Solución de problemas de inicio de sesión en ASR5500 debido a sesiones noTTY inactivas

Contenido

Introducción

Este documento describe cómo resolver problemas en situaciones en las que se pierde la conectividad de Secure Shell (SSH) en las IP de administración del router de servicios de agregación (ASR5500/ASR 5000).

Problemas de inicio de sesión en nodos ASR5500

No puede iniciar sesión en los nodos de núcleo de paquetes ASR5500. La conexión SSH se termina inmediatamente sin el mensaje de inicio de sesión. Las conexiones Telnet muestran un comportamiento similar.

Pasos para Solucionar Problemas

Paso 1. Intente iniciar sesión en el nodo a través de la conexión de consola.

Paso 2. En la mayoría de los casos, no se emiten trampas específicas del protocolo simple de administración de red (SNMP) que puedan señalar la causa de la falla de conexión.

Paso 3. Los registros relacionados con el login, constantemente presentes en los syslogs son:

```
evlogd: [local-60sec55.607] [tacacs+ 37201 error] [5/0/8908 <vpnmgr:1> authen_events.c:598]
[software internal system critical-info syslog] Authentication failed for user **** on tty
/dev/pts/0, application ssh, remote IP address XX.XX.XX.XX
evlogd: [local-60sec55.623] [cli 30028 debug] [5/0/8908 <vpnmgr:1> luser_auth.c:1448] [context:
local, contextID: 1]  [software internal system syslog] Login attempt failure for user **** IP
address XX.XX.XX.XX - Access type ssh/sftp
evlogd: [local-60sec53.652] [tacacs+ 37201 error] [5/0/8908 <vpnmgr:1> authen_events.c:598]
[software internal system critical-info syslog] Authentication failed for user **** on tty
/dev/pts/0, application ssh, remote IP address XX.XX.XX.XX
evlogd: [local-60sec53.679] [cli 30028 debug] [5/0/8908 <vpnmgr:1> luser_auth.c:1448] [context:
local, contextID: 1]  [software internal system syslog] Login attempt failure for user **** IP
address XX.XX.XX.XX - Access type ssh/sftp#####
evlogd: [local-60sec2.942] [tacacs+ 37201 error] [5/0/8908 <vpnmgr:1> authen_events.c:598]
[software internal system critical-info syslog] Authentication failed for user epcats on tty
/dev/pts/0, application ssh, remote IP address YY.YY.YY.YY
```

Paso 4. El comando **show crash list all** muestra las caídas recientes, tenga en cuenta que las relacionadas con **vpnmgr** son especialmente importantes.

Paso 5. El comando **show task resources all** garantiza que los procesos **vpnmgr** y **sshd** no deben estar en estado excesivo. **vpnmgr** es responsable de la administración del conjunto de direcciones IP y realiza todas las operaciones específicas del contexto. **sshd** admite el inicio de sesión seguro en la CLI de StarOS.

Paso 6. El reinicio de la instancia 1 **de vpnmgr**. ayuda a recuperar la conexión SSH con un impacto mínimo en algunos casos. Sin embargo, la conexión podría terminar después de un tiempo.

Paso 7. El switchover MIO resuelve el problema. Tenga en cuenta que en los escenarios donde un proceso puede alcanzar un valor de umbral o un estado de sobrecarga, el rebote MIO puede ayudar a borrarlo.

La solución alternativa es el switchover MIO. La siguiente sección habla sobre los pasos para el análisis de la causa raíz.

Análisis de la causa raíz

1. Utilice el comando **show administrator** para determinar el número de conexiones activas en el nodo. Sin embargo, es posible que el resultado no muestre un número excesivo de sesiones activas que podrían haber obstruido las conexiones con el nodo.

Ejemplo de resultado

```
[local]ASR5500-2# show administrators
Monday September 06 13:15:07 CDT 2021
Administrator/Operator Name    M Type    TTY            Start Time               Mode
Idle

------------------------------ - ------- -------------- ------------------------ ------------ --
--

admin                            admin   /dev/pts/4     Mon Sep 06 13:14:38 2021 Context User 29

admin                            admin   /dev/pts/3     Mon Sep 06 12:21:13 2021 Context User
749

admin                            admin   /dev/pts/2     Thu Sep 02 11:03:57 2021 Context User
342206
[local]ASR5500-2#
```

2. Además, ejecute estos comandos e indíquenos en el problema. Navegue hasta el shell de depuración a través del modo oculto.

```
cli test-command pass <password>
debug shell
```

Ejecute estos comandos en el comando debug shell:

```
ps -ef
```

```
setvr 1 bash
netstat -n
```

ps: lista de procesos. El comando **ps** le permite ver información técnica sobre los procesos actuales en un sistema, así como verificar su estado.

-e: muestra todos los procesos, independientemente del usuario.

-f: muestra los procesos en formato detallado.

El **comando netstat** es una de las opciones de línea de comandos más convenientes que se utiliza para mostrar todas las conexiones de socket presentes en el nodo. Posee la capacidad de enumerar todas las conexiones del socket tcp y udp, así como las conexiones unix. Esta CLI también se puede utilizar para enumerar los posibles zócalos de escucha que aún pueden esperar a que se establezca una conexión.

Ejemplo de resultado

```
ASR5500-2:card5-cpu0# ps -eF

UID        PID  PPID  C    SZ   RSS PSR STIME TTY          TIME CMD

root         1     0  0   511   640   4 Aug20 ?        00:00:13 init [5]

root         2     0  0     0     0   2 Aug20 ?        00:00:00 [kthreadd]

root         3     2  0     0     0   0 Aug20 ?        00:00:00 [ksoftirqd/0]

root         6     2  0     0     0   0 Aug20 ?        00:00:00 [migration/0]

root         7     2  0     0     0   0 Aug20 ?        00:00:01 [watchdog/0]

root         8     2  0     0     0   1 Aug20 ?        00:00:00 [migration/1]

root        10     2  0     0     0   1 Aug20 ?        00:00:00 [ksoftirqd/1]

root        11     2  0     0     0   0 Aug20 ?        00:00:31 [kworker/0:1]

root        12     2  0     0     0   1 Aug20 ?        00:00:00 [watchdog/1]

root        13     2  0     0     0   2 Aug20 ?        00:00:00 [migration/2]

root        15     2  0     0     0   2 Aug20 ?        00:00:00 [ksoftirqd/2]

root        16     2  0     0     0   2 Aug20 ?        00:00:00 [watchdog/2]

root        17     2  0     0     0   3 Aug20 ?        00:00:00 [migration/3]

root        19     2  0     0     0   3 Aug20 ?        00:00:00 [ksoftirqd/3]

root        20     2  0     0     0   3 Aug20 ?        00:00:00 [watchdog/3]

root        21     2  0     0     0   4 Aug20 ?        00:00:00 [migration/4]

root        22     2  0     0     0   4 Aug20 ?        00:00:00 [kworker/4:0]

root        23     2  0     0     0   4 Aug20 ?        00:00:00 [ksoftirqd/4]
```

.....

```
ASR5500-2:card5-cpu0# setvr 1 bash
bash-2.05b# netstat -n

Active Internet connections (w/o servers)

Proto Recv-Q Send-Q Local Address           Foreign Address         State

tcp        0      0 10.201.211.23:22        10.227.230.222:51781    ESTABLISHED

tcp        0      0 10.201.211.23:22        10.24.28.55:49918       ESTABLISHED

tcp        0      0 10.201.211.23:22        10.99.10.148:54915      ESTABLISHED

tcp        0      0 10.201.211.23:22        10.227.230.222:51783    ESTABLISHED

Active UNIX domain sockets (w/o servers)

Proto RefCnt Flags       Type       State         I-Node Path

unix  2      [ ]         DGRAM                    39221385

unix  2      [ ]         DGRAM                     27056

bash-2.05b# exit
```

Según el informe mencionado anteriormente, los servidores ejecutaban scripts que generaban conexiones al ASR55K box. Estos servidores abrieron muchas de estas conexiones que se encontraban en estado de inactividad o de inactividad, pero nunca se cerraron.

Incluso después de finalizar la conexión TeleTypeWriter (TTY), la conexión TCP permaneció activa en nuestras puertas de enlace.

Como resultado de estas conexiones, el ASR5500 alcanzó el número máximo de conexiones SSH permitidas, obstruyendo la conexión a la caja. Tan pronto como intente iniciar sesión en los servidores y matar los procesos principales, todas las conexiones se liberan instantáneamente y el SSH se restaura inmediatamente.

Estas conexiones SSH inactivas se establecen como conexiones TeleTypeWriter (noTTY). Los programas que están conectados utilizan estas conexiones noTTY de forma que no se muestre su resultado.

Comandos como SSH admin@asr55k hostname "display version" establece una conexión noTTY en la mayoría de los casos.

De manera similar, las declaraciones como SSH: *@notty indica que hay inicios de sesión de SSH en nuestros Gateways (GW) a los que no se les ha asignado un terminal visual, como un shell o pseudo-terminal. Esto puede ocurrir durante diversas operaciones relacionadas con secuencias de comandos, especialmente cuando se utilizan conexiones FTP/Secure Copy (SCP).

Solución propuesta

1. Implemente un tiempo de espera en las secuencias de comandos que se pueden utilizar para los servidores API. Varias conexiones SSH que ejecutan varias CLI pueden generar congestión de mensajeros y un uso significativo de la CPU en todos los procesos de sessmgr.

2. Para facilitar la resolución de problemas, configure esta opción:

```
logging filter runtime facility cli level debug critical-info
```

3. Aplique esta configuración al nodo. Este comando se utiliza para terminar sesiones SSH inactivas después de 5 minutos. Esto se utiliza como mecanismo de protección contra las sesiones obsoletas causadas por el servidor:

```
Exec > Global Configuration > Context Configuration
configure > context context_name
administrator encrypted password timeout-min-absolute 300 timeout-min-idle 300
```

Información Relacionada

- información CLI
- Guías de configuración de la serie ASR 5000 de Cisco
- **Soporte Técnico y Documentación - Cisco Systems**