

采样Keepalive脚本检查网页Webstring

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[示例脚本](#)

[相关信息](#)

简介

此脚本会在网页中检查 Web 字符串。如果该字符串缺失，则将该服务标记为 down。此脚本用于任何种类的页，尤其是通过脚本、ColdFusion 等生成的动态页。本文也解决脚本Keepalive的实施。此脚本方法是对功能最密切相关的，存在于远程接入服务器(RAS)拨号客户端、终端程序和通用脚本实用程序。此功能使用WebNS的丰富的脚本语言。

完成与一简单的插槽Application Program Interface (API) (connect或断开或送信/收信)，脚本Keepalive将产生用户能力剪裁他们自己的协议，或者请写步骤他们自己的顺序提供服务的一可靠的ALIVE或DOWN状态。没有脚本Keepalive功能，您当前被限制到FTP、HTTP、ICMP和TCP。然而，有了脚本Keepalive，您能通过写您自己的脚本保持在当前协议顶部。例如，您能开发特别的脚本，以连接到POP3服务器，无要求WebNS建立保活类型的POP3。此功能允许客户创建他们自己自定义的Keepalive匹配他们的特定需求。虽然这是内容服务交换机(CSS)的一部分，但是 Cisco技术支持中心不支持自定义脚本(Cisco TAC)。

TAC不正式支持下面的脚本Keepalive，然而已被测试，并且是可用的，请您谨慎使用。

先决条件

要求

对WebNS富脚本语言的熟悉。

使用的组件

本文档中的信息基于以下软件和硬件版本：

- WebNS版本3.x和更高
- CSS11x00 系列

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。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

示例脚本

下面脚本可以用于检查网页webstring。

```
!--- No echo. !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! !--- Filename:
ap-kal-httpstring !--- Parameters: WebsiteIP WebPage WebString [Port] !--- Requirements:
WebNS4.x or higher ! !--- Uses: !--- Checks the Web page for the Web string. If the string is
missing, !--- mark the service as down. Used with any sort of page, especially dynamic !--- ones
that are generated via scripts, ColdFusion, and so on. ! !--- Logic: !--- The script connects to
a Web server on port 80 by default. !--- It performs a GET on the specified page. !--- If the
Web string is returned, the service stays up. !--- If anything fails, the service is marked
down. ! !--- Notes: !--- The Web string is case-sensitive. !--- Only the first 10Kb of the
response is inspected. ! ! !--- Tested: 04/12/01-KGS !
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! no set CONTINUE_ON_ERROR no
set EXIT_MSG !--- Make sure the user has the proper number of arguments. if ${ARGS}[#] "LT" "3"
echo "Usage: ap-kal-httpcheck \'WebSiteIP WebPage WebString [Port]\'" exit script 1 endbranch !-
-- Set variables corresponding to the args. set WebSite "${ARGS}[1]" set WebPage "${ARGS}[2]"
set WebString "${ARGS}[3]" set WebPort "80" if ${ARGS}[#] "GT" "3" set WebPort "${ARGS}[4]"
endbranch echo "Requesting ${WebPage} from ${WebSite} on port ${WebPort}." !--- Connect to the
remote server. set EXIT_MSG "Connect: Failed. Could not connect to ${WebSite} on port
${WebPort}" set CONTINUE_ON_ERROR "1" socket connect host ${WebSite} port ${WebPort} tcp if
${STATUS} "NEQ" "0" exit script 1 endbranch no set CONTINUE_ON_ERROR !--- Request the desired
Web page. set EXIT_MSG "Send: Failed. Could not send to ${WebSite}:${WebPort}" socket send
${SOCKET} "GET ${WebPage} HTTP/1.0\n\nHost: ${WebSite}:${WebPort}\n" !--- Look for the Web
string. set EXIT_MSG "Waitfor: Failed. Did not find [${WebString}]" set CONTINUE_ON_ERROR "1"
socket waitfor ${SOCKET} "${WebString}" case-sensitive if ${STATUS} "NEQ" "0" exit script 1
endbranch no set CONTINUE_ON_ERROR !--- Disconnect from the server. no set EXIT_MSG socket
disconnect ${SOCKET} graceful exit script 0
```

相关信息

- [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](#)
- [CSS11500的\(仅限注册用户\)软件下载](#)
- [技术支持和文档 - Cisco Systems](#)