

Comment choisir le meilleur chemin de commutation de routeur pour votre réseau

Contenu

- Introduction
 - Commutation de processus
 - Commutation de contexte d'interruption
 - Commutation rapide
 - Commutation optimale
 - Cisco Express Forwarding**
 - Quel chemin de commutation est le meilleur ?
 - Informations connexes**
-

Introduction

Il y a une pléthore de chemins de commutation disponibles à de divers Routeurs de Cisco et releases de Cisco IOS. Quel est le plus adapté pour votre réseau, et comment fonctionnent-ils? Ce livre blanc tente d'expliquer chacun des trajets de commutation suivants, de sorte que vous puissiez prendre une décision éclairée au sujet du trajet de commutation le plus adapté à votre réseau.

D'abord, examinez le processus de transfert lui-même. Il y a trois étapes à expédier un paquet par un routeur :

1. Déterminez si la destination du paquet est accessible.
2. Déterminez le prochain saut vers la destination, et l'interface par laquelle ce prochain saut est accessible.
3. Réécrivez l'en-tête de Contrôle d'accès au support (MAC) sur le paquet ainsi il atteindra avec succès son prochain saut.

Chacune de ces étapes est essentielle pour que le paquet atteigne sa destination.

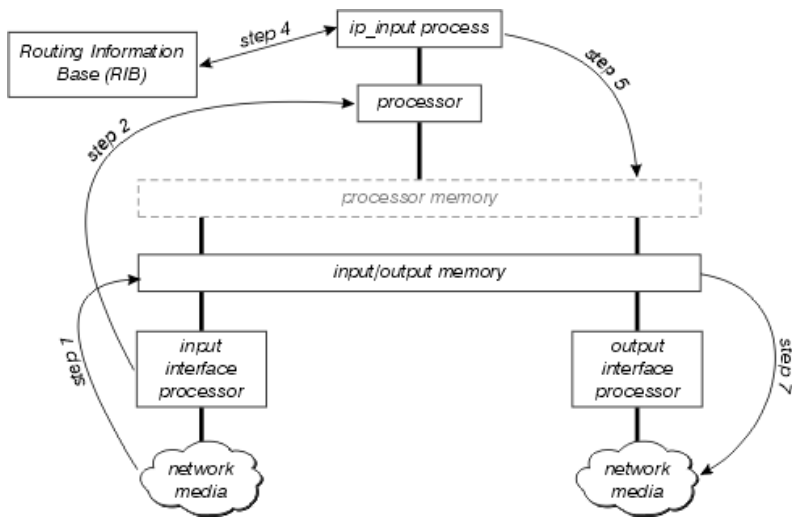
Remarque: Dans tout ce document, le chemin de Commutation IP est utilisé comme exemple ; pratiquement toutes les informations fournies ici s'appliquent aux chemins de commutation équivalents pour d'autres protocoles, s'ils existent.

Commutation de processus

La commutation de processus est le plus bas dénominateur commun dans des chemins de commutation ; il est disponible sur chaque version d'IOS, sur chaque plate-forme, et pour chaque type de trafic étant commuté. La commutation de processus est définie par deux concepts essentiels :

- La décision et les informations d'expédition utilisées pour réécrire l'en-tête MAC sur le paquet sont prises de la table de routage (de la base d'informations de routage, ou de la NERVURE) et du cache de Protocole ARP (Address Resolution Protocol), ou d'une autre table qui contient les informations de l'en-tête MAC tracées à l'adresse IP de chaque hôte qui est directement connecté au routeur.
- Le paquet est commuté par un processus exécuté normal dans l'IOS. En d'autres termes, la décision d'expédition est prise par un processus programmé par le planificateur d'IOS et l'exécution en tant que pair à d'autres processus sur le routeur, tel que des protocoles de routage. Des processus qui fonctionnent normalement sur le routeur ne sont pas interrompus pour traiter le commutateur un paquet.

La figure ci-dessous montre le chemin de commutation de processus.



Examinez ce diagramme plus en détail :

1. Le processeur d'interface le détecte d'abord il y a un paquet sur les supports réseau, et transfère ce paquet vers la mémoire d'entrée/sortie sur le routeur.
2. Le processeur d'interface génère une interruption de réception. Pendant cette interruption, le processeur central détermine quel type de paquet c'est (supposez que c'est un paquet IP), et le copie dans la mémoire du processeur s'il y a lieu (cette décision est personne à charge de plate-forme). En conclusion, le processeur place le paquet sur la file d'attente d'entrée des processus appropriés et l'interruption est libérée.
3. La prochaine fois que le programmeur fonctionne, il note le paquet dans la file d'attente d'entrée de l' **ip_input**, et programme ce processus pour fonctionner.
4. Quand l'**ip_input** fonctionne, il consulte la NERVURE pour déterminer le prochain saut et l'interface de sortie, alors consulte le cache d'ARP pour déterminer l'adresse correcte de couche physique pour ce prochain saut.
5. L'**ip_input** réécrit alors l'en-tête MAC du paquet, et place le paquet sur la file d'attente de sortie de l'interface sortante correcte.
6. Le paquet est copié de la file d'attente de sortie de l'interface sortante sur la file d'attente de transmission de l'interface sortante ; n'importe quelle qualité de service sortante a lieu entre ces deux files d'attente.
7. Le processeur d'interface de sortie détecte le paquet sur sa file d'attente de transmission, et transfère le paquet sur les supports réseau.

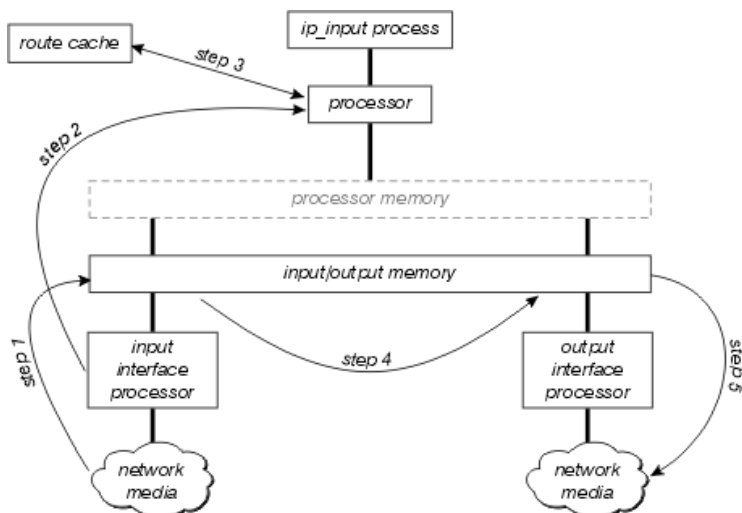
Presque toutes les caractéristiques qui effectuent la commutation de paquets, telle que le Traduction d'adresses de réseau (NAT) et le routage de stratégie, font leur début dans le chemin de commutation de processus. Une fois qu'ils ont été prouvés, et optimisés, ces caractéristiques pourraient, ou ne pourraient pas, apparaître dans la commutation de contexte d'interruption.

Commutation de contexte d'interruption

La commutation de contexte d'interruption est la deuxième des méthodes de commutation primaires utilisées par des Routeurs de Cisco. Les différences principales entre la commutation de contexte d'interruption et le changement de processus sont :

- Le processus s'exécutant actuellement sur le processeur est interrompu pour commuter le paquet. Les paquets sont à la demande commuté, plutôt que commutés seulement quand le processus d' **ip_input** peut être programmé.
- Le processeur emploie une certaine forme de cache d'artère pour trouver toutes les informations requises pour commuter le paquet.

Cette figure montre la commutation de contexte d'interruption :



Examinez ce diagramme plus en détail :

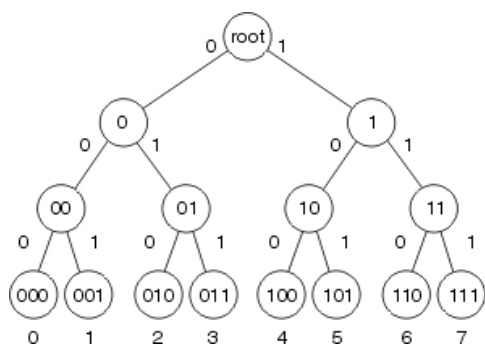
1. Le processeur d'interface le détecte d'abord il y a un paquet sur les supports réseau, et transfère ce paquet vers la mémoire d'entrée/sortie sur le routeur.
2. Le processeur d'interface génère une interruption de réception. Pendant cette interruption, le processeur central détermine quel type de paquet c'est (supposez que c'est un paquet IP), et puis commence à commuter le paquet.
3. Le processeur recherche le cache d'artère pour déterminer si la destination du paquet est accessible, ce qui être l'interface de sortie devrait, ce qui le prochain saut vers cette destination est, et en conclusion, quelle en-tête MAC le paquet si devez avec succès atteindre le prochain saut. Le processeur emploie ces informations pour réécrire l'en-tête MAC du paquet.
4. Le paquet est maintenant copié sur la transmission ou la file d'attente de sortie de l'interface sortante (selon de divers facteurs). L'interruption de réception retourne maintenant, et le processus qui s'exécutait sur le processeur avant que l'interruption se soit produite continue de s'exécuter.
5. Le processeur d'interface de sortie détecte le paquet sur sa file d'attente de transmission, et transfère le paquet sur les supports réseau.

La première question qui vient à l'esprit après lecture de cette description est « ce qui est dans le cache ? » Il y a trois réponses possibles, selon le type de commutation de contexte d'interruption :

- Commutation rapide
- Commutation optimale
- **Cisco Express Forwarding**

Commutation rapide

La commutation rapide enregistre les informations et la chaîne de réécriture de l'en-tête MAC d'expédition utilisant une arborescence binaire pour la consultation et la référence rapides. Cette figure montre une arborescence binaire :



Dans la commutation rapide, les informations d'accessibilité sont indiquées par l'existence d'un nœud sur l'arborescence binaire pour la destination du paquet. L'en-tête MAC et l'interface sortante pour chaque destination sont enregistrées en tant qu'élément des informations du nœud dans l'arborescence. L'arborescence binaire peut réellement avoir 32 niveaux (l'arborescence ci-dessus est extrêmement abrégée afin de l'illustration).

Afin de rechercher une arborescence binaire, vous commencez simplement à partir du gauche (avec le chiffre le plus significatif) dans le nombre (de binaire) que vous recherchez, et s'embrancher gauche ou droite dans l'arborescence basée sur ce nombre. Par exemple, si vous recherchez le relatif à l'information au numéro 4 dans cette arborescence, vous commenceriez par droite de embranchement, parce que le premier chiffre binaire est 1. Vous suivriez l'arborescence vers le bas, comparant le prochain chiffre dans le nombre (de binaire), jusqu'à ce que vous atteigniez l'extrémité.

Caractéristiques de la commutation rapide

La commutation rapide a plusieurs caractéristiques qui sont un résultat de la structure arborescente binaire et la mémoire des informations de réécriture de l'en-tête MAC en tant qu'élément des Nœuds d'arborescence.

- Puisqu'il n'y a aucune corrélation entre la table de routage et le contenu de cache rapide (réécriture de l'en-tête MAC, par exemple), le cache entries de bâtiment comporte tout le traitement qui doit être fait dans le chemin de commutation de processus. Par conséquent, des entrées en cache rapide sont établies car les paquets sont commutés par processus.
- Puisqu'il n'y a aucune corrélation entre les en-têtes MAC (utilisées pour des réécritures) dans le cache d'ARP et la structure du cache rapide, quand la table ARP change, une certaine partie du cache rapide doit être infirmée (et recréé par la commutation de processus des paquets).
- Le cache rapide peut seulement établir des entrées à une profondeur (une longueur de préfixe) pour n'importe quelle destination particulière dans la table de routage.
- Il n'y a aucune manière d'indiquer d'une entrée des autres dans le cache rapide (on s'attend à ce que l'en-tête MAC et les informations sur l'interface de sortie soient dans le nœud), ainsi toutes les récursions de routage doivent être résolues tandis qu'une entrée en cache rapide est établie. En d'autres termes, des artères récursives ne peuvent pas être résolues dans le cache rapide lui-même.

Entrées vieillissantes de commutation rapide

Afin de garder les entrées de commutation rapide de perdre leur synchronisation avec la table de routage et le cache d'ARP, et maintenir les entrées inutilisées dans le cache rapide de la mémoire à l'excès consommante sur le routeur, la 1/20th du cache rapide est infirmée, aléatoirement, chaque minute. Si la mémoire de Routeurs chute au-dessous d'un filigrane très bas, le 1/5th des entrées en cache rapide sont infirmés chaque minute.

Longueur de préfixe de commutation rapide

Pour quelle longueur de préfixe la commutation rapide établit-elle des entrées si elle peut seulement construire à une longueur de préfixe pour chaque destination ? Dans les termes de la commutation rapide, une destination est une destination accessible simple dans la table de routage, ou un réseau important. Les règles pour décider quelle longueur de préfixe pour établir une entrée donnée de cache soiez :

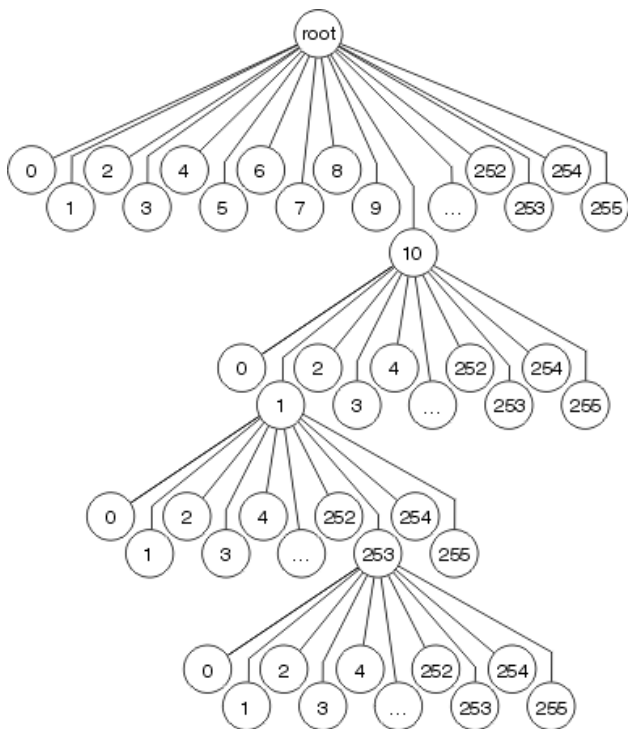
- Si établissant une entrée de stratégie rapide, cachez toujours à /32.
- Si établissant une entrée contre un circuit virtuel ATM fini multiprotocole (circuit virtuel MPOA), cachez toujours à /32.
- Si le réseau pas subnetted (c'est une entrée de réseau important) :
 - S'il est directement connecté, l'utilisation /32 ;
 - Autrement utilisez le masque de réseau principal.
- S'il est un super-réseau utilise le masque des supernet.
- Si le réseau subnetted :
 - Si directement connecté, utilisation /32 ;
 - S'il y a des plusieurs chemins à ce sous-réseau, l'utilisation /32 ;
 - Dans tous les autres cas, utilisez la plus longue longueur de préfixe en ce réseau principal.

Partager de chargement

La commutation rapide est entièrement destination basée ; partager de chargement se produit sur une base de par-destination. S'il y a des plusieurs chemins de coût égal pour un réseau de destination particulière, le cache rapide a une entrée pour chaque hôte accessible dans ce réseau, mais tout trafic destiné à un hôte spécifique suit un lien.

Commutation optimale

La commutation optimale stocke les informations d'expédition et les informations de réécriture de l'en-tête MAC dans une arborescence multiway de 256 manières (mtree de 256 manières). Utilisant un mtree réduit le nombre d'étapes qui doivent être prises en recherchant un préfixe, comme illustré dans la prochaine figure.



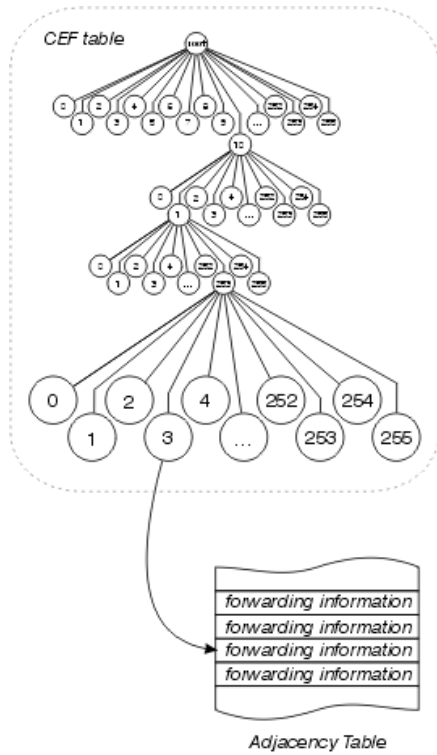
Chaque octet est utilisé pour déterminer lesquels des 256 branchements à prendre à chaque niveau de l'arborescence, qui signifie il y a, tout au plus, 4 consultations impliquées en trouvant n'importe quelle destination. Pour des longueurs de préfixe plus courtes, seulement des consultations d'one-three peuvent être exigées. Les informations d'interface de réécriture de l'en-tête MAC et de sortie sont stockées en tant qu'élément du nœud d'arborescence, ainsi cachez l'invalidation et le vieillissement se produisent toujours comme dans la commutation rapide.

Le changement optimal détermine également la longueur de préfixe pour chaque entrée de cache de la même manière que la commutation rapide.

Cisco Express Forwarding

Cisco Express Forwarding, emploie également une structure de 256 données de manière pour stocker l'expédition et les informations de réécriture de l'en-tête MAC, mais il n'utilise pas une arborescence. Cisco Express Forwarding utilise un trie, qui signifie que les informations réelles étant recherchées ne sont pas dans la structure de données ; au lieu de cela, les données sont enregistrées dans une structure de données séparée, et le trie indique simplement lui. En d'autres termes, plutôt qu'enregistrant l'interface et la réécriture de l'en-tête MAC sortantes dans

l'arborescence elle-même, Cisco Express Forwarding stocke ces informations dans une structure de données séparée appelée la table de juxtaposition.



Cette séparation des informations d'accessibilité (dans la table de Cisco Express Forwarding) et des informations d'expédition (dans la table de juxtaposition), fournit un certain nombre d'indemnités :

- La table de juxtaposition peut être construite séparément de la table de Cisco Express Forwarding, laissant chacun des deux pour construire sans processus commutant tous les paquets.
- La réécriture de l'en-tête MAC utilisée pour expédier un paquet n'est pas enregistrée dans le cache entries, change ainsi dans une chaîne de réécriture de l'en-tête MAC n'exigent pas l'invalidation du cache entries.
- Vous pouvez indiquer directement les informations d'expédition, plutôt qu'au prochain saut recursed, afin de résoudre les artères récursives.

Essentiellement, tout le vieillissement de cache est éliminé, et le cache est prêt à l'emploi basé sur les informations contenues dans la table de routage et le cache d'ARP. Il n'y a aucun besoin de traiter le commutateur n'importe quel paquet pour établir une entrée de cache.

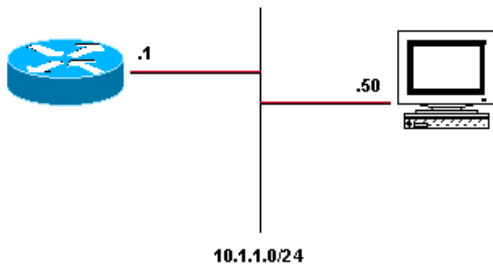
D'autres entrées dans la table de juxtaposition

La table de juxtaposition peut contenir des entrées autres que des chaînes de réécriture de l'en-tête MAC et des informations sur l'interface de sortie. Certains des divers types d'entrées qui peuvent être placées dans la table de juxtaposition incluent :

- **cache** — Une chaîne de réécriture de l'en-tête MAC et une interface sortante utilisées pour atteindre un hôte ou un routeur adjacent particulier.
- **recevez** — Des paquets destinés à cette adresse IP devraient être reçus par le routeur. Ceci inclut des adresses d'émission et des adresses configurées sur le routeur elle-même.
- **baisse** — Des paquets destinés à cette adresse IP devraient être lâchés. Ceci a pu être utilisé pour le trafic refusé par une liste d'accès, ou conduit à une interface null.
- **coup de volée** — Cisco Express Forwarding ne peut pas commuter ce paquet ; passez-le à la prochaine meilleure méthode de commutation (généralement commutation rapide) pour le traitement.
- **glanez** — Le prochain saut est directement relié, mais il n'y a aucune chaîne de réécriture de l'en-tête MAC actuellement disponible.

Glanez les contiguïtés

Une entrée de contiguïté de glaner indique qu'un prochain saut particulier devrait être directement connecté, mais il n'y a aucune informations disponibles de réécriture de l'en-tête MAC. Comment est-ce que ceux-ci obtiennent construit et utilisé ? Un routeur exécutant Cisco Express Forwarding et relié à un réseau de diffusion, suivant les indications de la figure ci-dessous, établit un certain nombre d'entrées de table de contiguïté par défaut.



Les quatre entrées de table de contiguïté établies par défaut sont :

```
10.1.1.0/24, version 17, attached, connected
0 packets, 0 bytes
  via Ethernet2/0, 0 dependencies
    valid glean adjacency
10.1.1.0/32, version 4, receive
10.1.1.1/32, version 3, receive
10.1.1.255/32, version 5, receive
```

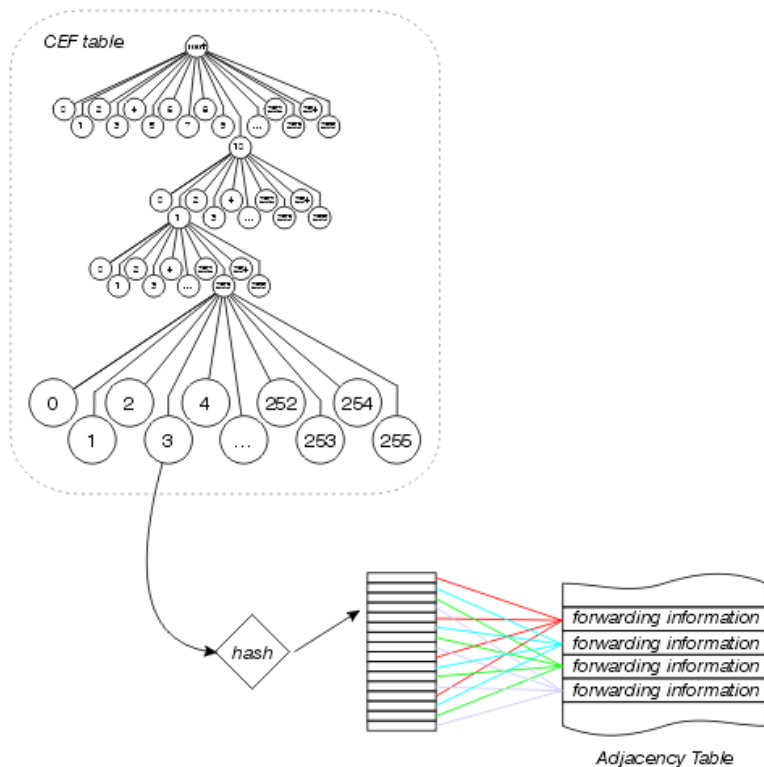
La note là sont quatre entrées : trois reçoit, et on glanent. Chacun reçoit l'entrée représente une adresse d'émission ou une adresse configurée sur le routeur, alors que l'entrée de glaner représente le reste de l'espace d'adressage sur le réseau relié. Si un paquet est reçu pour l'hôte 10.1.1.50, les tentatives de routeur de le commuter, et le trouve il l'a résolu à ceci glanent la contiguïté. Cisco Express Forwarding signale alors qu'une entrée de cache d'ARP est nécessaire pour 10.1.1.50, le processus d'ARP envoie un paquet d'ARP, et l'entrée de table appropriée de contiguïté est établie des nouvelles informations de cache d'ARP. Après que cette étape soit complète, la table de juxtaposition a une entrée pour 10.1.1.50.

```
10.1.1.0/24, version 17, attached, connected
0 packets, 0 bytes
  via Ethernet2/0, 0 dependencies
    valid glean adjacency
10.1.1.0/32, version 4, receive
10.1.1.1/32, version 3, receive
10.1.1.50/32, version 12, cached adjacency 208.0.3.2
0 packets, 0 bytes
  via 208.0.3.2, Ethernet2/0, 1 dependency
    next hop 208.0.3.2, Ethernet2/0
    valid cached adjacency
10.1.1.255/32, version 5, receive
```

Le paquet suivant que le routeur reçoit destiné à 10.1.1.50 est commuté par cette nouvelle contiguïté.

Partager de chargement

Cisco Express Forwarding tire profit également de la séparation entre la table de Cisco Express Forwarding et la table de juxtaposition pour fournir une meilleure forme de chargement partageant que tout autre mode de commutation de contexte d'interruption. Une table loadshare est insérée entre la table de Cisco Express Forwarding et la table de juxtaposition, comme illustré dans cette figure :



La table de Cisco Express Forwarding indique cette table loadshare, qui contient des pointeurs aux diverses entrées de table de contiguïté pour les chemins parallèles disponibles. La source et les adresses de destination sont traversées un algorithme de hachage pour déterminer quelle entrée dans la table loadshare à l'utiliser pour chaque paquet. Par chargement de paquet partager peut être configuré, dans ce cas chaque paquet utilise une entrée dans la table loadshare différente.

Chaque table loadshare a 16 entrées parmi lesquelles les chemins disponibles sont divisés ont basé sur le partage du trafic contre- dans la table de routage. Si les compteurs de partage du trafic dans la table de routage sont chacun des 1 (comme dans le cas des plusieurs chemins de coût égal), chaque prochain saut possible reçoit un nombre équivalent de pointeurs de la table loadshare. Si le nombre de chemins disponibles n'est pas également divisible dans 16 (puisque'il y a 16 entrées dans la table loadshare), quelques chemins auront plus d'entrées que d'autres.

Commençant dans le Logiciel Cisco IOS version 12.0, le nombre d'entrées dans la table loadshare est réduit pour assurer chaque chemin a un nombre proportionné d'entrées dans la table loadshare. Par exemple, s'il y a trois chemins de coût égal dans la table de routage, seulement 15 entrées dans la table loadshare sont utilisées.

Quel chemin de commutation est le meilleur ?

Autant que possible, vous voulez que vos Routeurs commutent dans le contexte d'interruption parce que c'est au moins une commande une grandeur plus rapide que la commutation de niveau de processus. La commutation de Cisco Express Forwarding est certainement plus rapide et mieux que n'importe quel autre mode de commutation. Nous vous recommandons utilisation Cisco Express Forwarding si le protocole et l'IOS que vous exécutez la prend en charge. C'est particulièrement vrai si vous avez un certain nombre de liens de parallèle à travers lesquels le trafic devrait être chargement partagé. Accédez à la page de navigateur de caractéristique de Cisco (clients enregistrés seulement) pour déterminer de quel IOS vous avez besoin pour le support de CEF.

Informations connexes

- **Comment vérifier la commutation Cisco Express Forwarding**
- **Dépannage de l'équilibrage de charge sur des liens parallèles utilisant Cisco Express Forwarding**
- **Livre blanc Technologies**