

目录

[简介](#)

[开始使用前](#)

[规则](#)

[先决条件](#)

[使用的组件](#)

[背景理论](#)

[配置](#)

[在服务中查看脚本Keepalive](#)

[Socket原始](#)

[Socket管理](#)

[复制脚本文件](#)

[脚本示例](#)

[验证](#)

[故障排除](#)

[相关信息](#)

简介

本文描述脚本Keepalive初始实施。执行脚本此方法是最密切相关对是存在可靠性、可用性和维护性的功能(RAS)拨号客户端、终端程序和一般执行脚本工具。此功能使用CSS的富脚本语言。

配置此与简单socket API (连接/断开/发送/接收)将给用户能力剪裁他们自己的协议或者写入步骤他们自己的顺序提供服务的一可靠ALIVEDOWN。您对FTP、HTTP、ICMP和TCP当前被限制。使用此新特性，您能保持在当前协议顶部在写入您自己的脚本旁边。例如，用户能开发特别地被定调子的脚本连接到POP3服务器，无需求思科构件保活类型POP3配合他们的需要。此功能将允许客户创建他们自己的自定义Keepalive适合他们的特定需求。

开始使用前

规则

有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

先决条件

本文档没有任何特定的前提条件。

使用的组件

本文适用于有WebNS软件的CSS 11000/CSS 11500或CSS 11800。

本文档中的信息都是基于特定实验室环境中的设备创建的。本文档中使用的所有设备最初均采用原

始（默认）配置。如果您是在真实网络上操作，请确保您在使用任何命令前已经了解其潜在影响。

背景理论

一旦开发命令行界面(CLI)脚本脱机使用文本编辑例如Notepad，您能上传该脚本到CSS的/script目录和配置在服务的脚本保活选项。您允许创建脚本Keepalive，不用有脚本在系统。一旦没有一份脚本的一脚本Keepalive运行在系统，一个不变在服务。这允许管理员写配置和在写出(或上传前)实现配置所有脚本。

脚本必须位于运行脚本Keepalive的/<current version>/script/目录能查找脚本。当配置脚本Keepalive时，路径名称没有接受，只有脚本名。如果脚本在别处是存在系统，脚本Keepalive假设不存在。这意味着，然而，当软件在系统时升级，旧有脚本位于旧版本？s脚本目录。要复制脚本从旧有目录，请参阅本文[Copying Script Files部分](#)。

您能通过在一个被引述的参数的128个字符。假设七个字符标准每个参数，您能获得大约在一份脚本的18个参数。然而line命令只将接受90个字符。

推荐脚本Keepalive比标准的Keepalive配置与一个更低频率，因为许多脚本将有一多步的进程例如连接，发送请求和等待答复的一种特定类型。由于此，推荐有频率十秒或更加高，以便Keepalive有时间完成完全。否则，状态转换可能经常发生。

脚本能返回状态码零或非零。在返回非零，CSS将标记服务状态作为DOWN。在返回零，CSS将标记服务状态如。参考示例的以下脚本：

CSS永远将返回DOWN状态，如果\$ {SOCKET}变量设置为负值一。检查您的脚本逻辑保证是非常重要的正确值得到返回。

配置

本部分提供有关如何配置本文档所述功能的信息。

对于需要使用脚本Keepalive的很大数量的服务，我们极力推荐他们使用全局Keepalive的更加小的子集处理他们的工作。

要配置脚本Keepalive，请遵从指南和一样其他保活类型的。命令语法如下所示。

```
CS100(config-service[serv1])# keepalive type script ap-kal-smtp "einstein"  
或
```

```
CS100(config-service[serv1])# keepalive type script ap-kal-pop3 "einstein vxworks mipspci"
```

第一行将配置当前服务Keepalive是类型脚本和有脚本名ap-kal-smtp与参数爱因斯坦。第二行显示ap-kal-pop3，类似于第一份脚本，但是通过三个参数：爱因斯坦、vxworks和mipspci。击选项卡(或?)也是可能的密钥发现所有脚本完整列表可用在运行version>/script目录的/<current。当您按下了选项卡或时，这鼓励您采用脚本命名规则，以便?您能清楚看见可用Keepalive的脚本使用。命名规则ap-kal-type使用脚本。例如，SMTP脚本将被命名得ap-kal-smtp。随意地，您能输入在该列表没找到的一，假设您希望以后上传它。脚本通过存档、结算和拷贝命令被操作，并且可以是加载/下载从/script目录在交换机。

脚本名可以是长32个的字符。在引用的字符串必须通过参数，并且可以是长128个的字符。要禁用脚本Keepalive，您通过发出以下命令会重新配置服务：

```
CS100(config-service[cs100])# keepalive type none
```

[在服务中查看脚本Keepalive](#)

当脚本Keepalive配置在服务下时，脚本名在**show service**命令输出中能被看到。脚本出现在保活类型下，并且所有潜在的参数可以被找到直接地在该线路下字段。如果没有脚本参数，该字段没有显示。服务cs100的示例与脚本的呼叫与参数vxworks的ap-kal-pop3，并且mipspci如下所示。

```
Name: cs100                Index: 1
Type: Local                State: Suspended
Rule ( 13.1.1.2  TCP  23 )
Redirect Domain:
Keepalive: (SCRIPT ap-kal-pop3 255 3 5 )
Script Arguments: "einstein vxworks mipspci"
Mtu: 1500                  State Transitions: 0
Connections: 0             Max Connections: 0
Total Connections: 0      Total Reused Conns: 0
Weight: 1                  Load: 255
```

使用此电源，一份脚本类似**ap-kal-pop3**可能采取三个参数，将匹配主机名、用户名和密码POP3服务器的。登陆对POP服务器是足够为了脚本能确定此服务器在ALIVE。

show running-config命令输出如下所示。

```
service cs100
  protocol tcp
  port 23
  ip address 13.1.1.2
  keepalive frequency 255
  keepalive type script ap-kal-pop3 "einstein vxworks mipspci"
```

上述命令输出正确地显示什么脚本(和参数)在此服务配置。如果参数不存在，被引述的文本在脚本名后不会存在。

当工作与脚本记录时，配置如下：

```
service S1
  ip address 192.168.2.3
  keepalive type script ap-kal-pop3 "192.168.2.3 lab labtest1"
  keepalive frequency 10
  keepalive logging file testlog.log
  active
```

此示例显示记录其脚本Keepalive的服务输出到**testlog.log**。

[Socket原始](#)

有在建立结构协议的一脚本Keepalive协助能使用的一些**socket**命令。socket原始将允许ASCII或十六进制发送并且接收功能。安排一个可选原始关键字更改从标准ASCII的数据到十六进制转换的每命令。例如，在ASCII的**abcd**将由**61626364**代表，表示0x61 0x62 0x63 0x64。

参考以下列表socket命令欲知更多信息：

- **socket连接主机名端口端口tcp**-此命令进行一TCP或UDP连接。执行TCP连接介入握手(SYN-SYNACK...)对一特定IP/port。执行UDP连接仅仅是主机/端口的预约。socket值在脚本的一\$ {SOCKET} 变量接收。**注意**：仅32个插槽可以在交换机的所有脚本间随时打开。关于每个参数的列表下面的提供信息。**host-a**应该由远程系统的主机名或IP地址跟随的关键字。**Port关键字**应该由端口跟随协商一连接。**tcp** -一连接使用TCP。**udp** -一连接使用UDP。**整数**-网络建立的超时值以秒钟。如果时间限制超时，在联系顺利地建立了前，尝试失败。因为UDP无连接，这只适用于TCP连接。默认值是五秒超时。**会话**-告诉socket保持开放的关键字，直到会话完成。那含义任何脚本开放插槽在会话上，并且不关闭他们独自地将保持开放，直到用户注销。
- **socket发送socket-字符串[原始]**-此命令通过一以前已连接TCP连接写数据。关于每个参数的列表下面的提供信息。**socket-** - socket文件描述符(整数表)。此描述符返回从连接。**字符串**-至长度128个字符的文本字符串。**原始**-，如果指定，造成字符串值转接作为实际十六进制字节而不是一个简单字符串。例如，**0D0A**没有发送？0 ???D ???0 ???A，'但是相当作为0x0D 0x0A (或回车，换行)。**base64** -这base64在发送它前将编码字符串通过连接。有用的为HTTP基本认证，当连接对密码保护网站时。
- **socket接收socket- [integer] [raw]**-此命令用进来自远程主机的数据填满10K内部缓冲器。此命令然后锁定缓冲区，以便新的数据在此内部缓冲器不会放置。您能继续到使用Inspect转存位于此内部10K缓冲区的所有数据对标准的输出。**注意**：在新的数据放置前，在10K内部缓冲器的所有上一个数据被驱赶出来。关于每个参数的列表下面的提供信息。**socket-** - socket文件描述符(整数表)。此描述符返回从连接。**整数**-代表毫秒的数量整数值在锁定内部10K缓冲区和返回前等待对用户。如果整数时刻没有指定，接收在返回前将等待100ms给用户。**原始**-，如果指定，造成字符串值转接作为实际十六进制字节而不是一个简单字符串。例如，**0D0A**没有发送？0 ???D ???0 ???A，'但是相当作为0x0D 0x0A (或回车，换行)。
- **socket waitfor socket-字符串[integer] [case-sensitive] [offset] [raw]**-此命令类似于socket接收，除了返回立即在查找指定的字符串参数。一旦找到指定的字符串，将返回\$ {}零。否则，它返回1。获取的数据可以通过发出**socket inspect**命令进一步查看。关于每个参数的列表下面的提供信息。**socket-** - socket文件描述符(整数表)。此描述符返回从连接。**字符串**-必须发现导致一\$的特定字符串{}零，指示找到。如果找到字符串，立即返回和不等待整个整数超时长度。**整数**-代表毫秒的数量整数值在锁定内部10K缓冲区和返回前等待对用户。如果整数时刻没有指定，接收在返回前将等待100ms给用户。**区分大小写**-，如果指定，表明串比较应该区分大小写。例如，**用户：**与**用户请勿**是等同的：**：**。**偏移量**-多少个字节到接收的数据里应该找到字符串。例如，如果搜索**a0**并且给偏移量**十**，您将寻找**a0**十字节到接收的数据。**原始**-，如果指定，造成字符串值转接作为实际十六进制字节而不是一个简单字符串。例如，**0D0A**没有发送？0 ???D ???0 ???A，'但是相当作为0x0D 0x0A (或回车，换行)。
- **socket Inspect socket- [pretty] [raw]**-此命令检查socket ?s数据缓冲区(内部)实际数据的。如果找到数据，该数据显示对标准的输出。如果显示的字符非可打印的，他们将由一个小点代表(。)可读性的。关于每个参数的列表下面的提供信息。**socket-** - socket文件描述符(整数表)。此描述符返回从连接。**原始**-，如果指定，造成字符串值显示作为实际十六进制字节相当然后一个简单字符串。相当然后打印**ABCD**对标准输出，它将打印**41424344** (1字节十六进制等同)。**无用的饰物**打印输出。每条线路将包含ASCII或十六进制等同每字节的数据的。将有在每条线路打印的16个字节。例如，**0x41 0x42 0x43 0x44 0x10 0x05 ABCD**。
- **socket断开socket- [graceful]**-此命令断开对远程主机的连接。这由发送RST完成对远程主机，以便知道执行发送数据的您。关于每个参数的列表下面的提供信息。**socket-** - socket文件描述符(整数表)。此描述符返回从连接。**优美**-发送FIN而不是RST到远程主机结束连接的优美的断开。

有32个开放(在使用中的)插槽的限制在随时交换机的。如果用户执行socket连接，并且以该socket的socket断开不结束？s文件描述符(保存以\$ {SOCKET})， socket依然是开放，直到socket断开呼叫与该socket作为参数。在脚本内打开的插槽关闭，当脚本结束时(除非会话参数在socket通过连接)。当会话结束时，在会话内打开的插槽关闭。

如果socket保持开放，这通常是用户建立了联系的案件，无需适当关闭。socket将保持开放和使用，直到它关闭或直到脚本(或终端连接)关闭。**show sockets命令**实现列出正在使用中的所有使用的socket文件描述符，以使用户了解什么是开放的，并且什么关闭。

注意： 如果远程主机计时一socket，或者socket由远程主机关闭， socket体系结构是足够聪明的整理它，并且从中拿走使用的插槽列表(找到通过发出**show sockets命令**)。这只将发生，在用户尝试做否则由远程主机的另一次转移在socket后(关闭坐等候用户需要)的空闲。

show sockets命令输出如下所示。

Socket ID	Remote Host:Port	User	Time
38	192.168.2.3:80	console	0 days 00:00:23
37	192.168.2.4:80	console	0 days 00:00:27
36	192.168.2.4:80	console	0 days 00:29:23
40	192.168.2.3:80	console	0 days 00:00:19
45	192.168.2.4:80	vty1	0 days 00:00:02
44	192.168.2.4:80	vty1	0 days 00:00:03

屏幕列出socket ID (文件描述符)，连接的主机/端口对、用户和计时器描述符多久是开放的。如在CLI的**show lines命令**输出中看到将代表线路标识符。

获取数据使用socket接收能一次缓冲10K价值数据。此缓冲区将依然是不可更改，直到用户执行另一socket接收，到时清除缓冲区并且重新装满与来更多的数据电线。每socket描述符(创建从socket请连接)将有其自己的10K缓冲区。

复制脚本文件

当升级对代码新版本，您在以前版本修改了的所有脚本文件时？s脚本目录将需要复制到新版本。

在升级您的交换机前遵从这些步骤：

1. FTP到CSS交换机。请使用管理端口或VLAN电路地址。
2. 对脚本目录的CD。
3. 下载您编辑到您的本地设备的所有脚本。
4. 升级您的交换机。
5. 上传从您的本地设备的脚本到交换机？s新脚本目录。

脚本示例

以下脚本是可用为了提供您一些默认实施。这些示例包含为DNS、响应、NetBios、Finger、时间、HttpList、PingList、HttpAuth、Imap4、CookieSet、POP3、HttpTag、Mailhost和SMTP写入的脚本。

发出**显示脚本？**发出命令发现脚本。示例命令输出如下所示。

```

CS150# show script ap-kal-dns NOV 17 09:58:36 1555 ap-kal-echo NOV
17 09:58:36 1920 ap-kal-finger NOV 17 09:58:36 1172 ap-kal-httpauth
NOV 17 09:58:36 1927 ap-kal-httpplist NOV 17 09:58:36 1674 ap-kal-
httpstag NOV 17 09:58:36 1180 ap-kal-imap4 NOV 17 09:58:36 1556
ap-kal-ldap NOV 17 09:58:36 1640 ap-kal-mailhost NOV 17 09:58:36
2437 ap-kal-netbios NOV 17 09:58:36 1632 ap-kal-pinglist NOV 17 09:58:36
739 ap-kal-pop3 NOV 17 09:58:36 1568 ap-kal-setcookie NOV 17 09:58:38
1436 ap-kal-smtp NOV 17 09:58:38 1310 ap-kal-ssl NOV 17 09:58:38
2053 ap-kal-time NOV 17 09:58:38 1064 cache.map NOV 17 09:58:38
1615 commit_redundancy NOV 17 09:58:38 109224 commit_vip_redund.. NOV 17 09:58:38
132147 default-profile NOV 17 09:58:38 1240 dnslookup NOV 17 09:58:40
8009 eql-cacheable NOV 17 09:58:40 1186 eql-graphics NOV 17 09:58:40
234 eql-multimedia NOV 17 09:58:40 279 flowinfo NOV 17 09:58:40
5665 monitor NOV 17 09:58:40 3734 pcm-collect-cfgs NOV 17 09:58:40
2373 pcm-repeat-cmd NOV 17 09:58:40 4995 service-load NOV 17 09:58:40
920 setup NOV 17 09:58:40 24328 showtech NOV 17 09:58:40
2528 testpeering NOV 17 09:58:40 34142 upgrade NOV 17 09:58:40
17117 ap-kal-ssl.txt NOV 24 09:18:00 2053

```

验证

当前没有可用于此配置的验证过程。

故障排除

请使用以下指南排除故障您的配置：

- 从line命令运行脚本通过发出**script play**命令。发出此命令，当登陆，确保它的超级用户完成不出错误。如果它执行错误，导致错误的线路应该发出。
- 采取嗅探器跟踪在CSS和Web服务器之间观察什么实际上返回，当脚本运行与时什么脚本期望发现。
- 在5.x编码版本中，**use-output**命令被添加了。此命令必须与使用**grep**命令的所有脚本一起使用。例如，**ap-kal-dns**保活类型脚本参数使用输出。

相关信息

- [WEB网络服务软件支持](#)
- [CSS 11000系列内容服务交换机硬件支持](#)
- [CSS 11500系列内容服务交换机硬件支持](#)
- [Cisco WebNS CSS11500软件下载](#)
- [Cisco WebNS CSS 11000软件下载](#)
- [技术支持 - Cisco Systems](#)