

# 对UC服务器中的僵尸/已失效进程进行故障排除

## 目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[使用UCOS管理CLI检查僵尸计算机](#)

[手动排除/清除僵尸](#)

[重新启动适当的服务](#)

[重启服务器](#)

[终止父进程](#)

[验证](#)

## 简介

本文档介绍如何使用管理CLI登录时在CUCM、IMnP和其他思科统一通信产品上看到的僵尸进程。

## 先决条件

### 要求

思科建议您了解使用统一通信服务器的管理CLI:

- 思科统一通信管理器 (CUCM)
- 思科统一即时消息和在线状态服务器(IMnP)
- 思科Unity Connection服务器(CUC)

### 使用的组件

本文档不限于特定的软件和硬件版本。

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

统一通信服务器本质上是基于Linux操作系统的应用。当进程在Linux上失效时,并非所有进程都立即从内存中删除,其进程描述符(PID)保留在内存中,只占用很少的内存。此进程将成为失效进程,并通知进程的父进程其子进程已死亡。然后,父进程应读取死进程的退出状态并将其从内存中完全删除。使用wait()系统调用完成此操作后,即会从进程表中排除僵尸进程。这称为获取僵尸进程。这通常发生得非常快,因此您不会看到僵尸进程在系统上累积。

但是，有时父进程不执行wait()信号调用，子进程将保留在内存中，直到清理完毕。换句话说，僵尸进程是执行已完成但仍在进程表中有一个条目的进程，因为父进程仍需要读取其子进程的退出状态。

## 使用UCOS管理CLI检查僵尸计算机

在CLI中，可使用命令show process load检查僵尸的存在。

```
14 admin:show process load
15 admin:show process load
16 top - 08:43:47 up 48 days, 4:20, 1 user, load average: 9.86, 6.17, 4.17
17 Tasks: 879 total, 1 running, 861 sleeping, 0 stopped, 17 zombie
18 Cpu(s): 40.9%us, 10.2%sy, 0.4%ni, 48.1%id, 0.1%wa, 0.0%hi, 0.2%si, 0.0%st
19 Mem: 8062228k total, 7888104k used, 174124k free, 78128k buffers
20 Swap: 4095996k total, 2891264k used, 1204732k free, 2368392k cached
21 PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
22 20038 xcpuser 20 0 813m 627m 21m S 18.9 8.0 4756:46 iabherd
```

## 手动排除/清除僵尸

除前面提到的用于保存PID的微小内存外，僵尸进程不使用任何系统资源，但它们保留其进程ID。在UC服务器中，提供给系统的内存很大，因此系统由于僵尸计算机的存在而耗尽其他进程的PID的可能性很小。

因此，僵尸计算机可以留在系统上，在下次系统重新启动时自动清除僵尸计算机。

但是，如果系统中有清除僵尸计算机的要求，您可以执行特定操作

## 重新启动适当的服务

需要确定相关流程，从而找出泄漏子流程的服务。

1. 从CLI输出中，获取show process list和show process list detail的输出结果。

```
3476 admin:show process list detail
3477 PID ARGS
3478 PID PPID USER COMMAND
-----
3765 admin:show process list
3766 PID ARGS
3767 PID COMMAND
```

2. 在文本编辑器中复制输出并搜索文本“已失效”。
3. 记下这些已失效进程的进程ID(pid)和父进程ID(ppid)。
4. 在文档中跟踪ppid以查找相关流程。

### 示例 1

CUCM:当我搜索文件中的文本“defent”时，我看到有个PID 22908已失效。

```
664 22908 29815 513 iprod <defunct> 0.0
```

该PID的ppid是29815。在本文中29815的跟踪中，我发现该过程与AMC服务有关。

```
3664: 22908 29815 513 iprod <defunct> 0.0 0.0 1231 0 0 0 Sun
3740: 29815 29812 513 amc 0.8 1.5 6877007 102 125112 83187
3955: 29815 | \_ /usr/local/cm/bin/amc /usr/local/cm/conf/amc/amcCfg.xml
```

解决方案 — 在此节点上重新启动AMC(警报管理器和收集器服务)可清除僵尸。

### 示例 2

CUCM:当搜索文本不存在的文件时，我看到有一个PID 10025已不存在。

```
Line 204: 10025 26732 root sudo <defunct>
```

该PID的ppid为26732。在文档中跟踪26732时，您会看到该进程与跟踪收集服务相关。

```
Line 201: 10025 26732 root sudo <defunct> 0.0 0.0 1099 0 0 0 Mon Jan 11 10:0
Line 273: 26732 1 513 tracecollection 0.0 2.0 8669698 34069 163696 821564 Thu Oct 8 16:
Line 578: 26732 /usr/local/cm/bin/tracecollection /usr/local/cm/conf/tracecollectionCfg.xml
```

解决方案 — 在此节点上重新启动跟踪收集服务可清除僵尸。

### 示例 3

CUCM：搜索文本不存在的文件时，您会看到有一个PID 23959已不存在。

```
Line 252: 23959 26764 513 du <defunct> 0.0 0.0 3725
```

该PID的ppid为26764。在文档中跟踪26764时，我看到该进程与CDR存储库服务(cdrrep)相关

```
Line 249: 23959 26764 513 du <defunct> 0.0 0.0 3725 1 0
Line 276: 26764 1 513 cdrrep 0.2 1.0 9631438 15471 80840 5368
Line 581: 26764 /usr/local/cm/bin/cdrrep /usr/local/cm/conf/cdrrep/cdrrepCfg.xml
```

解决方案 — 重新启动CDR存储库服务可清除此僵尸。

#### 示例 4

CUC：搜索文本不存在的文件时，您会看到有三个PID 325、370、387已不存在。

```
: 325 7827 265 sftp <defunct> 0.0 0.0 940
: 370 7827 265 sftp <defunct> 0.0 0.0 943
: 387 7827 265 sftp <defunct> 0.0 0.0 943
```

所有这些PID的ppid是7827。在文档中跟踪7827时，您会看到该进程与连接文件同步器服务相关。

```
415: 7827 /opt/cisco/connection/bin/CuFileSync
1509: 325 7827 265 sftp <defunct> 0.0 0.0
1511: 370 7827 265 sftp <defunct> 0.0 0.0
1513: 387 7827 265 sftp <defunct> 0.0 0.0
```

解决方案 — 重新启动连接文件同步器服务会清除僵尸。

#### 示例 5

IMnP：搜索文本不存在的文件时，您会看到PID 1806已不存在。

```
1806 1775 sftpuser ssh <defunct> 0.0
```

该PID的ppid是1775。在文档中跟踪1775时，您会看到该进程是与同一集群中另一个IMnP节点的SFTP连接。

```
287: 1775 sftp sftpuser@imnpsub.emea.lab
511: 1775 1 sftpuser sftp 0.0 0.0 944
512: 1806 1775 sftpuser ssh <defunct> 0.0 0.0 1
```

解决方案 — 在IMnP中，可能会看到SFTP拥有的已失效SSH进程。已发现它们是修饰的，可通过重新启动服务器来移除。

## 重启服务器

重新启动相关服务器会清除进程表中的所有过时条目，并因此清除系统中的僵尸。

## 终止父进程

在Linux中，你不能像使用SIGKILL信号扼杀正常进程那样扼杀僵尸进程 — 僵尸进程已经死亡。但是，您可以终止父进程。该场景中使用的命令如下：

**kill -9 <ppid>**

请联系TAC以执行此解决方法。在终止父进程时确保小心，以确保不突然中断任何关键服务。

## 验证

清除僵尸后，使用相同的命令**show process load**检查僵尸计数。

```
[admin:show process load
top - 22:53:32 up 14 days, 8:33, 1 user, load average: 0.26, 0.10, 0.02
Tasks: 256 total, 1 running, 255 sleeping, 0 stopped, 0 zombie
Cpu(s): 1.8%us, 1.2%sy, 0.0%ni, 96.9%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 3925428k total, 3801924k used, 123504k free, 258168k buffers
```