

# Présentation des algorithmes d'équilibrage de charge CSM

## Contenu

[Introduction](#)

[Avant de commencer](#)

[Conventions](#)

[Conditions préalables](#)

[Composants utilisés](#)

[Algorithmes d'équilibrage de charge](#)

[Recherche séquentielle](#)

[Moins connexions](#)

[Recherche séquentielle pesée et pesé moins connexions](#)

[Informations parasites IP de source et/ou de destination \(masque de sous-réseau configurable\)](#)

[Hachage URL](#)

[En avant](#)

[Informations connexes](#)

## [Introduction](#)

Le Module de commutation de contenu Cisco (CSM) prend en charge des algorithmes d'équilibrage de charge avec la commande de prédicteur. Émettez la commande prédicteur dans le sous-mode de configuration des grappes de serveur de l'équilibrage de charge de serveur (SLB) pour spécifier l'algorithme d'équilibrage de charge pour la grappe de serveurs.

## [Avant de commencer](#)

### [Conventions](#)

Pour plus d'informations sur les conventions des documents, référez-vous aux [Conventions utilisées pour les conseils techniques de Cisco](#).

### [Conditions préalables](#)

Aucune condition préalable spécifique n'est requise pour ce document.

### [Composants utilisés](#)

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Les informations présentées dans ce document ont été créées à partir de périphériques dans un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document commencé par une configuration (par défaut) effacée. Si vous travaillez dans un réseau opérationnel, assurez-vous de bien comprendre l'impact potentiel de toute commande avant de l'utiliser.

## Algorithmes d'équilibrage de charge

### Recherche séquentielle

L'algorithme par défaut, *circulaire*, mot clé dirige la connexion réseau au prochain serveur, et traite tous les serveurs comme égaux, indépendamment du nombre de connexions ou de temps de réponse. Bien que le predictor circulaire CSM ressemble à un Système de noms de domaine (DNS) circulaire, il est supérieur parce qu'aucun délai de propagation ou mise en cache ne gêne l'algorithme.

### Moins connexions

Le mot clé de *leastconns* dirige des connexions réseau au serveur avec les moins connexions. Bien qu'il puisse ne pas être intuitivement évident que le predictor de *leastconns* fournirait l'Équilibrage de charge efficace, en fait, il est tout à fait réussi. Sur des sites Web où il y a une collection de serveurs avec la représentation semblable, le predictor de *leastconns* est efficace dans la distribution sans heurt quand un serveur devient embourbé. Sur des sites Web où il y a de grandes différences en qualité de divers serveurs, le predictor de *leastconns* est également très efficace. En mettant à jour le même nombre de connexions à tous les serveurs, ces serveurs qui sont capables du traitement (et de ce fait de la terminaison) des connexions le plus rapide reçoivent plus de connexions au fil du temps. Un serveur considéré être deux fois plus puissant qu'un autre serveur reçoit environ deux fois autant de connexions par seconde.

### Recherche séquentielle pesée et pesé moins connexions

Le mot clé *pesé* te permet pour assigner un poids de représentation à chaque serveur. L'Équilibrage de charge pesé est semblable à la fonction des *leastconns* et les mots clé *circulaires*, cependant, des serveurs avec une valeur de poids plus élevé reçoivent un plus grand pourcentage des connexions en même temps. Les administrateurs CSM peuvent assigner un poids à chaque vrai serveur, et le CSM emploie ce poids pour déterminer le pourcentage du nombre en cours de connexions pour donner chaque serveur.

Émettez la commande de **poids** dans le vrai sous-mode de configuration du serveur SLB de configurer la capacité des vrais serveurs par rapport aux autres vrais serveurs dans la batterie de serveur. Émettez le **forme no de** cette commande de changer le poids du serveur à sa capacité par défaut.

la coefficient-valeur est la valeur à l'utiliser pour l'algorithme de predictor de ferme de serveur. La plage est de 1 à 100. Le poids par défaut est 8. par exemple, dans une configuration avec cinq serveurs, le pourcentage des connexions est calculé comme suit :

Server Number

Number of Connections

Weight of server 1

7

Weight of server 2	8
Weight of server 3	2
Weight of server 4	2
Weight of server 5	5

Total weight of all servers 24

Cette distribution a comme conséquence le serveur 1 7/24 obtenant du nombre en cours de connexions, le serveur 2 8/24 obtenant, le serveur 3 2/24 obtenant, et ainsi de suite. Si un nouveau serveur, le serveur 6, est ajouté avec un poids de 10, il reçoit 10/34, et ainsi de suite.

## [Informations parasites IP de source et/ou de destination \(masque de sous-réseau configurable\)](#)

La source ou la méthode d'informations parasites IP de destination trace l'adresse IP dans la demande à un vrai dans une batterie de serveur. Les options de configuration sont affichées ci-dessous.

```
predictor ip-hash netmask [source | destination] [netmask] !--- Source IP hash or predictor hash address.
```

Le par défaut (netmask) est 255.255.255.255. La valeur de hachage est produite avec l'algorithme suivant :

```
ip_addr = (Src_or_Dest_Ip_addr et Cfg_Netmask) hash_index = (ip_addr) + (ip_addr >> 8) + ip_addr >> 16) + (ip_addr >> 24) ;
```

## [Hachage URL](#)

Le hachage URL trace l'URL (ou la partie) pour la demande à un vrai dans une batterie de serveur. Pour spécifier seulement une partie de l'URL, vous devez configurer les chaînes commençantes et/ou finissantes de mot clé. Cette configuration est par objet de <vs-nam> de vsserver.

```
url-hash begin-pattern str [end-pattern str]
```

Tous les caractères dans les mots clé sont inclus en valeur de hachage. Afin de le tracer à un vrai, le CSM regarde les bits premiers en valeur de hachage. Si ce vrai est désactivé, le CSM trouve le prochain vrai disponible dans la liste. Dans la version 2.2(3) et ultérieures, le CSM regarde les bits plus élevés en valeur de hachage si les premiers bits d'informations parasites traçaient à un vrai handicapé. Si le CSM ne peut pas trouver un vrai activé, il regarde séquentiellement pour le prochain vrai disponible.

## [En avant](#)

Employez le mot clé **en avant** pour dire le CSM d'expédier le trafic selon ses tables de routage internes.

**Remarque:** L'ordre [nat de serveur](#) n'a aucun effet quand l'ordre [en avant de predictor](#) est configuré ; c'est parce que les serveurs ne peuvent pas être configurés.

## [Informations connexes](#)

- [Support et documentation techniques - Cisco Systems](#)