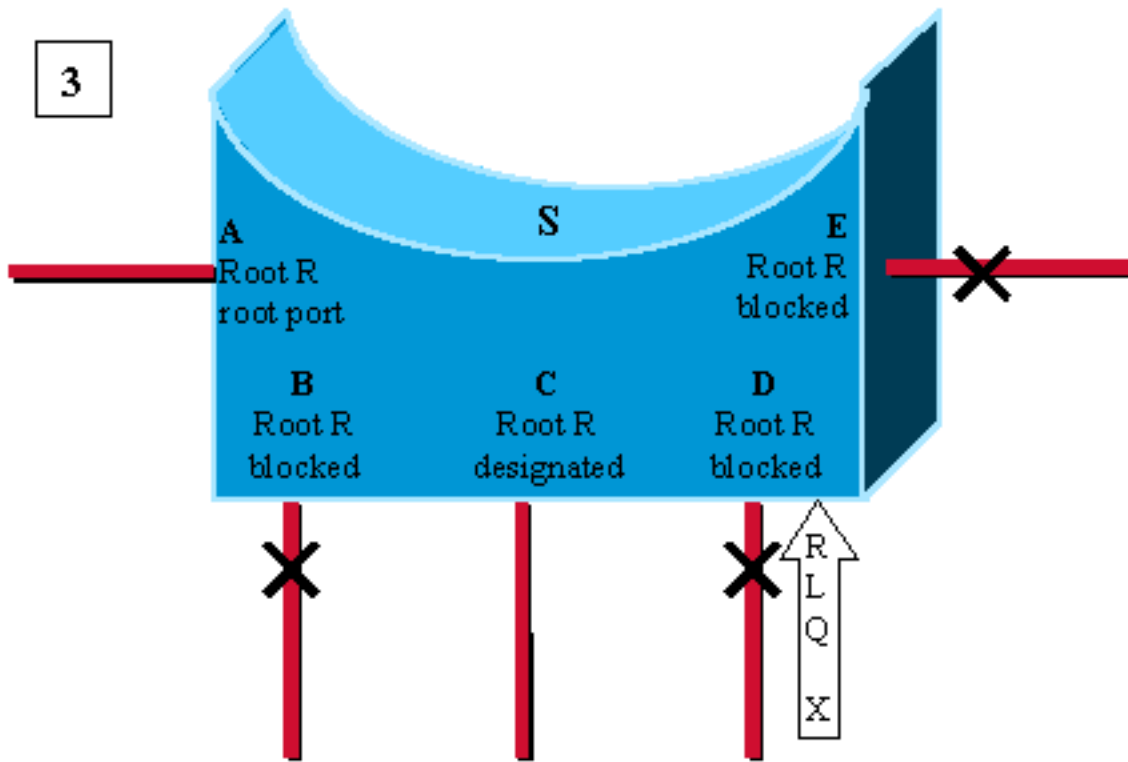
Port E receives an inferior BPDU, advertising root Z instead of root R stored on the different ports.

2



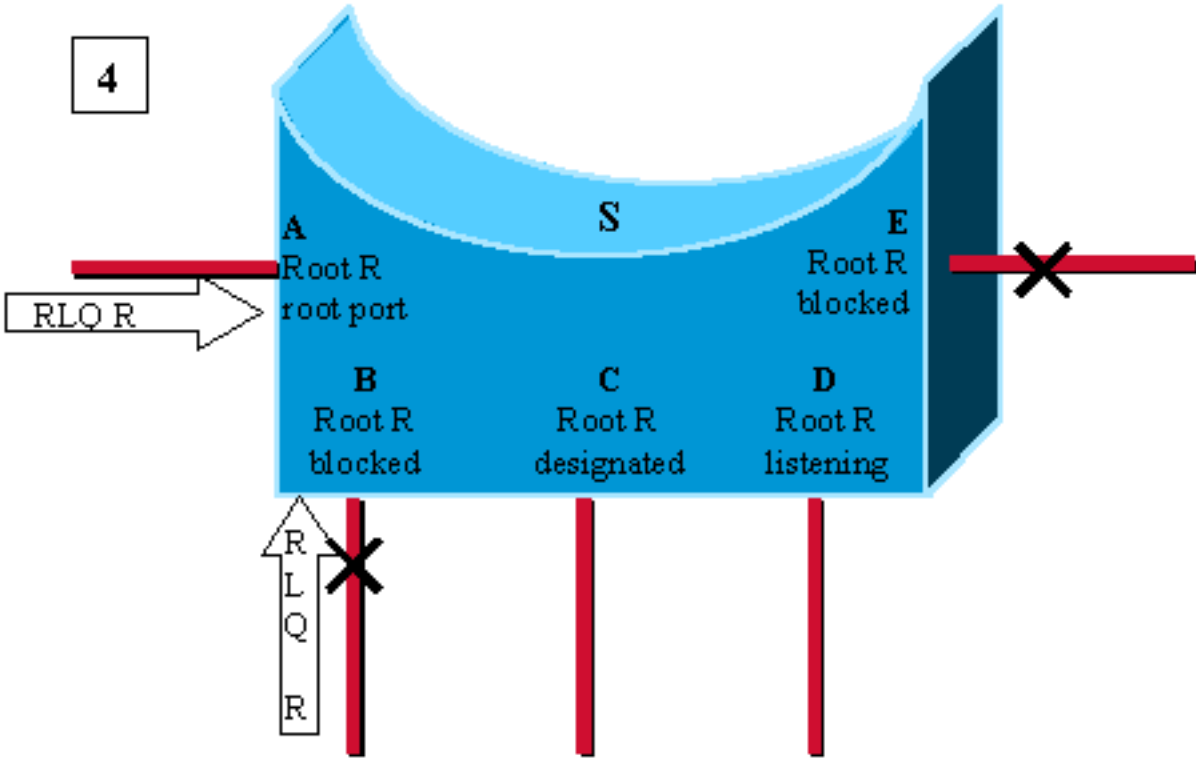
Switch S needs to recheck all its other non-designated ports. It sends out a RLQ request for root R on ports A,B and D.

3



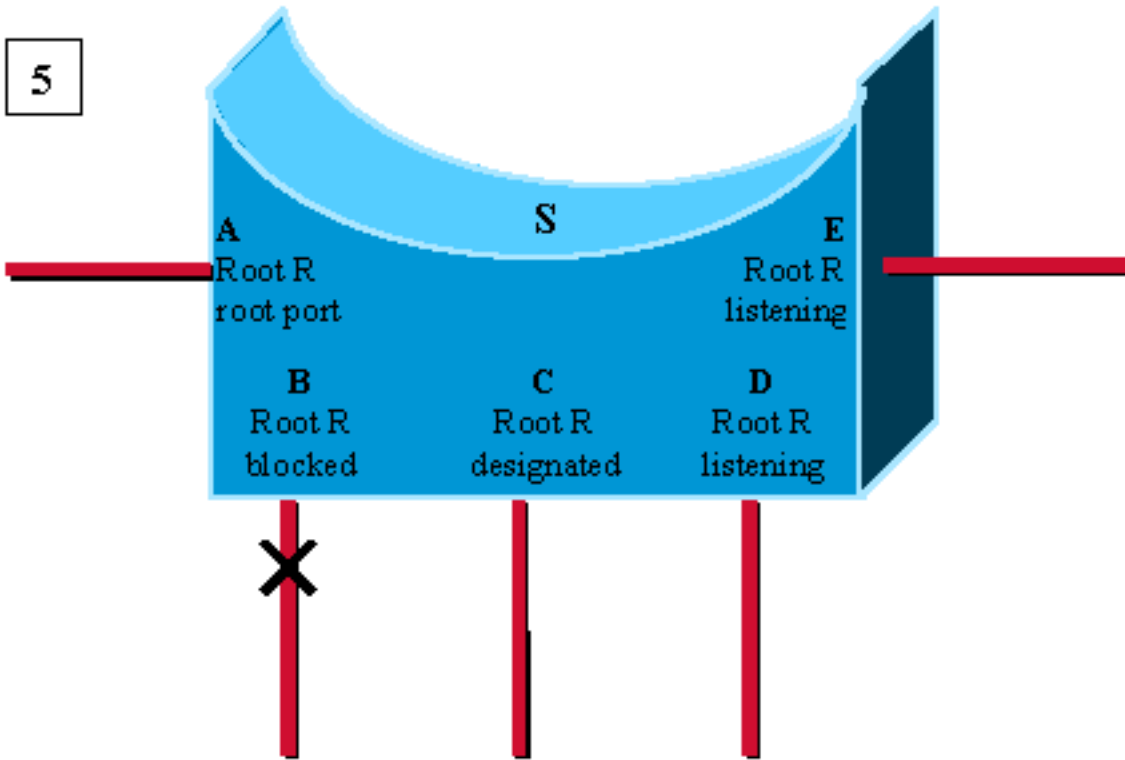
Port D is the first to receive and RLQ response from bridge X claiming to be the root. It is a negative response: D has lost connectivity to the root R. We age out immediately the BPDU on port D and go to listening. As we don't know if we still have connectivity to the root R, we don't age out port E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

Dans le cas où le commutateur a reçu seulement des réponses avec une racine différente de R, considérez la racine comme perdue et redémarrez immédiatement le calcul STP à partir de zéro. Notez que ce cas se produit également quand le seul port non désigné (et non en boucle automatique) sur le pont est le port racine, et vous recevez un BPDU inférieur sur ce port.

PDU de requête de liaison racine

Les deux formes de RLQ sont des requêtes de routage RLQ et des réponses RLQ.

La requête de routage RLQ est envoyée sur un port où vous recevez habituellement le BPDU, afin de vérifier que vous avez toujours la connectivité à la racine par ce port. Spécifiez dans la requête de routage que le pont est votre racine et la réponse RLQ revient éventuellement avec un pont racine qui peut être accédé par ce port. Si les deux racines sont identiques, la connectivité est encore présente, autrement, elle est perdue.

Un pont qui reçoit une requête de routage RLQ répond immédiatement s'il sait qu'il a perdu la connexion à la racine demandée parce qu'il a un pont racine différent de celui spécifié dans la requête RLQ, et si c'est la racine.

Si ce n'est pas le cas, alors, il transfère la requête vers la racine par son port racine.

Les réponses RLQ sont propagées sur les ports désignés. L'expéditeur de la requête de routage RLQ met son ID de pont dans le PDU. Cela afin de s'assurer que quand il reçoit en retour une

réponse à sa propre requête, il ne propage pas la réponse sur ses ports désignés.

Le PDU RLQ a la même structure de paquet qu'un BPDU STP normal. La seule différence est que deux adresses SNAP différentes spécifiques à Cisco sont utilisées : une pour la requête de routage et une pour la réponse.

Voici le format BPDU standard :

LE DA SA Longueur DSAP SSAP CNTL SNAP PDU

Le champ PDU est :

**Identificateur de protocole Version Type de message : Indicateurs ID de racine Coût de chemin racine
ID d'expéditeur ID du port Âge du message Âge maximum Délai Hello Délai de transmission**

Le type de message utilisé dans le PDU est également différent du BPDU standard.

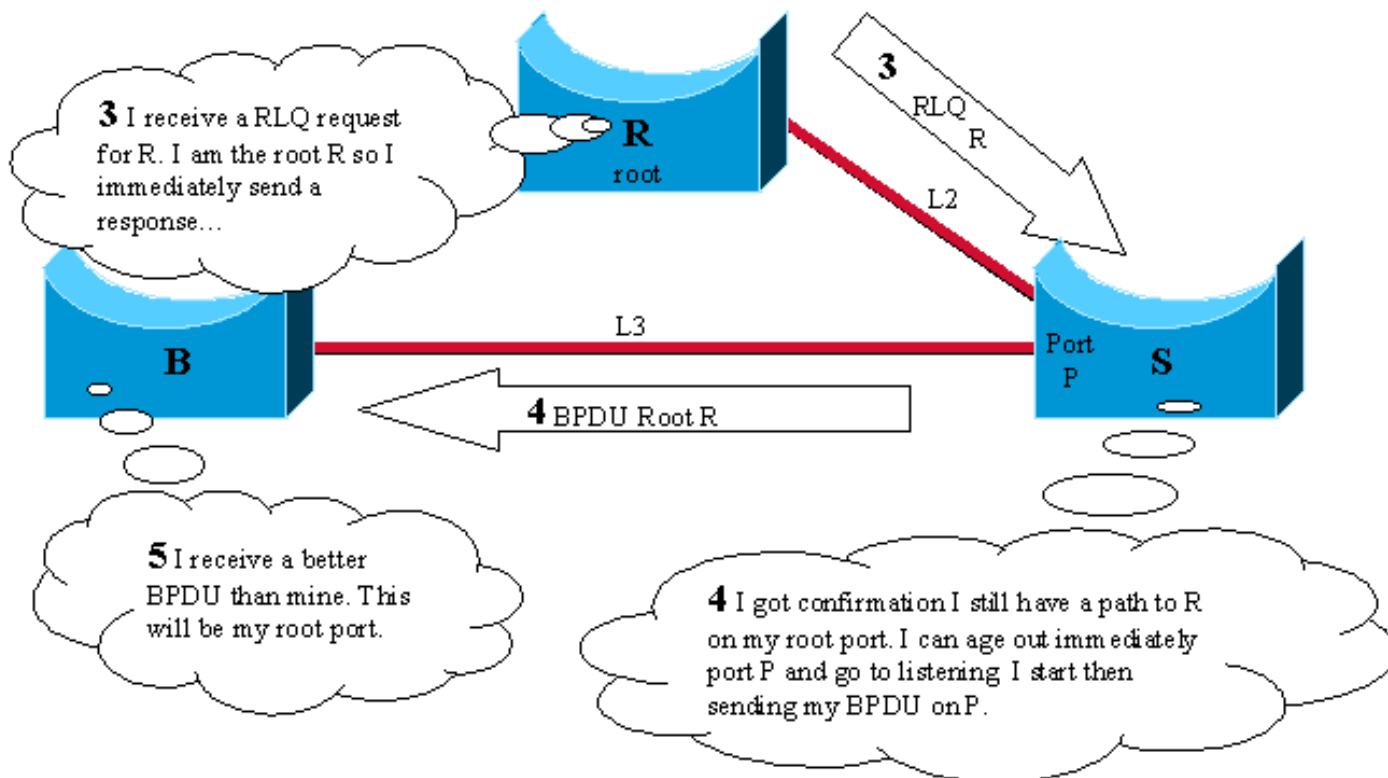
Les seuls champs utilisés sont l'ID de racine et l'ID de pont expéditeur.

Cette fonctionnalité de routage spécifique à Cisco doit être configurée sur tous les commutateurs dans le réseau afin de traiter ces PDU.

[Exemple de scénario avec la fonction Backbone Fast activée](#)

Ce scénario est basé sur le premier exemple, mais, cette fois avec Backbone Fast activé sur les trois commutateurs.

1. La première phase est exactement identique à celle expliquée précédemment.
2. Dès que S reçoit le BPDU inférieur de B, il commence à reconfirmer ses ports non désignés au lieu d'attendre le max_age. Il envoie une requête RLQ sur son port racine pour le pont racine R.
3. Le pont racine R reçoit la requête et répond immédiatement avec une réponse RLQ qui spécifie qu'il y a toujours une racine R dans cette direction.
4. S a maintenant vérifié tous ses ports non désignés, et il a toujours la connectivité à la racine. Il peut alors expirer immédiatement l'information enregistrée sur le port P. P passe à l'écoute et commence à envoyer des BPDU. À cette étape, vous avez déjà économisé des secondes max_age, et l'algorithme Spanning Tree standard (STA) s'applique alors.
5. B reçoit le meilleur BPDU de S (une meilleure racine R que B) et considère maintenant les ports qui mènent à L3 comme son port racine.



Configurez Backbone Fast pour CatOS et Cisco IOS

Une fois utilisé, Backbone Fast doit être activé sur tous les commutateurs dans le réseau parce qu'il exige l'utilisation du mécanisme de requête et de réponse RLQ afin d'informer les commutateurs de la stabilité du chemin racine. Le protocole de routage RLQ est en activité seulement quand Backbone Fast est activé sur un commutateur. En outre, le réseau peut également connaître des problèmes avec la propagation RLQ, si Backbone Fast n'est pas activé sur tous les commutateurs. Par défaut, Backbone Fast est désactivé.

Backbone Fast n'est pas pris en charge sur les commutateurs Catalyst 2900XL et 3500XL. Généralement vous devez activer Backbone Fast si le domaine du commutateur contient ces commutateurs en plus d'autres commutateurs Catalyst le prenant en charge. Quand vous mettez en application Backbone Fast dans des environnements comportant des commutateurs XL et des topologies strictes, vous pouvez activer la fonction à un emplacement au sein duquel le commutateur XL est le dernier commutateur de la ligne et n'est connecté qu'au noyau, à deux endroits. Ne mettez pas en application cette fonction si l'architecture des commutateurs XL est de mode guirlande.

Vous n'avez pas besoin de configurer Backbone Fast avec RSTP ou IEEE 802.1w parce que le mécanisme est inclus de manière native et automatiquement activé dans RSTP. Pour plus d'informations sur RSTP ou IEEE 802.1w, consultez [Exemple de configuration de la migration de Spanning Tree de PVST+ à Rapid-PVST](#).

[Configuration pour CatOS](#)

Pour les commutateurs de la gamme Catalyst 4000, 5000 et 6000 qui exécutent CatOS, employez ces commandes afin d'activer Backbone Fast globalement pour tous les ports et vérifier la configuration.

```
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)
```

Afin d'afficher les statistiques de Backbone Fast :

```
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)
```

[Configuration pour Cisco IOS](#)

Pour les commutateurs Catalyst qui fonctionnent avec le logiciel Cisco IOS, employez ces commandes afin d'activer Backbone Fast globalement pour toutes les interfaces.

```
CAT-IOS# configure terminal
```

```
CAT-IOS(config)# spanning-tree backbonefast
```

```
CAT-IOS(config)# end
```

```
CAT-IOS#
```

Afin de vérifier que Backbone Fast est activé et afficher les statistiques :

```
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#
```

[Informations connexes](#)

- [Utilisation de PortFast et d'autres commandes pour remédier aux délais de connectivité lors du démarrage de la station de travail](#)
- [Présentation et configuration de la fonction Cisco UplinkFast](#)
- [Améliorations du protocole STP \(Spanning Tree Protocol\) avec les fonctions de protection contre les boucles et de détection des différences de temps de propagation des BPDU](#)
- [Amélioration de la protection des BPDU en PortFast pour le spanning tree](#)
- [Exemple de configuration de la migration de Spanning Tree de PVST+ à Rapid-PVST](#)
- [Protocole Spanning Tree - Assistance](#)

- [Commutateurs - Assistance produit](#)
- [Technologie de commutation LAN - Assistance](#)
- [Soutien technique et documentation Cisco Systems](#)