Résolution des problèmes de mise en cache transparente inversée pour WCCP

Contenu

Introduction

Ce document explique comment effectuer le dépannage pour le Web Cache Communication Protocol (WCCP) lorsqu'il est utilisé afin d'intégrer la mise en cache transparente inversée.

Conditions préalables

Conditions requises

Aucune spécification déterminée n'est requise pour ce document.

Components Used

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

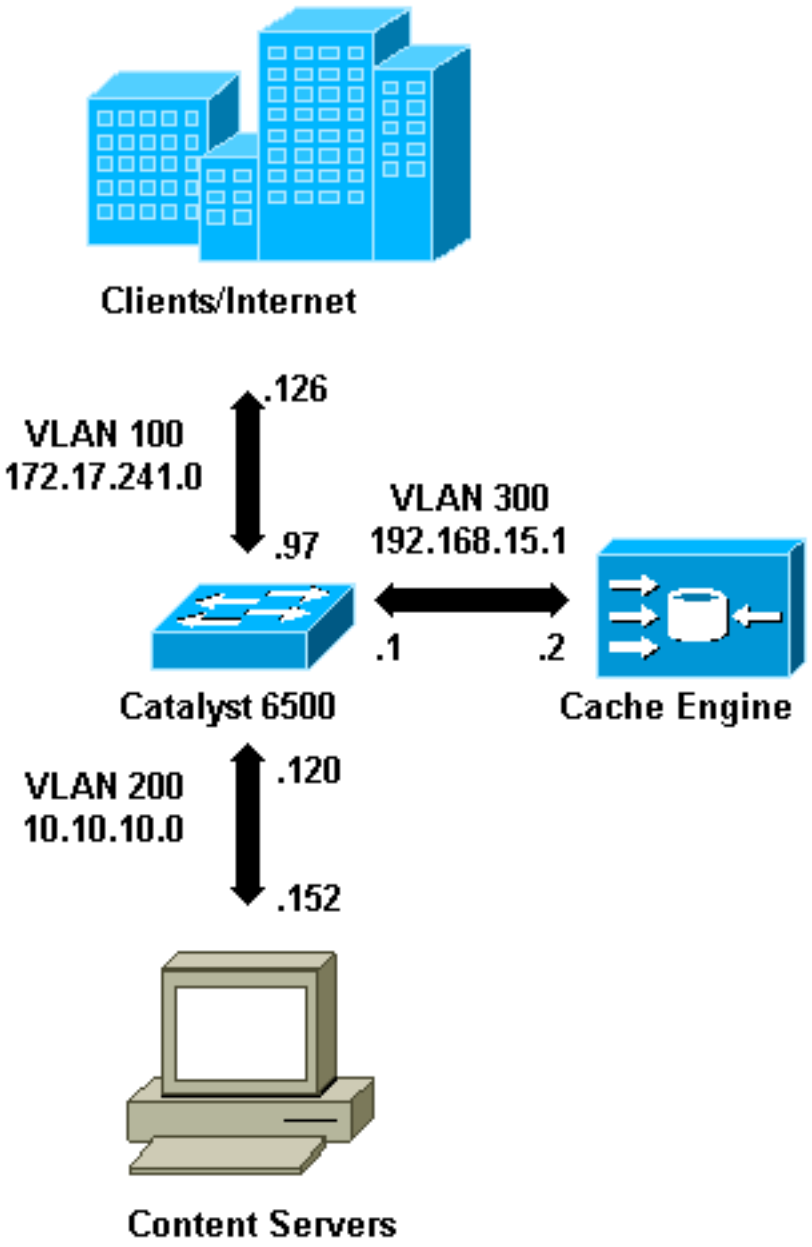
- Catalyst 6500 avec Supervisor 1 et MSFC 1 configurés en mode natif
- Logiciel Cisco IOS® Version 12.1(8a)EX (c6sup11-jsv-mz.121-8a.EX.bin)
- Cache Engine 550 avec version 2.51

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Conventions

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à Conventions relatives aux conseils techniques Cisco.

Configuration



Lorsque vous installez un Cache Engine, Cisco vous recommande de configurer uniquement les commandes nécessaires à la mise en oeuvre de WCCP. Vous pouvez ajouter d'autres fonctionnalités, telles que l'authentification au routeur et les listes de redirection des clients, à une date ultérieure.

Sur le Cache Engine, vous devez spécifier l'adresse IP du routeur et la version de WCCP que vous voulez utiliser.

```
wccp router-list 1 192.168.15.1
   wccp reverse-proxy router-list-num 1
   wccp version 2
```

Une fois l'adresse IP et la version de WCCP configurées, un message vous avertit que le service

99 doit être activé dans le routeur afin d'implémenter la mise en cache transparente inverse. Service 99 est l'identificateur de service WCCP pour la mise en cache transparente inverse. L'identificateur de la mise en cache transparente normale est le mot « cache web » dans Cisco IOS. Afin d'activer le service 99 (mise en cache transparente inversée) sur le routeur et afin de spécifier le port où la redirection sera effectuée, ajoutez ces commandes en mode de configuration globale :

```
ip wccp 99
interface Vlan200
    ip address 10.10.10.120 255.255.255.0
    ip wccp 99 redirect out
```

Lorsque vous configurez la mise en cache transparente inversée, le routeur qui exécute le service WCCP 99 intercepte les requêtes dirigées vers les serveurs Web. La commande **ip wccp 99 redirect out** est appliquée sur l'interface où vous voulez intercepter les paquets HTTP du client dans leur chemin vers votre serveur Web. Il s'agit généralement du VLAN du serveur Web. Il ne s'agit généralement pas du VLAN sur lequel le Cache Engine est installé.

Une fois WCCP actif, le routeur écoute sur tous les ports dont la redirection WCCP est configurée. Pour signaler sa présence, le Cache Engine envoie continuellement WCCP **ici**, **je suis** des paquets aux adresses IP configurées dans la liste des routeurs.

Une connexion WCCP entre le routeur et le cache est formée. Afin d'afficher les informations de connexion, émettez la commande **show ip wccp**.

L'identificateur de routeur est l'adresse IP du routeur telle qu'elle est vue par les moteurs de mémoire cache. Cet identificateur n'est pas nécessairement l'interface de routeur utilisée par le trafic redirigé pour atteindre le cache. L'identificateur de routeur dans cet exemple est 192.168.15.1.

```
Router#show ip wccp
   Global WCCP information:
       Router information:
           Router Identifier:                  192.168.15.1
           Protocol Version:                   2.0
    Service Identifier: 99
           Number of Cache Engines:            1
           Number of routers:                  1
           Total Packets Redirected:           0
           Redirect access-list:               -none-
           Total Packets Denied Redirect:      0
           Total Packets Unassigned:           0
           Group access-list:                  -none-
           Total Messages Denied to Group:     0
           Total Authentication failures:      0
```

La commande **show ip wccp 99 detail** fournit des informations détaillées sur les caches.

```
Router#show ip wccp 99 detail
   WCCP Cache-Engine information:
           IP Address:               192.168.15.2
```

```
        Protocol Version:         2.0
        State:                    Usable
        Redirection:              GRE
        Initial Hash Info:        FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                                  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
        Assigned Hash Info:       FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
                                  FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
        Hash Allotment:           256 (100.00%)
        Packets Redirected:       0
        Connect Time:             00:00:39
```

Le champ `Redirection` représente la méthode utilisée pour rediriger les paquets du routeur vers le Cache Engine. Cette méthode est soit l'encapsulation de routage générique (GRE), soit la couche 2. Avec GRE, les paquets sont encapsulés dans un paquet GRE. Avec la couche 2, les paquets sont envoyés directement au cache, mais le Cache Engine et le commutateur ou le routeur doivent être adjacents à la couche 2 pour la redirection de la couche 2.

L'`allocation de hachage` représentée au format hexadécimal dans les champs `Infos de hachage initiales` et `Infos de hachage affectées` est le nombre de compartiments de hachage affectés à ce cache. Toutes les adresses Internet source possibles sont divisées en 64 plages de taille égale, un regroupement par plage, et chaque cache reçoit le trafic d'un certain nombre de ces plages d'adresses source de groupement. Cette quantité est gérée dynamiquement par WCCP en fonction de la charge et de la pondération de charge du cache. Si un seul cache est installé, tous les compartiments peuvent être affectés à ce cache.

Lorsque le routeur commence à rediriger les paquets vers le Cache Engine, le nombre dans le champ `Total Packets Redirect` augmente.

Le champ `Total des paquets non affectés` indique le nombre de paquets qui n'ont pas été redirigés parce qu'ils n'ont été affectés à aucun cache. Dans cet exemple, le nombre de paquets est 5. Les paquets peuvent ne pas être attribués lors de la découverte initiale de caches ou pendant un petit intervalle lorsqu'un cache est supprimé.

```
Router#show ip wccp
   Global WCCP information:
       Router information:
           Router Identifier:                    192.168.15.1
           Protocol Version:                     2.0
    Service Identifier: 99
           Number of Cache Engines:              1
           Number of routers:                    1
           Total Packets Redirected:             28
           Redirect access-list:                 -none-
           Total Packets Denied Redirect:        0
           Total Packets Unassigned:             5
           Group access-list:                    -none-
           Total Messages Denied to Group:       0
           Total Authentication failures:        0
```

Si le cache n'est pas acquis par le routeur, il peut être utile de déboguer l'activité WCCP. Chaque fois que le routeur reçoit un paquet **ici je suis** du cache, il répond avec un paquet **je vous vois**, et ceci est signalé dans les débogages. Les commandes **debug** disponibles sont **debug ip wccp events** et **debug ip wccp packets**.

Remarque : Consulter les renseignements importants sur les commandes de débogage avant

d’utiliser les commandes de **débogage**.

Ce résultat fournit un exemple de messages de débogage WCCP normaux :

```
Router#debug ip wccp event
   WCCP events debugging is on
  Router#debug ip wccp packet
   WCCP packet info debugging is on
   Router#
   2d18h: WCCP-EVNT:S00: Built new router view: 0 routers,
          0 usable web caches, change # 00000001
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to
          192.168.15.2 w/ rcv_id 00000001
   2d18h: WCCP-EVNT:S00: Redirect_Assignment packet from
          192.168.15.2 fails source check
   2d18h: %WCCP-5-SERVICEFOUND: Service web-cache
          acquired on Web Cache 192.168.15.2
   2d18h: WCCP-PKT:S00: Received valid Here_I_Am packet
          from 192.168.15.2 w/rcv_id 00000001
   2d18h: WCCP-EVNT:S00: Built new router view: 1
          routers, 1 usable web caches, change # 00000002
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2
          w/ rcv_id 00000002
   2d18h: WCCP-EVNT:S00: Built new router view: 1 routers,
          1 usable web caches, change # 00000002
   2d18h: WCCP-PKT:S00: Received valid Redirect_Assignment
          packet from 192.168.15.2 w/rcv_id 00000002
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2
          w/ rcv_id 00000003
   2d18h: WCCP-EVNT:S00: Built new router view: 1 routers,
          1 usable web caches, change # 00000002
   2d18h: WCCP-PKT:S00: Received valid Redirect_Assignment
          packet from 192.168.15.2 w/rcv_id 00000003
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2
          w/ rcv_id 00000004
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2
          w/ rcv_id 00000005
   2d18h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2
          w/ rcv_id 00000006
   2d18h: WCCP-EVNT:S00: Built new router view: 1 routers,
          1 usable web caches, change # 00000002
   2d18h: WCCP-PKT:S00: Received valid Redirect_Assignment
          packet from 192.168.15.2 w/rcv_id 00000006
```

Afin d'augmenter le niveau de débogage, vous pouvez suivre le trafic de paquets IP afin de vérifier si le routeur reçoit des paquets du Cache Engine. Afin d'éviter la surcharge d'un routeur dans un environnement de production et afin d'afficher uniquement le trafic intéressant, vous pouvez utiliser une liste de contrôle d'accès pour limiter les débogages uniquement aux paquets dont l'adresse IP du cache est source. Un exemple de liste de contrôle d’accès est **access-list 130 permit ip host 192.168.15.2 host 192.168.15.1**.

```
Router#debug ip wccp event
   WCCP events debugging is on
  Router#debug ip wccp packet
   WCCP packet info debugging is on
  Router#debug ip packet 130
   IP packet debugging is on for access list 130
   2d19h: WCCP-EVNT:S00: Built new router view: 1 routers,  1 usable web caches,
```

```
       change # 00000002
2d19h: WCCP-PKT:S00: Received valid Redirect_Assignment  packet from 192.168.15.2
       w/rcv_id 0000001B
2d19h: datagramsize=174, IP 18390: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 0000001C
2d19h: datagramsize=174, IP 18392: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 0000001D
2d19h: datagramsize=174, IP 18394: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 0000001E
2d19h: datagramsize=378, IP 18398: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 364, fragment 0, fo 0, rcvd 3
2d19h: WCCP-EVNT:S00: Built new router view: 1 routers,  1 usable web caches,
       change # 00000002
2d19h: WCCP-PKT:S00: Received valid Redirect_Assignment  packet from 192.168.15.2
       w/rcv_id 0000001E
2d19h: datagramsize=174, IP 18402: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 0000001F
2d19h: datagramsize=174, IP 18404: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 00000020
2d19h: datagramsize=174, IP 18406: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 00000021
2d19h: datagramsize=378, IP 18410: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 364, fragment 0, fo 0, rcvd 3
2d19h: WCCP-EVNT:S00: Built new router view: 1 routers,  1 usable web caches,
       change # 00000002
2d19h: WCCP-PKT:S00: Received valid Redirect_Assignment  packet from 192.168.15.2
       w/rcv_id 00000021
2d19h: datagramsize=174, IP 18414: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
2d19h: WCCP-PKT:S00: Sending I_See_You packet to 192.168.15.2  w/ rcv_id 00000022
2d19h: datagramsize=174, IP 18416: s=192.168.15.2  (Vlan300), d=192.168.15.1
       (Vlan300), totlen 160, fragment 0, fo 0, rcvd 3
```

Si aucun cache n’est visible par le routeur et qu’aucune activité WCCP n’est détectée, vérifiez la connectivité de base. Essayez d'envoyer une requête ping au cache à partir du routeur ou du routeur à partir du cache. Si la requête ping fonctionne, une erreur peut exister dans la configuration.

Si le cache est acquis, mais qu'aucun paquet n'est redirigé, vérifiez que le routeur reçoit le trafic et que le trafic est transféré à l'interface où la commande **ip wccp 99 redirect out** est appliquée. Souvenez-vous que le trafic intercepté et redirigé est uniquement le trafic dirigé vers le port TCP 80.

Si le trafic n'est toujours pas redirigé et que le contenu Web provient directement des serveurs, vérifiez que le cache passe correctement les instructions sur ce qu'il faut intercepter. Vous devez disposer d'informations générales sur WCCP pour effectuer cette action.

WCCP reconnaît deux types de services différents : *standard* et *dynamique*. Le routeur connaît implicitement un service standard. En d’autres termes, il n’est pas nécessaire de demander au routeur d’utiliser le port 80, car il le sait déjà. La mise en cache transparente normale (cache Web - service standard 0) est un service standard.

Dans tous les autres cas (qui incluent la mise en cache transparente), le routeur est informé du port à intercepter. Ces informations sont transmises dans le paquet **Me voici**.

Vous pouvez émettre la commande **debug ip packet dump** afin d'examiner les paquets eux-mêmes. Utilisez la liste de contrôle d'accès créée pour déboguer uniquement les paquets envoyés par le Cache Engine.

```
Router#debug ip packet 130 dump
   2d19h: datagramsize=174, IP 19576: s=192.168.15.2 (Vlan300), d=192.168.15.1
      (Vlan300), totlen 160, fragment 0, fo 0,
   rcvd 3
   072C5120:                        0004 9B294800            ...)H.
!--- Start IP header. 072C5130: 00500F0D 25360800 450000A0 4C780000 .P..%6..E.. Lx.. 072C5140:
3F118F81 C0A80F02 C0A80F01 08000800 ?...@(..@(..... 072C5150: 008CF09E 0000000A 0200007C
00000004  ..p........|....
!--- Start WCCP header. 072C5160: 00000000 00010018 0163E606 00000515 .........cf..... 072C5170:
00500000 00000000 00000000 00000000  .P..............
!--- Port to intercept (0x50=80). 072C5180: 0003002C C0A80F02 00000000 FFFFFFFF
...,@(..........
!--- Hash allotment (FFFF...). 072C5190: FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF  ................
   072C51A0: FFFFFFFF FFFFFFFF FFFF0000 00000000  ................
   072C51B0: 00050018 00000002 00000001 C0A80F01  ............@(..
   072C51C0: 0000000C 00000001 C0A80F02 00080008  ........@(......
   072C51D0: 00010004 00000001 30                 ........0
```

Avec cette commande, vous pouvez déterminer si le port est annoncé ou non sans avoir à consulter l'intégralité de la requête de commentaires (RFC). Si le port n'est pas annoncé, le problème est plus probable dans la configuration du cache.

Référez-vous à Web Cache Coordination Protocol V2.0 pour plus d'informations.

Si le cache est acquis et que les paquets sont redirigés, mais que vos clients Internet ne peuvent pas parcourir vos serveurs, vérifiez si le cache est connecté à Internet et à vos serveurs. Envoyez une requête ping à partir du cache vers différentes adresses IP sur Internet et vers certains de vos serveurs internes. Si vous envoyez une requête ping à des domaines complets (URL) au lieu d'adresses IP, assurez-vous de spécifier le serveur DNS à utiliser dans la configuration du cache.

Si vous ne savez pas si le cache traite les requêtes, vous pouvez déboguer l'activité HTTP dans le cache. Afin de déboguer l'activité HTTP dans le cache, vous devez restreindre le trafic pour éviter de surcharger le cache. Sur le routeur, créez une liste de contrôle d'accès avec l'adresse IP source d'un client sur Internet que vous pouvez utiliser comme périphérique pour vos tests et utilisez l'option **redirect-list** de la commande globale **ip wccp 99**.

```
Router(config)#access-list 50 permit 172.17.241.126
Router(config)#ip wccp 99 redirect-list 50
```

Une fois que vous avez créé et appliqué la liste de contrôle d'accès, procédez comme suit :

1. Activez le débogage HTTP dans le cache à l'aide de la commande **debug http all** (Cisco Cache Engine version 2.x) ou **debug http all** (Cisco Cache Engine version 3 et ACNS version 4, 5).
2. Activez la surveillance du terminal (émettez la commande **term mon**).
3. Essayez de parcourir l'un de vos serveurs à partir du client que vous avez configuré dans la liste de contrôle d'accès.

Voici un exemple du résultat :

```
irq0#conf tcework_readfirstdata() Start the recv: 0xb820800 len 4096 timeout
   0x3a98 ms ctx 0xb87d800
   cework_recvurl() Start the request: 0xb20c800 0xb20c838 0xb20c8e0
   Http Request headers received from client:
   GET / HTTP/1.1
   Host: 10.10.10.152
   User-Agent: Links (0.92; Linux 2.2.16-22 i686)
   Accept: */*
   Accept-Charset: us-ascii, ISO-8859-1, ISO-8859-2, ISO-8859-4, ISO-8895-5,
      ISO-8859-13, windows-1250, windws-1251, windows-1257, cp437, cp850, cp852,
       cp866, x-cp866-u, x-mac-ce, x-kam-cs, x-koi8-r, x-koi8-u, utf8
   Connection: Keep-Alive

Protocol dispatch: mode=1 proto=2
   ValidateCode() Begin: pRequest=0xb20c800
   Proxy: CACHE_MISS: HealProcessUserRequest
   cework_teefile() 0xb20c800: Try to connect to server: CheckProxyServerOut():
      Outgoing proxy is not enable: 0xb20c800 (F)
   GetServerSocket(): Forwarding to server: pHost = 10.10.10.152, Port = 80
   HttpServerConnectCallBack : Connect call back socket = 267982944, error    = 0
   Http request headers sent to server:

   GET / HTTP/1.1
   Host: 10.10.10.152
   User-Agent: Links (0.92; Linux 2.2.16-22 i686)
   Accept: */*
   Accept-Charset: us-ascii, ISO-8859-1, ISO-8859-2, ISO-8859-4, ISO-8895-5,
      ISO-8859-13, windows-1250, windws-1251, windows-1257, cp437, cp850, cp852,
      cp866, x-cp866-u, x-mac-ce, x-kam-cs, x-koi8-r, x-koi8-u, utf8
   Connection: keep-alive
   Via: 1.1 irq0
   X-Forwarded-For: 172.17.241.126

cework_sendrequest: lBytesRemote = 386, nLength = 386 (0xb20c800)
   ReadResCharRecvCallback():  lBytesRemote = 1818, nLength = 1432 0xb20c800)
   IsResponseCacheable() OBJECTSIZE_IS_UNLIMITED, lContentLength = 3194
   cework_processresponse() : 0xb20c800 is cacheable
   Http response headers received from server:
   HTTP/1.1 200 OK
   Date: Tue, 20 Nov 2001 10:46:14 GMT
   Server: Apache/1.3.12 (Unix)  (Red Hat/Linux) mod_ssl/2.6.6 OpenSSL/0.9.5a
      mod_perl/1.24
   Last-Modified: Fri, 12 Oct 2001 12:55:23 GMT
   ETag: "5e23-c7a-3bc6e83b"
   Accept-Ranges: bytes
   Content-Length: 3194
   Keep-Alive: timeout=15, max=100
   Connection: Keep-Alive
   Content-Type: text/html

GetUpdateCode(): GET request from client, GET request to server.
   GetUpdateCode(): nRequestType = -1
   SetTChain() 0xb20c800: CACHE_OBJECT_CLIENT_OBJECT sendobj_and_cache
   Http response headers sent to client:
   HTTP/1.1 200 OK
   Date: Tue, 20 Nov 2001 10:46:14 GMT
   Server: Apache/1.3.12 (Unix)  (Red Hat/Linux) mod_ssl/2.6.6 OpenSSL/0.9.5a
      mod_perl/1.24
   Last-Modified: Fri, 12 Oct 2001 12:55:23 GMT
   ETag: "5e23-c7a-3bc6e83b"
   Content-Length: 3194
```

```
   Keep-Alive: timeout=15, max=100
   Content-Type: text/html
   Connection: keep-alive

cework_tee_sendheaders() 0xb20c800: sent 323 bytes to client
   cework_tee_send_zbuf() 0xb20c800: Send 1087 bytes to client (1087)
   UseContentLength(): Valid Content-Length (T)
   cework_tee_recv_zbuf() 0xb20c800: Register to recv 2107 bytes timeout 120 sec
   HttpServerRecvCallBack(): Recv Call Back socket 267982944, err 0, length  2107
   HttpServerRecvCallBack(): lBytesRemote = 3925, nLength = 2107 (186697728)
   cework_tee_send_zbuf() 0xb20c800: Send 2107 bytes to client (2107)
   UseContentLength(): Valid Content-Length (T)
   cework_setstats(): lBytesLocal = 0, lBytesRemote = 3925 (0xb20c800)
   cework_readfirstdata() Start the recv: 0xb84a080 len 4096 timeout 0x3a98
      ms ctx 0xb87d800
   cework_cleanup_final() End the request: 0xb20c800 0xb20c838 0xb20c8e0
```

Les informations pertinentes que vous pouvez trouver dans le débogage sont mises en surbrillance en **gras**.

Voici les différentes phases d'une transaction de page Web :

1. En-têtes de requête HTTP reçus du client.
2. En-têtes de requête HTTP envoyés au serveur.
3. En-têtes de réponse HTTP reçus du serveur.
4. En-têtes de réponse HTTP envoyés au client.

Si la page Web que vous parcourez contient plusieurs objets, plusieurs instances de cette séquence d'événements existent. Utilisez la requête la plus simple possible pour réduire la sortie de débogage.

Sur un routeur Catalyst 6500 ou Cisco 7600, un gestionnaire de fonctionnalités gère toutes les fonctionnalités configurées dans Cisco IOS afin de fournir une couche supplémentaire de dépannage. Lorsqu'une fonctionnalité de couche 3 est configurée dans ces périphériques, les informations qui définissent la manière de gérer les trames reçues sont transmises aux fonctions de contrôle de couche 2 du commutateur ou du routeur (gestionnaire de fonctionnalités). Pour WCCP, ces informations de contrôle définissent les paquets interceptés par IOS et WCCP et dirigés vers le cache transparent.

La commande **show fm Features** affiche les fonctionnalités activées dans Cisco IOS. Vous pouvez utiliser cette commande afin de vérifier si le port à intercepter est correctement annoncé par le Cache Engine.

```
Router#show fm features
   Redundancy Status: stand-alone
   Interface: Vlan200 IP is enabled
     hw[EGRESS] = 1, hw[INGRESS] = 1
     hw_force_default[EGRESS] = 0, hw_force_default[INGRESS] = 0
     mcast = 0
     priority = 2
     reflexive = 0
     vacc_map :
     outbound label: 5
           merge_err: 0
           protocol: ip
             feature #: 1
```

```
              feature id: FM_IP_WCCP
              Service ID: 99
              Service Type: 1

The following are the used labels
     label 5:
           swidb: Vlan200
           Vlous:

The following are the features configured
     IP WCCP: service_id = 99, service_type = 1, state = ACTIVE
           outbound users:
             user_idb: Vlan200
           WC list:
             address: 192.168.15.2
           Service    ports:
             ports[0]:    80

The following is the ip ACLs port expansion information
     FM_EXP knob configured: yes

FM mode for WCCP: GRE (flowmask: destination-only)

FM redirect index base: 0x7E00

The following are internal statistics
     Number of pending tcam inserts: 0
     Number of merge queue elements: 0
```

La commande **show fm int vlan 200** affiche le contenu exact de la mémoire TCAM (Ternary Content Addressable Memory).

```
Router#show fm int vlan 200
 Interface: Vlan200 IP is enabled
  hw[EGRESS] = 1, hw[INGRESS] = 1
  hw_force_default[EGRESS] = 0, hw_force_default[INGRESS] = 0
  mcast = 0
  priority = 2
  reflexive = 0
  vacc_map :
  outbound label: 5
   merge_err: 0
   protocol: ip
    feature #: 1
    feature id: FM_IP_WCCP
    Service ID: 99
    Service Type: 1
     (only for IP_PROT) DestAddr SrcAddr          Dpt   Spt  L4OP TOS Est  prot  Rslt
     vmr IP value #1:   0.0.0.0  192.168.15.2     0     0    0    0   0    6     permit
     vmr IP mask  #1:   0.0.0.0  255.255.255.255  0     0    0    0   0    FF
     vmr IP value #2:   0.0.0.0  0.0.0.0          80    0    0    0   0    6     bridge
     vmr IP mask  #2:   0.0.0.0  0.0.0.0          FFFF  0    0    0   0    FF
     vmr IP value #3:   0.0.0.0  0.0.0.0          0     0    0    0   0    0     permit
     vmr IP mask  #3:   0.0.0.0  0.0.0.0          0     0    0    0   0    0
```

La `valeur IP vmr n° 1 :` définit le contournement d'interception sur les trames provenant du Cache Engine. Sans cela, il y aurait une boucle de redirection. La `valeur IP vmr n° 2 :` définit l'interception de tous les paquets dont le port 80 est leur destination. Si le port 80 n'est pas affiché dans la deuxième ligne, mais que WCCP est actif et que le cache est utilisable par le routeur, il

peut y avoir un problème dans la configuration du cache. Récupérez un vidage du paquet **ici je suis** afin de déterminer si le port est envoyé ou non par le cache.

Si vous ne parvenez pas à résoudre le problème après le dépannage, signalez le problème au centre d'assistance technique Cisco.

Voici quelques informations de base que vous devez fournir au TAC Cisco. À partir du routeur, collectez ces informations :

- Résultat de la commande **show tech**. La sortie des commandes **show running-config** et **show version output** peut être remplacée si la taille de la sortie **show tech** pose problème.
- Sortie de la commande **show ip wccp**.
- Sortie de la commande **show ip wccp web-cache detail**.
- S'il semble y avoir un problème de communication entre le routeur et le cache Web, fournissez le résultat des commandes **debug ip wccp events** et **debug ip wcp packets** pendant que le problème se produit.

Sur le Cache Engine (Cisco Cache Engine uniquement), collectez le résultat de la commande **show tech**.

Lorsque vous contactez le TAC, procédez comme suit :

1. Fournir une description claire du problème. Vous devez inclure les réponses à ces questions :Quels sont les symptômes ?Est-ce que cela arrive tout le temps ou rarement ?Le problème a-t-il commencé après une modification de la configuration ?Les caches Cisco ou tiers sont-ils utilisés ?
2. Fournir une description claire de la topologie. Inclure un diagramme si cela le rendra plus clair.
3. Fournissez toute autre information que vous jugez utile pour résoudre le problème.

Voici le résultat d'un exemple de configuration :

```
********************************** Router Configuration ****************************
Router#show running
   Building configuration...
Current configuration : 4231 bytes
   !
   version 12.1
   service timestamps debug uptime
   service timestamps log uptime
   no service password-encryption
   !
   hostname Router
   !
   boot buffersize 126968
   boot bootldr bootflash:c6msfc-boot-mz.120-7.XE1
   !
   redundancy
    main-cpu
     auto-sync standard
   ip subnet-zero
   ip wccp 99
   !
   !
   !
```

```
   interface FastEthernet3/1
    no ip address
    switchport
    switchport access vlan 100
    switchport mode access
   !
   interface FastEthernet3/2
    no ip address
    switchport
    switchport access vlan 200
    switchport mode access
   !
   interface FastEthernet3/3
    no ip address
    switchport
    switchport access vlan 300
    switchport mode access
   !
   interface FastEthernet3/4
    no ip address
   !
!
   interface Vlan100
    ip address 172.17.241.97 255.255.255.0
   !
   interface Vlan200
    ip address 10.10.10.120 255.255.255.0
    ip wccp 99 redirect out
   !
   interface Vlan300
    ip address 192.168.15.1 255.255.255.0
   !
   ip classless
   ip route 0.0.0.0 0.0.0.0 172.17.241.1
   no ip http server
   !
   access-list 30 permit 192.168.15.2
   !
   !
   line con 0
    exec-timeout 0 0
   line vty 0 4
    login
    transport input lat pad mop telnet rlogin udptn    nasi
   !
   end
********************************** Cache Configuration ******************************
Cache#show running
Building configuration...
   Current configuration:
   !
   !
   logging disk /local/syslog.txt debug
   !
   user add admin uid 0  capability admin-access
   !
   !
   !
   hostname Cache
   !
   interface ethernet 0
    ip address 192.168.15.2 255.255.255.0
    ip broadcast-address 192.168.15.255
    exit
```

```
!
interface ethernet 1
 exit
!
ip default-gateway 192.168.15.1
ip name-server 172.17.247.195
ip domain-name cisco.com
ip route 0.0.0.0 0.0.0.0 192.168.15.1
cron file /local/etc/crontab
!
wccp router-list 1 192.168.15.1
wccp reverse-proxy router-list-num 1
wccp version 2
!
authentication login local enable
authentication configuration local enable
rule no-cache url-regex .*cgi-bin.*
rule no-cache url-regex .*aw-cgi.*
!
!
end
```

Informations connexes

- Logiciel Cisco Cache
- Moteurs de mémoire cache de la gamme Cisco 500
- Protocole de communication de cache Web (WCCP)
- Page de téléchargement du logiciel Cisco Cache Engine 2.0 (clients enregistrés uniquement)
- Page de téléchargement du logiciel Cisco Cache Engine 3.0 (clients enregistrés uniquement)
- Support et documentation techniques - Cisco Systems