

Solucionar problemas de ejecución de script Diagnostics.sh incompleta

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[Problema](#)

[Solución](#)

Introducción

Este documento describe el procedimiento para resolver problemas de ejecución incompleta del script diagnostics.sh en Cisco Policy Suite (CPS).

Colaboración de Ullas Kumar E, ingeniero del TAC de Cisco.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Linux
- CPS

Nota: Cisco recomienda que tenga acceso raíz privilegios a CLI de CPS.

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- CPS 21.1
- Centos 8.0
- Unified Computing System (UCS)-B

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Antecedentes

Diagnostics.sh es el comando básico de solución de problemas que se puede ejecutar en el nodo pcrfclient o installer de CPS para verificar el estado actual del sistema.

Proporciona una lista detallada de parámetros como parte de la comprobación de estado de CPS.

Este script se ejecuta en los distintos puntos de acceso, supervisión y configuración de un sistema CPS en ejecución.

En entornos de alta disponibilidad (HA) o georredundante (GR), el script siempre realiza una comprobación de ping para todas las máquinas virtuales (VM) antes de cualquier otra comprobación y agrega cualquier comprobación que no supere la prueba de ping a la variable IGNORED_HOSTS. Esto ayuda a reducir la posibilidad de errores de función de script.

Examples:

```
/var/qps/bin/diag/diagnostics.sh -q  
/var/qps/bin/diag/diagnostics.sh --basic_ports --clock_skew
```

Estas son las comprobaciones destacadas que realiza este script.

```
--basic_ports : Run basic port checks  
For AIO: 80, 11211, 27017, 27749, 7070, 8080, 8090, 8182, 9091, 9092  
For HA/GR: 80, 11211, 7070, 8080, 8081, 8090, 8182, 9091, 9092, and Mongo DB ports based on /etc/broadh  
--clock_skew : Check clock skew between lb01 and all vms (Multi-Node Environment only)  
--diskspace : Check diskspace  
--get_active_alarms : Get the active alarms in the CPS  
--get_frag_status : Get fragmentation status for Primary members of DBs viz. session_cache, sk_cache, c  
--get_replica_status : Get the status of the replica-sets present in environment. (Multi-Node Environme  
--get_shard_health : Get the status of the sharded database information present in environment. (Multi-  
--get_sharding_status : Get the status of the sharding information present in environment. (Multi-Node  
--get_session_shard_health : Get the session shard health status information present in environment. (M  
--get_peer_status : Get the diameter peer information present in environment. (Multi-Node Environment c  
--get_sharded_replica_status : Get the status of the shards present in environment. (Multi-Node Environ  
--ha_proxy : Connect to HAProxy to check operation and performance statistics, and ports (Multi-Node En  
    http://lbvip01:5540/haproxy?stats  
    http://lbvip01:5540//haproxy-diam?stats  
--help -h : Help - displays this help  
--hostnames : Check hostnames are valid (no underscores, resolvable, in /etc/broadhop/servers) (AIO onl  
--ignored_hosts : Ignore the comma separated list of hosts. For example --ignored_hosts='portal01,porta  
    Default is 'portal01,portal02,portallb01,portallb02' (Multi-Node Environment only)  
--ping_check : Check ping status for all VM  
--policy_revision_status : Check the policy revision status on all QNS, LB, UDC VMs.  
--lwr_diagnostics : Retrieve diagnostics from CPS LWR kafka processes  
--qns_diagnostics : Retrieve diagnostics from CPS java processes  
--qns_login : Check qns user passwordless login  
--quiet -q : Quiet output - display only failed diagnostics  
--radius : Run radius specific checks  
--redis : Run redis specific checks  
--whisper : Run whisper specific checks  
--aido : Run Aido specific checks  
--svn : Check svn sync status between pcrfclient01 & pcrfclient02 (Multi-Node Environment only)  
--tacacs : Check Tacacs server reachability  
--swapspace : Check swap space  
--verbose -v : Verbose output - display *all* diagnostics (by default, some are grouped for readability)  
--virtual_ips : Ensure Virtual IP Addresses are operational (Multi-Node Environment only)  
--vm_allocation : Ensure VM Memory and CPUs have been allocated according to recommendations
```

Problema

Es posible que en algunas situaciones, la ejecución de los scripts `diagnostics.sh` se bloquee en un punto y no pueda avanzar ni finalizar la ejecución del script.

Puede ejecutar la secuencia de comandos y observar que la secuencia de comandos está bloqueada en la opción "Comprobar siEl estado de las operaciones inteligentes automáticas de base de datos (AIDO) no continúa con la comprobación del número de subversión (SVN) ni con otras comprobaciones.

```
<#root>

[root@installer ~]#

diagnostics.sh

CPS Diagnostics HA Multi-Node Environment
-----
Ping check for all VMs...
Hosts that are not 'pingable' are added to the IGNORED_HOSTS variable...[PASS]
Checking basic ports for all VMs...[PASS]
Checking qns passwordless logins for all VMs...[PASS]
Validating hostnames...[PASS]
Checking disk space for all VMs...[PASS]
Checking swap space for all VMs...[PASS]
Checking for clock skew for all VMs...[PASS]
Retrieving diagnostics from pcrfclient01:9045...[PASS]
Retrieving diagnostics from pcrfclient02:9045...[PASS]
Checking redis server instances status on lb01...[PASS]
Checking redis server instances status on lb02...[PASS]
Checking whisper status on all VMs...[PASS]
Checking AIDO status on all VMs...[PASS]
.
.
```

Cuando usted verifica el resultado detallado de `diagnostics.sh`, hay un paso para verificar el estado de SVN, el script no se mueve más allá de allí. Indica que la secuencia de comandos `diagnostics.sh` se ha atascado en la comprobación de factores.

```
<#root>

[[32mPASS[0m] AIDO Pass
[[ -f /var/tmp/aido_extra_info ]]
cat /var/tmp/aido_extra_info
There is no provision to check AIDO service status of installer from this host
/bin/rm -fr /var/tmp/aido_extra_info
check_all_svn
++ is_enabled true
++ [[ ' ' == \t\r\u\e ]]
++ [[ true != \f\a\l\s\e ]]
++ echo true
[[ true == \t\r\u\e ]]
++ awk '{ $1=""; $2=""; print }'

++ /usr/bin/ssh root@pcrfclient01 -o ConnectTimeout=2 /usr/bin/facter.
```

```
++ grep svn_slave_list
```

El script inicia sesión en pcrfclient01 y verifica la **svn_slave_list** desde el resultado del comando **factor**, que no se ejecuta completamente.

Además, puede iniciar sesión en pcrfclient01 y verificar si el comando factor se ejecuta correctamente y da el resultado deseado.

```
<#root>
[root@pcrfclient01 ]#
factor | grep eth

[root@installer ~]# ^C
```

Cuando verifica el promedio de carga de pcrfclient01, se observa como muy alto.

```
<#root>
[root@pcrfclient01 pacemaker]#
top

top - 15:34:18 up 289 days, 14:55, 1 user, load average:
2094.68, 2091.77, 2086.36
```

Verifique que los procesos relacionados con factores estén atascados y resulten en una carga promedio alta.

```
<#root>
[root@pcrfclient01 ~]#
ps -ef | grep factor | wc -l

2096
```

Solución

La solución definitiva para eliminar estos procesos atascados y reducir el promedio de carga es reiniciar la máquina virtual pcrfclient01. El procedimiento para borrar los procesos atascados de factor y resolver el problema de bloqueo de la ejecución de diagnostics.sh:

Paso 1. Inicie sesión en el nodo pcrfclient y ejecute el comando reboot.

```
<#root>
```

```
[root@pcrfclient01 ~]#
```

```
init 6
```

Paso 2. Verifique que la máquina virtual pcrfclient01 esté activa y estable.

```
<#root>
```

```
[root@pcrfclient01 ~]#
```

```
uptime
```

```
10:07:15 up 1 min, 4:09, 1 user, load average: 0.33, 0.33, 0.36
```

```
[root@pcrfclient01 ~]#
```

Paso 3. Verifique que el promedio de carga de pcrfclient01 sea normal.

```
<#root>
```

```
[root@instapcrfclient01l1er ~]#
```

```
top
```

```
top - 10:07:55 up 1 min, 4:10, 1 user, load average: 0.24, 0.31, 0.35
```

Paso 4. Ejecute diagnostics.sh y verifique que la ejecución del script haya finalizado.

```
<#root>
```

```
[root@instapcrfclient01l1er ~]#
```

```
diagnostics.sh
```

Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).