

단일 무장 프록시 모드를 위해 SCA 팜에 SSL을 로드 밸런싱하도록 CSM 구성

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[표기 규칙](#)

[구성](#)

[네트워크 다이어그램](#)

[구성](#)

[다음을 확인합니다.](#)

[문제 해결](#)

소개

이 문서에서는 SCA(Secure Content Accelerator) 팜에 대한 SSL(Secure Socket Layer) 트래픽의 CSM(Content Switching Module) 로드 밸런싱에 대한 샘플 컨피그레이션을 제공합니다. 이 컨피그레이션은 단일 포트 모드에서 연결을 사용하는 비투명 프록시 모드의 SCA를 위한 것입니다.

비투명 모드에서는 SCA가 SCA IP 주소를 웹 서버에 대한 일반 텍스트 연결의 소스로 사용합니다.

참고: SCA 및 웹 서버에 서로 다른 두 개의 VLAN/IP 하위 네트워크를 사용합니다. 하나의 하위 네트워크는 모든 SCA를 위한 것이고, 별도의 하위 네트워크는 모든 웹 서버를 위한 것입니다. 두 팜을 동일한 L2(Layer 2) 도메인에 배치할 경우 소스 NAT(Network Address Translation)가 필요합니다. 소스 NAT는 패킷이 CSM으로 반환되고 Catalyst 하드웨어가 단순히 L2에서 패킷을 전환하지 않도록 보장합니다.

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

이 문서의 정보는 다음 VLAN/하위 네트워크를 기반으로 합니다.

- 클라이언트 측: 가상 IP(VIP) 및 업스트림 라우터(MSFC[Multilayer Switch Feature Card])
- Catalyst 6500/6000(슬롯 5에 CSM 포함)

- 서버측 1:웹 서버
- 서버 측 2:SCA

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다.이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다.현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

표기 규칙

문서 규칙에 대한 자세한 내용은 [Cisco 기술 팁 표기 규칙](#)을 참조하십시오.

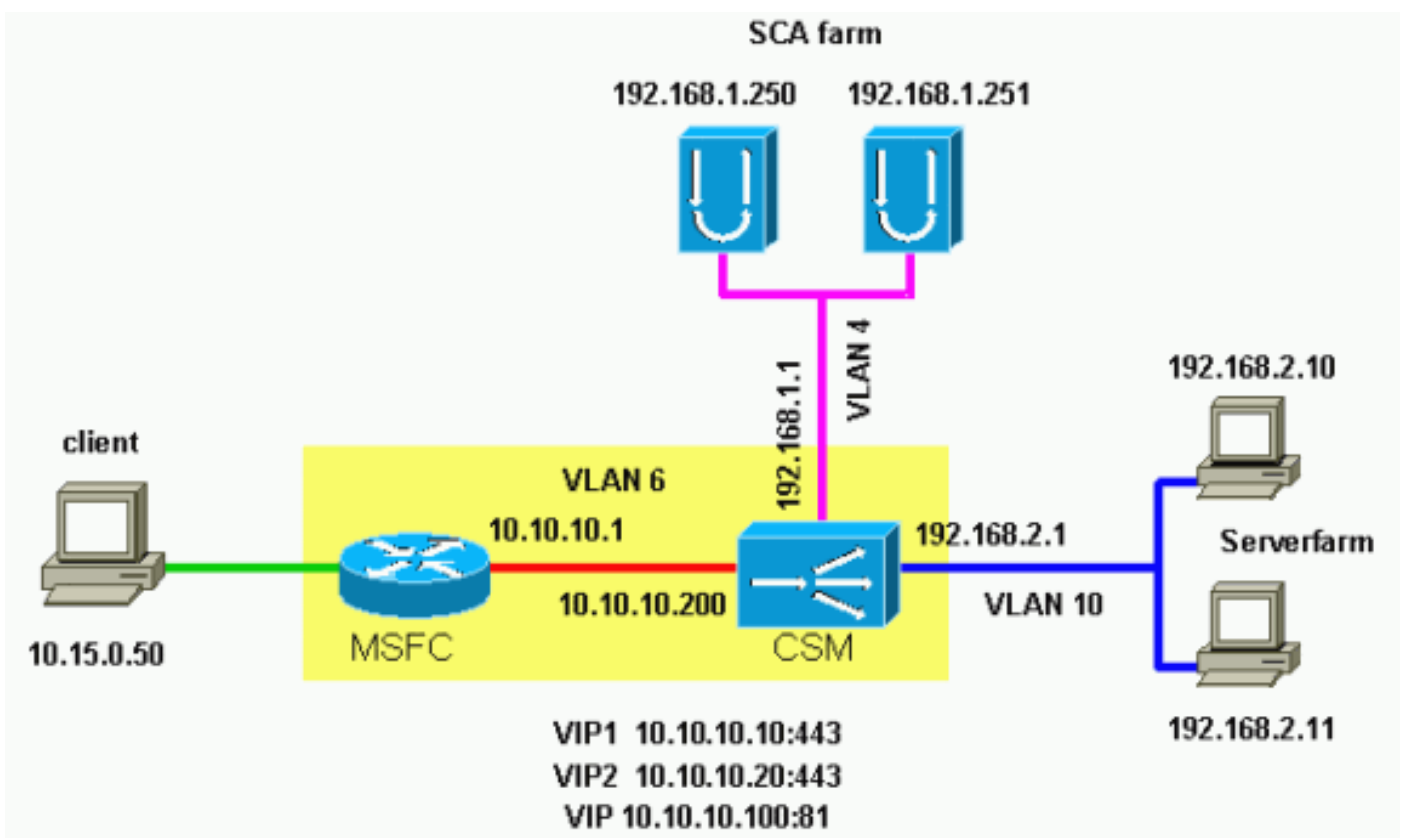
구성

이 섹션에서는 이 문서에 설명된 기능을 구성하는 정보를 제공합니다.

참고: 이 문서에 사용된 명령에 대한 추가 정보를 찾으려면 [명령 조회 도구](#)([등록된](#) 고객만 해당)를 사용합니다.

네트워크 다이어그램

이 문서에서는 다음 네트워크 설정을 사용합니다.



구성

이 문서에서는 다음 구성을 사용합니다.

- Catalyst 6000/CSM Slot 5
- SCA 1

• SCA 2

Catalyst 6000/CSM Slot 5

```
!--- This is the configuration of nontransparent SSL
load balance. Cat6k# show running-config | begin Module
5
module ContentSwitchingModule 5
  vlan 6 client
    ip address 10.10.10.200 255.255.255.0
    gateway 10.10.10.1
  !--- This is the CSM IP address on the client side and
  !--- CSM upstream gateway (the MSFC). ! vlan 4 server ip
address 192.168.1.1 255.255.255.0 !--- This is the CSM
IP address on the SCA server farm VLAN. !--- SCAs use
this IP address as the default gateway. ! vlan 10 server
ip address 192.168.2.1 255.255.255.0 !--- This is the
CSM IP address on the web server farm VLAN. !--- The web
servers use this IP address as the default gateway. !
static drop real 192.168.2.0 255.255.255.0 !--- This
drops every new connection that the web servers
originate, !--- unless the connection matches a VIP. !
serverfarm SCA443 nat server !--- When connections are
directed to this server farm, !--- the IP address of the
SCA selection replaces !--- the destination IP address.
no nat client real 192.168.1.250 443 inservice real
192.168.1.251 443 inservice !--- The configurations of
both SCAs are such that, !--- with the send of a
connection to this server farm, the destination port !--
- translates to 443. In this example, there is no
translation, as !--- the VIP listens to port 443. !---
This is different in the following server farm, SCA444.
! serverfarm SCA444 nat server no nat client real
192.168.1.250 444 inservice real 192.168.1.251 444
inservice !--- With the selection of this server farm,
there is a !--- modification of connections that go to
either SCA. !--- The destination IP changes to match the
IP of one of the SCAs !--- (NAT server), and the
destination port becomes 444. ! serverfarm WEBFARM nat
server no nat client real 192.168.2.10 80 inservice real
192.168.2.11 80 !--- Specify port 80 to translate from
port 81 inservice. !--- (The SCA communicates on port
81, according to the SCA setup.) !--- This is a standard
web server farm. ! sticky 10 ssl timeout 60 sticky 20
ssl timeout 60 !--- This creates two distinct sticky
groups with SSL ID as a basis. !--- The timeout is 60
seconds. ! vserver TESTSITE1 virtual 10.10.10.10 tcp
https serverfarm SCA443 sticky 60 group 10 persistent
rebalance inservice !--- The vserver for the first site
(www.testsite1.com) listens !--- to 10.10.10.10 on port
443. !--- Connections go to the SCAs without a change in
the !--- destination port. (See the configuration of
server farm SCA443.) ! vserver TESTSITE2 virtual
10.10.10.20 tcp https serverfarm SCA444 sticky 60 group
20 persistent rebalance inservice !--- The vserver for
the second site (www.testsite2.com) listens !--- to
10.10.10.10 on port 443. !--- Connections go to the SCAs
and change the !--- destination port to 444. (See the
configuration of server farm SCA444.) ! vserver WEB-
DECRYPT virtual 10.10.10.100 tcp 81 serverfarm WEBFARM
persistent rebalance inservice ! !--- This is the
vserver for the plain-text connections. !--- This
```

vserver receives connections on port 81 from the SCAs.
!--- As the configuration of this vserver does not specify a VLAN, !--- the vserver can also receive connections directly !--- from the client side. !--- To prevent direct client access of this VIP, !--- you can use the VLAN 4 option. !--- You can also place this VIP in the SCA subnetwork. In that case, !--- clients do not even have a route to that subnetwork. (Clients only !--- have a route if you configure the upstream router !--- with a static route.)

SCA 1

```
!--- This configures SCA in one-port, nontransparent mode. scal# show run
#
# Cisco CSCA Device Configuration File
#
# Written:      Sun Feb  6 01:46:35 2106
# Inxcfg:      version 2.3 build 200108071342
# Device Type:  CSS-SCA
# Device Id:    S/N 119cd6
# Device OS:    MaxOS version 2.5.1 build 200108071341
by Dan L. Reading

### Device ###

mode one-port
ip address 192.168.1.250 netmask 255.255.255.0
hostname scal
password enable
"2431245A572441713173717748626D734B35516B794F64336A51652
F"
no ip domain-name
no rdate-server
timezone "MST7MDT"
no rip
ip route 0.0.0.0 0.0.0.0 192.168.1.1 metric 1

### Interfaces ###

interface network
  auto
end
interface server
  auto
end

### Remote Management ###

no remote-management access-list
remote-management enable

### SNMP Subsystem ###

no snmp
telnet enable
no telnet access-list
web-mgmt enable
no web-mgmt access-list

### SSL Subsystem ###
```

```
ssl
  server test1 create
    ip address 10.10.10.100
    sslport 443
    remoteport 81
    key default
    cert default
    secpolicy default
    cachesize 20
    no transparent
  end
  server test2 create
    ip address 10.10.10.100
    sslport 444
    remoteport 81
    key default
    cert default
    secpolicy default
    cachesize 20
    no transparent
  end
end
sca1#
```

SCA 2

```
!--- This configures SCA in one-port, nontransparent mode. sca2# sca2# show run
#
# Cisco CSCA Device Configuration File
#
# Written:      Fri Feb 13 21:18:29 1970
# Inxcfg:      version 2.3 build 200108071342
# Device Type:  CSS-SCA
# Device Id:    S/N 119ca2
# Device OS:    MaxOS version 2.5.1 build 200108071341
by Dan L. Reading

### Device ###

mode one-port
ip address 192.168.1.251 netmask 255.255.255.0
hostname sca2
password enable
"2431245A572441713173717748626D734B35516B794F64336A51652
F"
no ip domain-name
no rdate-server
timezone "MST7MDT"
no rip
ip route 0.0.0.0 0.0.0.0 192.168.1.1 metric 1

### Interfaces ###

interface network
  auto
end
interface server
  auto
end
```

```

### Remote Management ###

no remote-management access-list
remote-management enable

### SNMP Subsystem ###

no snmp
telnet enable
no telnet access-list
web-mgmt enable
no web-mgmt access-list

### SSL Subsystem ###

ssl
  server test1 create
    ip address 10.10.10.100
    sslport 443
    remotepoint 81
    key default
    cert default
    secpolicy default
    cachesize 20
    no transparent
  end
  server test2 create
    ip address 10.10.10.100
    sslport 444
    remotepoint 81
    key default
    cert default
    secpolicy default
    cachesize 20
    no transparent
  end
end
sca2#

```

다음을 확인합니다.

이 섹션에서는 컨피그레이션이 제대로 작동하는지 확인하는 데 사용할 수 있는 정보를 제공합니다.

일부 **show** 명령은 [출력 인터프리터](#) 를에서 지원되는데(등록된 고객만), 이 틀을 사용하면 **show** 명령 출력의 분석 결과를 볼 수 있습니다.

```

!--- A client opens a connection to www.testsite1.com. Cat6k# show module csm 5 vserver detail
TESTSITE1, state = OPERATIONAL, v_index = 10
  virtual = 10.10.10.10/32:443, TCP, service = NONE, advertise = FALSE
  idle = 3600, replicate csrp = none, vlan = ALL, pending = 0
  max parse len = 600, persist rebalance = TRUE
  conns = 1, total conns = 1
  Default policy:
    server farm = SCA443
    sticky: timer = 60, subnet = 0.0.0.0, group id = 10
  Policy          Tot Conn      Client pkts  Server pkts
  -----
  (default)      1          9           11
!--- The client connection to port 443 hits the vserver TESTSITE1 !--- and is load balanced to

```

an SCA. TESTSITE2, state = OPERATIONAL, v_index = 11 virtual = 10.10.10.20/32:443, TCP, service = NONE, advertise = FALSE idle = 3600, replicate csrp = none, vlan = ALL, pending = 0 max parse len = 600, persist rebalance = TRUE conns = 0, total conns = 0 Default policy: server farm = SCA444 sticky: timer = 60, subnet = 0.0.0.0, group id = 20 Policy Tot Conn Client pkts Server pkts ----- (default) 0 0 0 WEB-DECRYPT, state = OPERATIONAL, v_index = 13 virtual = 10.10.10.100/32:81, TCP, service = NONE, advertise = FALSE idle = 3600, replicate csrp = none, vlan = 4, pending = 0 max parse len = 600, persist rebalance = TRUE conns = 1, total conns = 1 Default policy: server farm = WEBFARM sticky: timer = 0, subnet = 0.0.0.0, group id = 0 Policy Tot Conn Client pkts Server pkts ----- (default) 1 7 5 !--- The SCA opens a connection to 10.10.10.100 port 81, !--- which is load balanced to a web server. Cat6k# show module csm 5 conns detail

prot	vlan	source	destination	state
In TCP	4	192.168.1.250:4376	10.10.10.100:81	ESTAB
Out TCP	10	192.168.2.11:81	192.168.1.250:4376	ESTAB

vs = WEB-DECRYPT, ftp = No, csrp = False

!--- This provides details of the connection from the SCA to the web server. !--- The connection comes from VLAN 4 (the SCA VLAN), destined to !--- 10.10.10.100 port 81. !--- This is different from what happens in transparent mode. !--- In this case, the SCA opens the connections with use of !--- the SCA IP address, 192.168.1.250. The server does not see the IP !--- of the original client. !--- The connection goes to VLAN 10 (web servers VLAN) !--- to the web server selection. (The destination IP address !--- changes accordingly. The port does not change.) !--- If the servers listen to port 80 instead of port 81, you can configure !--- the translation of the destination port. You can add a port !--- to the definition of the real servers. !--- NOTE: The Out line swaps source with destination. !--- "Out" refers to the return traffic packets that the CSM !--- receives from that VLAN.

In TCP	6	10.15.0.50:2324	10.10.10.10:443	ESTAB
Out TCP	4	192.168.1.250:443	10.15.0.50:2324	ESTAB

vs = TESTSITE1, ftp = No, csrp = False

!--- This provides details of the connection from the client to the VIP. !--- The connection comes from VLAN 6 (the client VLAN), destined to !--- 10.10.10.10 port 443. !--- The connection goes to VLAN 4 (the SCA VLAN) !--- to the SCA selection. The destination IP changes !--- from the 10.10.10.10 (the VIP) to 192.168.1.250 (the SCA), !--- as the server farm had the option NAT server. !--- This is different in nontransparent mode. !--- The same client opens a second connection, !--- this time to www.testsite2.com. Cat6k# Cat6k# show module csm 5 conns detail

prot	vlan	source	destination	state
In TCP	4	192.168.1.250:4377	10.10.10.100:81	ESTAB
Out TCP	10	192.168.2.10:81	192.168.1.250:4377	ESTAB

vs = WEB-DECRYPT, ftp = No, csrp = False

!--- This connection is from SCA to VIP .100, load balanced to !--- web server .10. In TCP 4 192.168.1.250:4376 10.10.10.100:81 ESTAB Out TCP 10 192.168.2.11:81 192.168.1.250:4376 ESTAB vs = WEB-DECRYPT, ftp = No, csrp = False !--- This connection is from SCA to VIP .100, load balanced to !--- webserver .11. In TCP 6 10.15.0.50:2325 10.10.10.20:443 ESTAB Out TCP 4 192.168.1.250:444 10.15.0.50:2325 ESTAB vs = TESTSITE2, ftp = No, csrp = False !--- This connection is from client to VIP .20, load balanced to !--- SCA .250, port 444. In TCP 6 10.15.0.50:2324 10.10.10.10:443 ESTAB Out TCP 4 192.168.1.250:443 10.15.0.50:2324 ESTAB vs = TESTSITE1, ftp = No, csrp = False !--- This connection is from client to VIP .10, load balanced to !--- SCA .250, port 443. Cat6k#show module csm 5 real detail

192.168.2.10, WEBFARM, state = OPERATIONAL
 conns = 1, maxconns = 4294967295, minconns = 0
 weight = 8, weight(admin) = 8, metric = 0, remainder = 1
 total conns established = 1, total conn failures = 0
 192.168.2.11, WEBFARM, state = OPERATIONAL
 conns = 1, maxconns = 4294967295, minconns = 0
 weight = 8, weight(admin) = 8, metric = 0, remainder = 1
 total conns established = 1, total conn failures = 0
 192.168.1.250:443, SCA443, state = OPERATIONAL
 conns = 1, maxconns = 4294967295, minconns = 0
 weight = 8, weight(admin) = 8, metric = 0, remainder = 1
 total conns established = 1, total conn failures = 0

```

192.168.1.251:443, SCA443, state = OPERATIONAL
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0
192.168.1.250:444, SCA444, state = OPERATIONAL
  conns = 1, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 1
  total conns established = 1, total conn failures = 0
192.168.1.251:444, SCA444, state = OPERATIONAL
  conns = 0, maxconns = 4294967295, minconns = 0
  weight = 8, weight(admin) = 8, metric = 0, remainder = 0
  total conns established = 0, total conn failures = 0

```

*!--- This output shows that each web server has received a !--- connection. !--- The SCA .250 has received two connections, one to port 443 and !--- one to port 444. !--- The SCA .251 has not yet received any connection because !--- only two connections are open. One is open to each site !--- (10.10.10.10 and 10.10.10.20). A different port (443 or 444) !--- on the SCAs handles each site. The first !--- connection for each site goes to the first SCAs. !--- The following connection to either .10 or .20 goes to !--- .251, port 443 or 444, respectively. !--- This is SCA1 output. !--- There is one open connection. scal# **show netstat***

```

Pro State Recv-Q Send-Q Local Address          Remote Address
R-Win S-Win
-----
tcp ESTAB      0      0 192.168.1.250:443     10.15.0.50:2324
33580 16529
tcp ESTAB      0      0 192.168.1.250:4376    10.10.10.100:81
33304 17232
udp            0      0 *:4099                *: *
0      0
udp            0      0 *:4098                *: *
0      0
tcp LISTEN     0      0 *:2932                *: *
0      0
udp            0      0 *:2932                *: *
0      0
udp            0      0 *:520                 *: *
0      0
udp            0      0 *:514                 *: *
0      0
tcp LISTEN     0      0 *:444                 *: *
0      0
tcp LISTEN     0      0 *:443                 *: *
32768      0
tcp LISTEN     0      0 *:80                  *: *
0      0
tcp LISTEN     0      0 *:23                  *: *
0      0
scal#

```

*!--- There are two open connections. scal# **show netstat***

```

Pro State Recv-Q Send-Q Local Address          Remote Address
R-Win S-Win
-----
tcp ESTAB      0      0 192.168.1.250:444     10.15.0.50:2325
33580 16529
tcp ESTAB      0      0 192.168.1.250:443     10.15.0.50:2324
33580 16529
tcp ESTAB      0      0 192.168.1.250:4377    10.10.10.100:81
33304 17232
tcp ESTAB      0      0 192.168.1.250:4376    10.10.10.100:81
33304 17232
udp            0      0 *:4099                *: *
0      0
udp            0      0 *:4098                *: *
0      0
tcp LISTEN     0      0 *:2932                *: *

```



```
0      0
udp    0      0 *:2932      *: *
0      0
udp    0      0 *:520       *: *
0      0
udp    0      0 *:514       *: *
0      0
tcp LISTN 0      0 *:444       *: *
32768  0
tcp LISTN 0      0 *:443       *: *
32768  0
tcp LISTN 0      0 *:80        *: *
0      0
tcp LISTN 0      0 *:23        *: *
0      0
scal#
```

문제 해결

현재 이 컨피그레이션에 사용할 수 있는 특정 문제 해결 정보가 없습니다.