

Cómo configurar el CSS para equilibrar la carga de los servidores DNS y utilizar las señales de mantenimiento con secuencias de comando DNS.

Contenido

[Introducción](#)

[Diagrama de la red](#)

[Uso de señales de mantenimiento en secuencias de comando DNS](#)

[Secuencia de comandos ap-kal-dns de CSS](#)

[Configuración](#)

[CSS11150 estructura 8 de WebNS que se ejecuta 4.01](#)

[Resultados del comando show](#)

[show keepalive](#)

[show keepalive-summary](#)

[show service summary](#)

[Información Relacionada](#)

Introducción

Con las versiones 4.0 y posteriores del software Cisco IOS, los clientes pueden utilizar señales de mantenimiento en secuencia de comandos para servicios no estándar o especializados, como el Sistema de nombres de dominio (DNS).

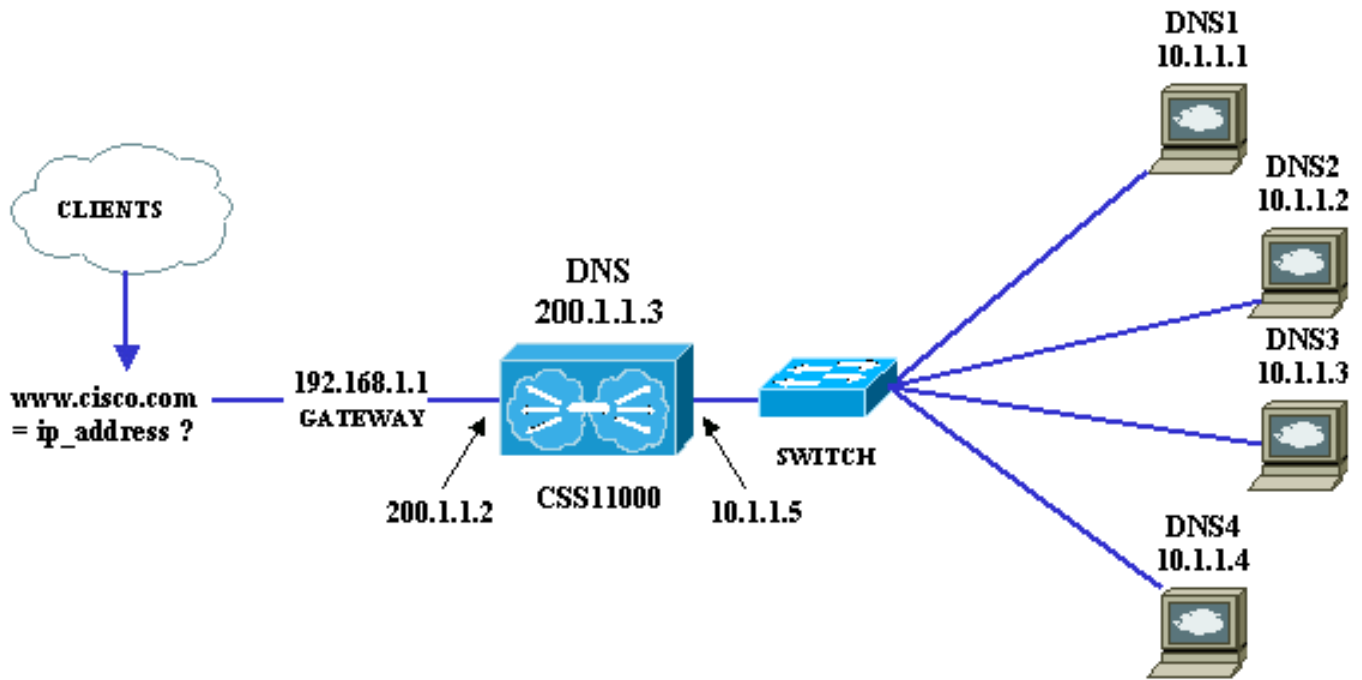
Como las consultas de DNS se basan en el Protocolo de datagrama de usuario (UDP), usted debe configurar un grupo de origen en el Switch de servicios de contenido (CSS) para que parezca que las respuestas del servidor DNS provienen de la misma dirección de la cual se enviaron las consultas originalmente (probablemente, la dirección virtual IP (VIP)). Al activar este grupo de origen, los keepalive con secuencias de comandos de DNS no funcionan ya que la respuesta a la consulta keepalive pasa a través del servicio de Conversión de direcciones de red (NAT), lo que hace que el CSS reciba una respuesta de una dirección IP distinta de aquella a la cual usted dirigió originalmente el pedido.

Esta configuración fue desarrollada y probada utilizando las versiones de software y hardware indicadas a continuación.

- Todas las plataformas CSS (CSS11000 y CSS11500)
- Software Release 5.0 y Posterior de Cisco WebNS (los keepalives en secuencia de comandos no fueron agregados hasta la versión 4.0)

La información de este documento se originó a partir de dispositivos dentro de un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener un comando antes de ejecutarlo.

Diagrama de la red



Uso de señales de mantenimiento en secuencias de comando DNS

Consejos para utilizar señales de mantenimiento secuenciadas por DNS:

- El acceso a los servidores de raíz del Sistema de nombres de dominio (DNS) de Internet se requiere para la instrumentación satisfactoria de los keepalives en secuencia de comandos DNS.
- La dirección IP de servicio no es portadora del servicio de DNS en absoluto. Puede ingresarse cualquier dirección y esto no afecta el estado; sin embargo, debe ingresarse alguna dirección o de lo contrario aparecerá un error de "dirección IP incorrecta".
- El argumento para la secuencia de comando debe ser la dirección IP o el nombre del host del servidor DNS que desea controlar. Típicamente es la dirección IP configurada en el servicio.
- [La secuencia de comandos está codificada por software para resolver www.cisco.com.](#) No importa si el servidor DNS puede resolver este direccionamiento o no, mientras se vuelva una respuesta de DNS que el servicio está vivo. Este keepalive está probando solamente si un servidor DNS puede responder a una interrogación, no si puede resolver un nombre específico. Las interrogaciones del script del CSS a los servidores internos DNS no pierden de vista la Disponibilidad de los servidores DNS.

Secuencia de comandos ap-kal-dns de CSS

```
!no echo
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Filename: ap-kal-dns
! Parameters: DNS_Server
!
! Description: !--- This script resolves a domain name from a specific DNS
!--- server. This builds a UDP packet based on RFC 1035 . !
```

```

! Failure Upon: !--- Not resolving the hosts's IP from the domain name.
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! if ${ARGS}[#] "NEQ" "1" echo
"Usage: ap-kal-dns \'Hostname\'" exit script 1 endbranch ! Defines: set HostName "${ARGS}[1]" !
Connect to the remote host set EXIT_MSG "Connection failed" socket connect host ${HostName} port
53 udp !--- This may require a little explanation. Since we just want to see
!--- if the DNS server is alive, we send a simple DNS Query. This
!--- query is hard-coded in hexadecimal and sent raw to the DNS server.
!--- The DNS request has a 12-byte header (as seen for the first 12 bytes
!--- of hex) and then a DNS name (for example, www.cisco.com).
!--- Lastly, it follows with some null termination and a few bytes
!--- representing query type. !--- See RFC 1035 for more information. set EXIT_MSG "Send:
failure" socket send ${SOCKET} "000201000001000000000000003777777057961686f6f03636f6d00000 10001"
raw !--- Receive an unexplained response, but it is not important because
!--- an unstable DNS server or a non-existent one would probably not send
!--- back any data at all. set EXIT_MSG "Receive: Failed to receive data" socket waitfor
${SOCKET} "cisco" 4000 no set EXIT_MSG socket disconnect ${SOCKET} exit script 0

```

Si los servidores DNS no pueden responder a una interrogación, después usted necesita crear un ACL con esta lógica:

- Cualquier cosa originado a partir del uno de los servidores DNS destinados para el CSS (IP Address del Virtual LAN (VLAN) del circuito) desvía todas las reglas de contenido y grupos fuente.
- Otro tráfico originado desde los servidores DNS pasa a través del grupo de origen configurado.

Configuración

CSS11150 estructura 8 de WebNS que se ejecuta 4.01

```

!***** GLOBAL *****
ip redundancy
no restrict xml
username predictive des-password xecchdhdhnglhueig5csfbe4fievhjg
username admin des-password uezfqq6eoeic3e2d superuser
acl enable
ip route 0.0.0.0 0.0.0.0 200.1.1.1 1

!***** INTERFACE *****

interface ethernet-1
bridge vlan 2
phy 100Mbps-FD
interface ethernet-2
bridge vlan 3
phy 100Mbps-FD

!***** CIRCUIT *****

circuit VLAN2
redundancy
ip address 10.1.1.5 255.255.255.0
circuit VLAN3
redundancy
ip address 200.1.1.2 255.255.255.0

!***** SERVICE *****

service DNS1

```

```

ip address 10.1.1.1
keepalive type script ap-kal-dns "10.1.1.1"
active
service DNS2
ip address 10.1.1.2
keepalive type script ap-kal-dns "10.1.1.2"
active
service DNS3
ip address 10.1.1.3
keepalive type script ap-kal-dns "10.1.1.3"
active
service DNS4
ip address 10.1.1.4
keepalive type script ap-kal-dns "10.1.1.4"
active
service Router1
ip address 200.1.1.1
type redundancy-up
active

!***** OWNER *****

owner L3_Owner
content L3_Rule
vip address 200.1.1.3
add service DNS1
add service DNS2
add service DNS3
add service DNS4
active

!***** GROUP *****

group dns
vip address 200.1.1.3
active

!***** ACL *****

acl 20
clause 10 permit any any destination any
apply circuit-(VLAN3)
acl 10
clause 10 bypass any 10.1.1.1 255.255.255.255 destination 10.1.1.5 255.255.255.255
clause 20 bypass any 10.1.1.2 255.255.255.255 destination 10.1.1.5 255.255.255.255
clause 30 bypass any 10.1.1.3 255.255.255.255 destination 10.1.1.5 255.255.255.255
clause 40 bypass any 10.1.1.4 255.255.255.255 destination 10.1.1 5 255.255.255.255
clause 50 permit any 10.1.1.0 255.255.255.0 destination any sourcegroup dns
clause 60 permit any any destination any
apply circuit-(VLAN2)

```

Resultados del comando show

Ejemplos de salidas del comando show:

show keepalive

```

52-css150-4# show keepalive Keepalives: Name: AUTO_nexthop00002 Index: 0 State: Alive
Description: Auto generated for service nexthop00002 Address: 200.1.1.1 Port: Any Type: ICMP
Frequency: 5 Max Failures: 3 Retry Frequency: 5 Dependent Services: nexthop00002 Name: AUTO_DNS1
Index: 1 State: Down Description: Auto generated for service DNS1 Address: 10.1.1.1 Port: Any

```

```
Type: SCRIPT ap-kal-dns Script Arguments: "10.1.1.1" Script Error: Script error in line: 41 !---
Note: This service has no access to root servers, which causes the error.
Script Run Time: 4 seconds Frequency: 5 Max Failures: 3 Retry Frequency: 5 Dependent Services:
DNS1 Name: AUTO_DNS3 Index: 2 State: Alive Description: Auto generated for service DNS3 Address:
10.1.1.3 Port: Any Type: SCRIPT ap-kal-dns Script Arguments: "10.1.1.3" Script Error: None !---
Note: This service has access to Internet root servers. Script Run Time: 0 seconds Frequency: 5
Max Failures: 3 Retry Frequency: 5 Dependent Services: DNS3 Name: AUTO_DNS4 Index: 3 State:
Alive Description: Auto generated for service DNS4 Address: 10.1.1.4 Port: Any Type: SCRIPT ap-
kal-dns Script Arguments: "10.1.1.4" Script Error: None Script Run Time: 0 seconds Frequency: 5
Max Failures: 3 Retry Frequency: 5 Dependent Services: DNS4 Name: AUTO_Router1 Index: 4 State:
Down Description: Auto generated for service Router1 Address: 200.1.1.1 Port: Any Type: ICMP
Frequency: 5 Max Failures: 3 Retry Frequency: 5 Dependent Services: Router1 Name: AUTO_DNS2
Index: 5 State: Down Description: Auto generated for service DNS2 Address: 10.1.1.2 Port: Any
Type: SCRIPT ap-kal-dns Script Arguments: "10.1.1.2" Script Error: Script error in line: 41
Script Run Time: 4 seconds Frequency: 5 Max Failures: 3 Retry Frequency: 5 Dependent Services:
DNS2 52-css150-4#
```

show keepalive-summary

```
52-css150-4# show keepalive-summary Keepalives: AUTO_nexthop00002 State: Alive 200.1.1.1
AUTO_DNS1 State: Down 10.1.1.1 AUTO_DNS3 State: Alive 10.1.1.3 AUTO_DNS4 State: Alive 10.1.1.4
AUTO_Router1 State: Down 200.1.1.1 AUTO_DNS2 State: Down 10.1.1.2
```

show service summary

```
52-css150# show service summary Service Name State Conn Weight Avg State Load Transitions
AUTO_DNS1 Down 0 1 2 0 AUTO_DNS3 Alive 0 1 2 1 AUTO_DNS4 Alive 0 1 255 1 AUTO_DNS2 Down 0 1 255
0
```

Información Relacionada

- [Soporte de productos del Entrega de contenido y video](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)