

CSS HTTP Keepalives

Contenu

[Introduction](#)

[Avant de commencer](#)

[Conventions](#)

[Conditions préalables](#)

[Composants utilisés](#)

[Configurez](#)

[Diagramme du réseau](#)

[Configuration](#)

[Dépannez](#)

[Configurer la méthode keepalive de HTTP](#)

[Directives d'utilisation](#)

[Informations connexes](#)

[Introduction](#)

Le Commutateur de services de contenu (CSS) de Cisco vote l'activité des serveurs employant différents protocoles pour vérifier l'intégrité des serveurs aux couches 3 5. L'interrogation peut être faite utilisant l'ICMP, le HTTP (TÊTE, somme de contrôle de page, URL), le TCP, et le FTP. Excepté le Système de noms de domaine (DNS), parce que le CSS est application de DN avertie, Cisco ne le prend en charge pas utilisant le Keepalives de Protocole UDP (User Datagram Protocol). Utilisant le HTTP, le vote peut vérifier si une page Web est servie et si elle a été modifiée. L'interrogation peut être exécutée aux niveaux de service individuel ou être attachée au service connexe.

[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](#)

Les informations dans ce document sont basées sur des Commutateurs de services satisfaits de gammes 11000 et 11500 de Cisco CSS et toutes les versions de Cisco WebNS.

Configurez

Le client a trois serveurs qui distribuent le contenu de base. Le deuxième serveur manipule également des transactions de base de données, et le troisième serveur manipule le contenu spécifique. Si la base de données principale descend, le deuxième serveur devrait être pris hors service ; de même, si le serveur de contenu descend, il devrait être aussi bien pris hors service.

Remarque: Pour obtenir des informations supplémentaires sur les commandes utilisées dans ce document, utilisez l'[Outil de recherche de commande \(clients enregistrés\)](#) seulement).

Diagramme du réseau

Configuration

Sydney CSS

```
!Generated on 05/01/2004 02:57:42
!Active version: sg0710102

configure

!***** GLOBAL
*****
ip route 0.0.0.0 0.0.0.0 209.165.202.130 1

!***** INTERFACE
*****
interface 1/2
bridge vlan 2

interface 2/2
bridge vlan 2

!***** CIRCUIT
*****
circuit VLAN1

ip address 209.165.202.128 255.255.255.224

circuit VLAN2

ip address 192.168.5.1 255.255.255.0

!***** KEEPALIVE
*****
keepalive ContentServer
!--- Poll the database server using HTTP. Use the
default HEAD method, which looks for !--- a 200 OK
response from the real server. type http uri
"/content.html" ip address 192.168.5.15 active keepalive
DatabaseServer type http !--- Use a GET and compare the
checksum of the login page. If there is a problem, it is
!--- displayed on the login page. method get ip address
192.168.5.10 uri "/dblogin.html" active
!***** SERVICE
*****
!--- Service rule for the
real servers. The CSS polls the server using the
```

```

specified named !--- keepalive. service ContentServer ip
address 192.168.5.15 keepalive type named ContentServer
active service DatabaseServer ip address 192.168.5.10
keepalive type named DatabaseServer active !--- Service
rule for the real servers. The CSS polls the server
using the default HTTP !--- keepalive type. service
IISWEB ip address 192.168.5.2 keepalive type http active
|***** OWNER
***** owner Website !--- Specify a
Layer 7 content rule. The CSS matches the most specific
rule first. content CONTENT protocol tcp vip address
209.165.202.129 port 80 url "/content.html" add service
ContentServer active content DATABASE protocol tcp vip
address 209.165.202.129 port 80 url "/dblogin.html" add
service DatabaseServer active content WWW protocol tcp
add service IISWEB vip address 209.165.202.129 port 80
active !--- Specify a group only if you have real
servers with private IP addresses; this will NAT !---
the packet on return to the client.
|***** GROUP
***** group WWW vip address
209.165.202.129 add destination service IISWEB active

```

Dépannez

Cette section présente des informations que vous pouvez utiliser pour vous assurer que votre configuration fonctionne correctement.

La commande **SH de keepalive** fournit un résumé complet de tout le Keepalives sur le CSS. Si vous voyez l'état : *Actif*, vous savez que ce service a été configuré, et le CSS peut voir le périphérique. Cette commande fournit également des informations sur quel type de keepalive c'est et les particularités sur ce que fera l'échouer de service. Les services commençant par *AUTO_* sont des services par défaut générés par le CSS quand vous ajoutez un service ; ceci inclut la passerelle par défaut. Si un service est affiché comme état : *Vers le bas*, vous devez vérifier votre connexion du CSS à votre vrai serveur (par défaut il prend trois manques d'une keepalive particulière de marquer un service en tant que vers le bas). État : *La mort* signifie que le CSS n'a pas encore eu le contact avec le vrai serveur.

```
SYDNEY# sh keepalive
```

```
Keepalives:
```

```
Name: AUTO_nexthop00004 Index: 0 State: Alive
Description: Auto generated for service nexthop00004
Address: 209.165.202.130 Port: Any
Type: ICMP
Frequency: 5
Max Failures: 3
Retry Frequency: 5
Dependent Services:
nexthop00004
```

```
Name: AUTO_IISWEB Index: 1 State: Alive
Description: Auto generated for service IISWEB
Address: 192.168.5.2 Port: 80
Type: HTTP:HEAD:/
Frequency: 5
Max Failures: 3
Retry Frequency: 5
```

Dependent Services:
IISWEB

Name: DatabaseServer Index: 2 State: Alive
Description:
Address: 192.168.5.10 Port: 80
Type: HTTP:GET:/dblogin.html
Hash: 95bd5419e38977e967b399853729c86f
Frequency: 5
Max Failures: 3
Retry Frequency: 5
Dependent Services:
DatabaseServer

Name: ContentServer Index: 3 State: Alive
Description:
Address: 192.168.5.15 Port: 80
Type: HTTP:HEAD:/content.html
Frequency: 5
Max Failures: 3
Retry Frequency: 5
Dependent Services:
ContentServer

Si vous avez un grand nombre de services, il est possible de répertorier seulement un service spécifié en tapant au nom du service que vous êtes intéressé dedans.

```
SYDNEY# sh keepalive ContentServer
Name: ContentServer Index: 3 State: Alive
Description:
Address: 192.168.5.15 Port: 80
Type: HTTP:HEAD:/content.html
Frequency: 5
Max Failures: 3
Retry Frequency: 5
Dependent Services:
ContentServer
```

[Configurer la méthode keepalive de HTTP](#)

Émettez la commande de **méthode keepalive** de spécifier la méthode keepalive de HTTP pour un service. La syntaxe et les options pour cette commande de mode de service sont :

- **la méthode obtiennent** — Le CSS émet un HTTP OBTIENNENT la méthode au service, calculent une valeur de hachage à la page, et enregistrent la valeur de hachage comme informations parasites de référence. GETs ultérieur exigent d'un état de 200 OKS (réponse CORRECTE terminée par commande de HTTP) et de la valeur de hachage d'égaliser la valeur de hachage de référence. Si l'état de 200 OKS n'est pas retourné, ou si l'état de 200 OKS est retourné mais la valeur de hachage est différent de la valeur de hachage de référence, le CSS considère le service vers le bas. Quand vous spécifiez les informations satisfaites d'un identifiant de ressource uniforme en HTTP (URI) pour une keepalive de HTTP, le CSS calcule une valeur de hachage pour le contenu. Si les changements des informations satisfaites, la valeur de hachage ne s'assortit plus la valeur de hachage d'origine et le CSS suppose que le service est en baisse. Pour empêcher le CSS de supposer qu'un service est en baisse en raison d'une non-concordance de valeur de hachage, spécifiez la méthode keepalive en tant que tête.
- **tête de méthode** (par défaut) — Le CSS fournit une méthode de HTTP HEAD au service et un

état de 200 OKS est exigé. Le CSS ne calcule pas une valeur de hachage de référence pour ce type de keepalive. Si l'état de 200 OKS n'est pas retourné, le CSS considère le service vers le bas. Par exemple, entrez :

Si vous changez la méthode keepalive à un service actif, assurez-vous que vous interrompez et réactivez le service pour la modification pour le prendre effet.

[Directives d'utilisation](#)

Quand vous spécifiez un identifiant de ressource uniforme (?) (URI) pour une keepalive de HTTP, le CSS calcule une valeur de hachage pour la page Web spécifiée dans l'URI. Si la page Web change, la valeur de hachage n'apparie plus la valeur de hachage d'origine et le CSS suppose que le service est mort. Pour empêcher le CSS de supposer qu'un service est complètement dû à une non-concordance de valeur de hachage, définissez la **méthode keepalive** en tant que **tête**. Le CSS ne calcule pas une valeur de hachage pour ce type de keepalive.

Si vous spécifiez l'URI d'une page Web avec le contenu changeable et ne spécifiez pas la méthode keepalive principale, vous devez interrompre et réactiver le service chaque fois que la page Web change.

[Maximum 512 Keepalives \(255 types de keepalive de script\) avec la version 5.0 de Cisco WebNS en avant](#)

Un CSS 11000 ou la configuration 11500 peut avoir jusqu'à 255 Keepalives de n'importe quel un type avec un total de tous les types étant inférieur ou égal à 512.

Exemple

Les besoins d'exemple.com de client d'équilibrer la charge 500 serveurs Web.

```
SYDNEY# sh keepalive ContentServer
Name: ContentServer Index: 3 State: Alive
Description:
Address: 192.168.5.15 Port: 80
Type: HTTP:HEAD:/content.html
Frequency: 5
Max Failures: 3
Retry Frequency: 5
Dependent Services:
ContentServer
```

[Informations connexes](#)

- [Guide de configuration de base CSS](#)
- [Commutateurs de services de contenu de la gamme Cisco CSS 11000](#)
- [Commandes de mode de configuration de keepalive](#)
- [Support technique - Cisco Systems](#)