

# 对不完整的Diagnostics.sh脚本执行进行故障排除

## 目录

---

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[问题](#)

[解决方案](#)

---

## 简介

本文档介绍对思科策略套件(CPS)中不完整的diagnostics.sh脚本执行进行故障排除的过程。

作者：Ullas Kumar E，思科TAC工程师。

## 先决条件

### 要求

Cisco 建议您了解以下主题：

- Linux
- CPS

注意：Cisco建议您必须具有根访问权限 权限 到CPS CLI。

### 使用的组件

本文档中的信息基于以下软件和硬件版本：

- CPS 21.1
- Centos 8.0
- 统一计算系统(UCS)-B

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

## 背景信息

Diagnostics.sh是基本的故障排除命令，可以在pcrfclient或CPS的安装程序节点中执行，以验证系统的当前状态。

它提供详细的参数列表，作为CPS运行状况检查的一部分。

此脚本针对运行CPS系统的各种访问、监控和配置点运行。

在高可用性(HA)或地区冗余(GR)环境中，该脚本始终先对所有虚拟机(VM)执行ping检查，然后再执行任何其他检查，并将未通过ping测试的所有虚拟机添加到IGNORED\_HOSTS变量。这有助于降低脚本函数错误的可能性。

Examples:

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

以下是此脚本执行的突出检查。

```
--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/broad  
--clock_skew : Check cclock 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 Environm  
--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  
--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 E  
    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 on  
--ignored_hosts : Ignore the comma separated list of hosts. For example --ignored_hosts='portal01,port  
    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 readabilit  
--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
```

## 问题

在某些情况下，执行diagnostics.sh脚本时可能会挂起，无法进一步移动或完成脚本执行。

您可以执行脚本，并观察脚本卡在“正在检查Auto Intelligent DB Operations(AIDO)Status“(自动智能数据库操作(AIDO)状态”不继续执行Subversion Number(SVN)检查和进一步操作。

```
<#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]  
.  
.
```

当您检查diagnostics.sh的详细输出时，有一个步骤可以检查SVN状态，脚本不会进一步执行下去。它表示diagnostics.sh脚本在出厂检查时卡住了。

```
<#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\s ]]  
++ 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
```

脚本登录到pcrfclient01并从factor命令输出中检查svn\_slave\_list，该命令未完全执行。

此外，您可以登录到pcrfclient01，并检查factor命令是否正确运行并提供所需的输出。

```
<#root>
```

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

```
factor | grep eth
```

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

当您检查pcrfclient01的负载平均值时，发现它非常高。

```
<#root>
```

```
[root@pcrfclient01 pacemaker]#
```

```
top
```

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

```
2094.68, 2091.77, 2086.36
```

验证与因素相关的进程是否停滞并导致高平均负载。

```
<#root>
```

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

```
ps -ef | grep factor | wc -l
```

```
2096
```

## 解决方案

清除这些停滞进程并降低平均负载的最终解决方案是重新启动pcrfclient01 VM。清除因素滞留进程和解决diagnostics.sh执行挂起问题的过程：

步骤1:登录到pcrfclient节点并执行重新启动命令。

```
<#root>
```

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

```
init 6
```

第二步：验证pcrfclient01 VM已启动且稳定。

```
<#root>
```

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

```
uptime
```

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

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

第三步：验证pcrfclient01的平均负载是否正常。

```
<#root>
```

```
[root@instapcrfclient0111er ~]#
```

```
top
```

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

第四步：运行diagnostics.sh并验证脚本执行已完成。

```
<#root>
```

```
[root@instapcrfclient0111er ~]#
```

```
diagnostics.sh
```

## 关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。