

Configuration de l'usurpation IP sur le Cache Engine dans une installation transparente avec commutateur de services de contenu

Contenu

[Introduction](#)

[Avant de commencer](#)

[Conventions](#)

[Conditions préalables](#)

[Composants utilisés](#)

[Théorie générale](#)

[Configurer](#)

[Diagramme du réseau](#)

[Configurations](#)

[Vérifier](#)

[Dépanner](#)

[Introduction](#)

Ce document fournit une configuration d'échantillon pour la mise en cache transparente et usurpation d'adresse IP simultanément sans utiliser le Protocole WCCP (Web Cache Communications Protocol) sur le moteur de cache de Cisco et le Commutateur de services de contenu (CSS) 11000 de Cisco ou l'équilibreur de charge CSS 11500.

[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 les versions de logiciel et de matériel ci-dessous.

- Moteur de cache (CE) 500 application en cours d'exécution et logiciel réseau de Réseau de diffusion de contenu (ACNS) 4.2 ou plus tard
- CSS 11000 ou CSS 11500

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 ont démarré avec une configuration effacée (par défaut). Si vous travaillez dans un réseau opérationnel, assurez-vous de bien comprendre l'impact potentiel de toute commande avant de l'utiliser.

[Théorie générale](#)

La mise en cache transparente signifie que le trafic d'un client à un serveur est silencieusement réorienté par un routeur ou un commutateur de la couche 4 à un périphérique de cache (moteur de cache de Cisco dans ce cas).

Si le périphérique de cache a déjà une copie du contenu le client recherche, le cache répondra au nom du serveur. Si le contenu n'est pas présent sur le cache, le périphérique essaiera de l'obtenir du serveur avant de répondre à la demande de client.

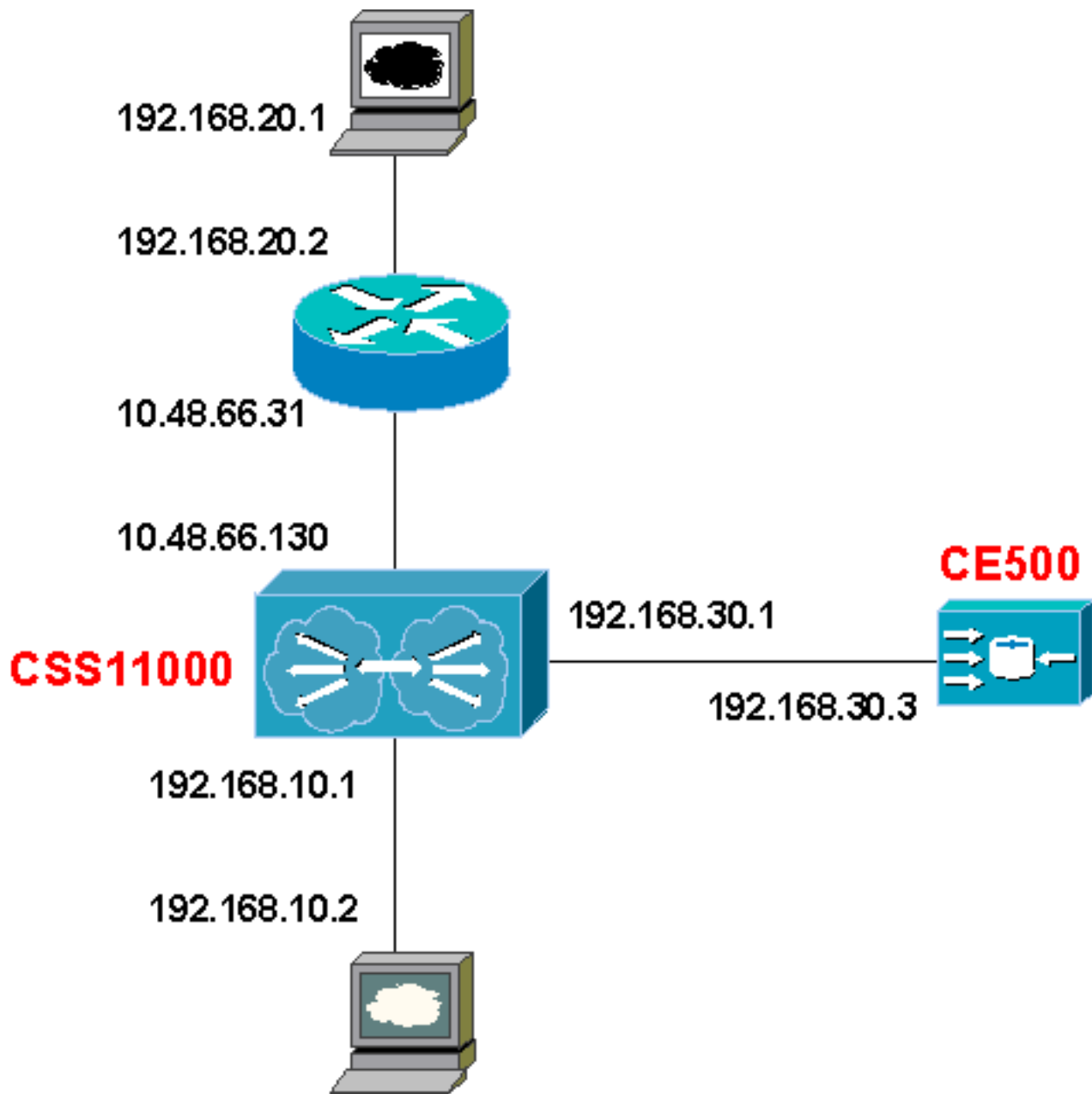
Par défaut, le cache contactera le serveur utilisant sa propre adresse IP. Il est parfois nécessaire, cependant, pour utiliser l'adresse IP de client. C'est faisable en configurant l'usurpation d'adresse IP.

[Configurer](#)

Cette section vous fournit des informations pour configurer les fonctionnalités décrites dans ce document.

[Diagramme du réseau](#)

Ce document utilise la configuration réseau indiquée dans le diagramme suivant :



Configurations

Ce document utilise les configurations présentées ci-dessous.

- CSS 11000
- Cache Engine 500

CSS 11000

```
!Generated on 04/18/2003 09:30:41
!Active version: ap10500007s

configure

!***** GLOBAL
*****
no restrict web-mgmt
no restrict xml
bridge spanning-tree disabled
persistence reset remap
```

```

acl enable
!--- An Access Control List (ACL) is needed. Enable the
ACL. ip route 0.0.0.0 0.0.0.0 10.48.66.1 1 ip route
192.168.10.0 255.255.255.0 192.168.20.100 1 ip route
192.168.20.0 255.255.255.0 10.48.66.31 1 ip route
192.168.20.0 255.255.255.0 192.168.30.3 1 !--- Very
important !!!! !--- For the ECMP feature of the CSS to
work, !--- you need one route pointing to the upstream
router, !--- and one identical route pointing to the
cache. !--- The CSS will know which one to use based on
where !--- the traffic came in first.
!*****
***** INTERFACE
***** interface e1 phy 100Mbps-FD
interface e2 bridge vlan 149 phy 100Mbps-FD interface
e3 bridge vlan 161 phy 100Mbps-FD
!*****
***** CIRCUIT
***** circuit VLAN1 ip address
10.48.66.130 255.255.254.0 circuit VLAN149 ip address
192.168.10.70 255.255.255.0 circuit VLAN161 ip address
192.168.30.1 255.255.255.0 !*****
SERVICE ***** service agra !---
Definition of the cache device. ip address 192.168.30.3
type transparent-cache !--- It is important to set the
type to transparent-cache !--- so that the CSS does not
NAT the destination IP address. !--- Only the
destination MAC address is modified. port 80 active
!*****
***** EQL
***** eql CacheMe !--- Definition
of what objects are cacheable. extension gif extension
html extension pdf extension zip extension gz
!*****
***** OWNER
***** owner gilles content ToCache
!--- Definition of the content rule to redirect the
traffic. !--- No VIP address specified since you want to
intercept all HTTP traffic. protocol tcp port 80 url
"/*" eql CacheMe !--- Redirect all requests of a
cacheable object. add service agra active
!*****
***** ACL
***** acl 1 clause 10 bypass tcp
any destination 192.168.10.2 eq 80 !--- This ACL is
necessary to make sure that the HTTP requests from !---
the cache itself are not intercepted by the content
rule. clause 20 permit any any destination any apply
circuit-(VLAN161) acl 2 !--- Permit all traffic for the
other interfaces. clause 20 permit any any destination
any apply circuit-(VLAN149) apply circuit-(VLAN1)

```

Cache Engine 500

```

hostname CE500
!
http 14-switch enable
!--- Tells the Cache Engine to accept traffic with any
IP destination. http 14-switch spoof-client-ip enable
!--- This is a new command in ACNS 5.x. this command
replaces the !--- wccp spoof-client-ip enable command.
!
!
!
!
!
!

```

```

!
exec-timeout 0
!
!
!
interface FastEthernet 0/0
 ip address 192.168.30.3 255.255.255.0
 exit
interface FastEthernet 0/1
 shutdown
 exit
!
!
ip default-gateway 192.168.30.1
!
primary-interface FastEthernet 0/0
!
!
!
logging console enable
!
!
!
!
!
!
!
wccp version 2
wccp spoof-client-ip enable
!--- This commands enable IP spoofing, and it works !---
even if you do not use WCCP. This command only works
with !--- WCCP redirected traffic if you have ACNS 5.x.
!--- Therefore, if you are using version 5.x of ACNS,
this command !--- should be replaced with the command
http 14-switch spoof-client-ip enable !--- mentioned
above.
!
!
CE500#

```

Vérifier

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

Certaines commandes **show** sont prises en charge par l'[Output Interpreter Tool](#) ([clients enregistrés](#) uniquement), qui vous permet de voir une analyse de la sortie de la commande show.

Commandes show CSS 11000

- **résumé d'exposition** — Affiche que les compteurs de hit de règle de contenu voyaient si le CSS est recevant et réorientant le trafic.
- *nom de **show service*** — Affiche le statut du service.

Commandes de moteur de cache

- **affichez le nom de demande de HTTP stat** — Affiche le nombre de demande de HTTP reçu par le cache.

- **affichez le *nom de l'épargne de HTTP stat*** — Affiche le nombre de hits et le coup manqué sur le moteur de cache.

Dépanner

Pour dépanner ce problème, utilisez les commandes ci-dessus. Le plus souvent, cependant, il est nécessaire d'employer un renifleur afin de découvrir le chemin précis suivi du trafic.

Vous pouvez également émettre la **commande trace d'écoulement** CSS disponible mettez au point dedans le mode. Vous obtiendrez un meilleur résultat avec un renifleur cependant.