

コンテンツ スイッチング モジュールでの透過キャッシングの設定例

目次

[概要](#)

[はじめに](#)

[要件](#)

[使用するコンポーネント](#)

[表記法](#)

[設定](#)

[ネットワーク図](#)

[設定](#)

[確認](#)

[トラブルシューティング](#)

[関連情報](#)

概要

このドキュメントでは、Cisco Cache Engine およびコンテンツ スイッチング モジュール (CSM) を使用する透過キャッシングの設定例を紹介します。透過キャッシングとは、Web ブラウザからのトラフィックを透過的にインターセプトし、以前キャッシュされていた内容を取得するためにキャッシュ デバイスにリダイレクトするために使用される技術です。

トランスペアレント キャッシングをするもう一つの方式は Web キャッシュ通信プロトコル (WCCP) です。WCCP 上のトランスペアレント キャッシングの長所はことクライアントが要求する URL の CSM 外観、トラフィックがキャッシュに送信されるかどうか決定します。gif または jpeg イメージのような静的ファイルのための要求はキャッシュから動的ページ (スクリプトの結果) はへからキャッシュ行かないで直接サーバ取得されるが、取得されます。

はじめに

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントの情報は、次のハードウェアとソフトウェアのバージョンに基づくものです。

- CSM バージョン 3.x
- アプリケーション Content Networking Software (ACNS) バージョン 5.1

表記法

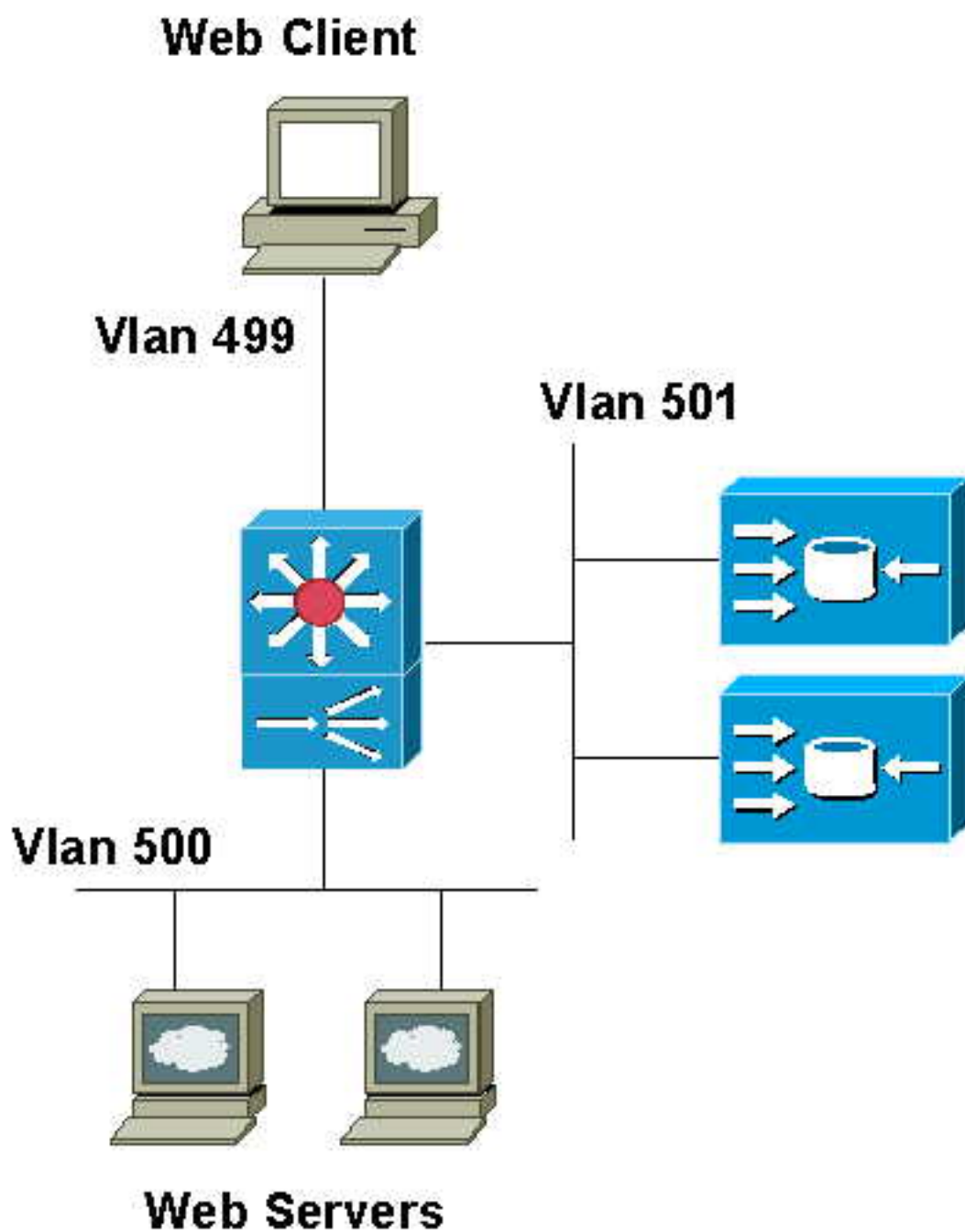
ドキュメント表記の詳細は、『[シスコテクニカルティップスの表記法](#)』を参照してください。

設定

この項では、このドキュメントで説明する機能の設定に必要な情報を提供します。

ネットワーク図

このドキュメントでは次の図に示すネットワーク



設定

このドキュメントでは次の設定を使用しています。

```
module ContentSwitchingModule 4 vlan 501 server ip address 192.168.30.97 255.255.254.0 ! vlan
499 client ip address 192.168.10.97 255.255.254.0 gateway 192.168.10.1 ! vlan 500 server ip
address 192.168.20.97 255.255.254.0 ! serverfarm CACHES no nat server !--- This is a transparent
redirect; do not change the destination IP address. no nat client predictor hash url !--- Use
URL hashing to make sure the request for a specific URL always goes to the same server. real
192.168.30.200 inservice real 192.168.30.201 inservice ! serverfarm FORWARD no nat server no nat
client predictor forward !--- This serverfarm tells the CSM not to load balance. !--- The CSM
instead uses its routing table to forward the traffic. ! map CACHEABLE url !--- In this example,
you want to only redirect requests for certain file types. !--- This is not mandatory. !--- You
can also adjust this to something more realistic. match protocol http url *.html match protocol
http url *.gif match protocol http url *.jpg match protocol http url *.exe match protocol http
url *.zip ! policy CACHEABLE !--- The policy is the way to link the map with a serverfarm. url-
map CACHEABLE serverfarm CACHES ! vserver FROMCACHE !--- This rule is for traffic originating
from the caches (when they have !--- to retrieve content from the origin server). virtual
0.0.0.0 0.0.0.0 any vlan 501 !--- The VLAN command guarantees that you limit this vserver to the
cache VLAN. serverfarm FORWARD !--- Use the serverfarm FORWARD command to disable load balancing
for this traffic. !--- In this example, you need forward requests from the caches to the origin
server. !--- You could, however, load balance this traffic to a series of Web servers, that is,
!--- when doing reverse proxy caching. persistent rebalance inservice ! vserver INTERCEPT !---
This is the rule to transparently redirect requests from the client to the caches. virtual
0.0.0.0 0.0.0.0 tcp www vlan 499 serverfarm FORWARD !--- The default action is forward; no load
balancing. !--- This is for requests that do not match the policy. persistent rebalance slb-
policy CACHEABLE !--- Traffic matching the policy is load balanced to the caches. inservice !
vserver NONHTTP !--- Non-HTTP traffic from the clients is forwarded. virtual 0.0.0.0 0.0.0.0 any
vlan 499 serverfarm FORWARD persistent rebalance inservice !
```

確認

このセクションでは、設定が正常に動作しているかどうかを確認する際に役立つ情報を提供しています。

- show mod csm X vserver **ネーム** **ネーム**詳細
- show mod csm X conns 詳細

```
EOMER#show mod csm 4 vser name intercept det INTERCEPT, type = SLB, state = OPERATIONAL, v_index
= 22 virtual = 0.0.0.0/0:80 bidir, TCP, service = NONE, advertise = FALSE idle = 3600, replicate
csrp = none, vlan = 499, pending = 30, layer 4 max parse len = 2000, persist rebalance = TRUE
ssl sticky offset = 0, length = 32 conns = 0, total conns = 3 Default policy: server farm =
FORWARD, backup = <not assigned> sticky: timer = 0, subnet = 0.0.0.0, group id = 0 Policy Tot
matches Client pkts Server pkts ----- CACHEABLE
2 410 926 (default) 5 20 17
```

トラフィックがポリシー（キャッシュにリダイレクトされるトラフィック）と一致したこと、またはトラフィックが転送されたら確認して下さい（デフォルトポリシーの一致）。

```
EOMER#show mod csm 4 conn det prot vlan source destination state -----
----- In ICMP 499 192.168.11.41 192.168.21.4 ESTAB Out ICMP
500 192.168.21.4 192.168.11.41 ESTAB vs = NONHTTP, ftp = No, csrp = False In ICMP 501
192.168.10.107 10.48.66.102 ESTAB Out ICMP 499 10.48.66.102 192.168.10.107 ESTAB vs = FROMCACHE,
ftp = No, csrp = False In TCP 499 192.168.11.41:4402 192.168.21.4:80 REQ_WAIT Out TCP 501
192.168.21.4:80 192.168.11.41:4402 REQ_WAIT vs = INTERCEPT, ftp = No, csrp = False In TCP 501
192.168.11.41:32784 192.168.21.4:80 ESTAB Out TCP 500 192.168.21.4:80 192.168.11.41:32784 ESTAB
vs = FROMCACHE, ftp = No, csrp = False
```

キャッシュは IPスプーフィングのために設定されました。そこにそれの上の出力ですクライアント 192.168.11.41 からの VLAN 499 で見られるサーバ 192.168.21.4 への接続および VLAN 501 で見られる同じような接続見ることができます。最初の 1 つはキャッシュにリダイレクトされた

クライアントからの実際の接続です (出口 VLAN は 501) あり、第 2 1 は起源 サーバへのキャッシュ (スプーフィング クライアントIPアドレス) からの接続です。

[トラブルシューティング](#)

ここでは、設定のトラブルシューティングに役立つ情報について説明します。

[関連情報](#)

- [コンテンツ スイッチング モジュールでのセキュア \(ルータ \) モードの設定](#)
- [コンテンツ スイッチング モジュール ハードウェアに関するサポート](#)
- [Cisco Cat 6000 他のインテリジェントモジュール SW ダウンロード \(登録ユーザのみ \)](#)
- [テクニカルサポートとドキュメント - Cisco Systems](#)