

Fazendo keepalives de HTTP e servidores coldfusion do Cisco CSS 11000 trabalhe junto

Índice

[Introdução](#)

[Entendendo os cabeçalhos HTTP](#)

[Compreendendo diferenças entre métodos do HTTP HEAD e HTTP GET](#)

[Como os servidores ColdFusion respondem às manutenções de atividade de HTTP](#)

[As respostas da manutenção de atividade de HTTP que o CSS 11000 entende](#)

[Outro problema de URI de manutenção de atividade e ColdFusion](#)

[Utilizando manutenções de atividade em script como uma solução](#)

[Informações Relacionadas](#)

Introdução

Um Content Services Switch da Cisco CSS 11000 Series envia periodicamente um mensagem de keepalive a um serviço para determinar se o estado do serviço está vivo, morrendo, ou absolutamente. Você pode usar um keepalive enquanto um exame médico completo para se certificar do serviço está acima de modo que o CSS11000 possa usar o serviço para o Balanceamento de carga. Caso contrário, o CSS 11000 removerá o serviço do algoritmo de balanceamento de carga. Um tipo avançado de manutenção de atividades (Camada 5) é um Hypertext Transfer Protocol (HTTP) que usa um método HTTP HEAD (padrão) ou HTTP GET. Um CSS 11000 executando a versão 4.10 ou posterior do software Cisco Web Network Services (WebNS) que usa a manutenção de atividade HTTP espera o pacote de resposta em um formato específico conforme descrito abaixo.

Servidores ColdFusion (versão 4.5.X e posteriores) ou servidores JRun executados atrás de um CSS 11000, por padrão, não respondem às manutenções de atividades HTTP do CSS 11000 no formato esperado. É necessário configurar tais servidores para retornar valores esperados específicos em resposta à manutenção de atividades de HTTP do CSS 11000. Como paliativo, você pode usar manutenção de atividade feito script no CSS 11000, usando o WebNS 4.10 e mais recente.

Entendendo os cabeçalhos HTTP

O seguinte é um pedido típico HTTP GET que seja feito pelo navegador da Web:

```
HTTP: ----- Hypertext Transfer Protocol -----
```

```
HTTP:
```

```
HTTP: Line 1: GET / HTTP/1.1
```

```
HTTP: Line 1: GET / HTTP/1.1
```

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

Uma resposta do servidor de Web é composta de um identificador da versão do protocolo, de um código de status, de uma linha de status de resposta compreensível para o utilizador, de uns cabeçalhos da resposta, e de uma informação para o pedido.

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: GET / HTTP/1.1

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

Entendendo as diferenças entre métodos de um HTTP HEAD e um HTTP GET

A manutenção de atividade HTTP do CSS 11000 usa um dos dois métodos disponíveis:

- TÍTULO
- GET

Eles são especificados com o comando method no modo de configuração de manutenção de atividade (config-keepalive).

Quando uma manutenção de atividade HTTP HEAD é emitida no CSS 11000, uma mensagem 200 Status OK é esperada no pacote de resposta a partir do servidor que está consultando. Se o status 200 OK não for retornado, o CSS 11000 considerará o serviço inativo. Se o CSS 11000 receber qualquer outro código de status como o 404 Objeto não encontrado ou o 302 O objeto mudou de endereço, o CSS 11000 entenderá como uma resposta incorreta e marcará o serviço

como desconectado. O método de início é o padrão. O CSS11000 não computa um valor de hash da referência para este tipo de keepalive.

Quando uma manutenção de atividade GET HTTP é usada, o CSS 11000 espera não apenas ver o status 200 OK no pacote de resposta, mas também executará uma soma de verificação no corpo da entidade. Na primeira vez que o CSS 11000 sair da consulta, o servidor executará uma soma de verificação no corpo da entidade e armazenará esse valor de hash para consultas futuras. Se as respostas futuras passam para trás uma APROVAÇÃO do estado 200 e o valor de hash que está calculado é diferente do valor armazenado, a seguir o serviço está considerado para baixo. Se o 200 Staus OK não for retornado ou se o status 200 OK for retornado, mas o valor de combinação for diferente do valor de combinação de referência, o CSS 11000 considera o serviço inativo.

Quando você definir um URL para uma manutenção de atividade de HTTP, o CSS 11000 calculará um valor de hash para a página da Web especificada no URL. Se houver alteração na página da Web, o valor de hash não será mais compatível com o valor de hash original e o CSS 11000 conclui que o serviço está inativo. Para evitar que o CSS 11000 considere que um serviço está desativo devido a uma incompatibilidade de valor hash, especifique o método de manutenção de atividade como o começo. Devido à natureza de manutenções de atividades HTTP GET e ao modo como são calculadas, não use-as em páginas alteradas dinamicamente porque suas somas de verificação mudam a todo momento.

Como os servidores ColdFusion respondem às manutenções de atividade de HTTP

Os servidores coldfusion respondem diferentemente aos keepalives de HTTP. Uma resposta ColdFusion vem em dois pacotes: Geralmente, o ColdFusion funciona assim:

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

Um pacote de resposta de um servidor coldfusion olha similar ao seguinte:

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

```
HTTP: Line 1: HTTP/1.1 200 OK
```

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

Na maioria dos casos você pode esperar ver a seguinte saída:

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

As respostas da manutenção de atividade de HTTP que o CSS 11000 entende

Necessidade dos pacotes de conter um índice-comprimento: <value> tag.

Depois de receber o cabeçalho de resposta, o CSS 11000 pesquisa um caractere de comprimento de conteúdo para determinar a quantidade de dados que o corpo da mensagem deve incluir. Os CSS 11000 monitora o decréscimo desse valor para determinar se ele tem uma resposta completa. Isto é imperativo em fluxos persistentes HTTP 1.1. Se o tipo de resposta for HTTP 1.0, a duração do conteúdo será opcional. O CSS11000 usá-lo-á se esta presente. Normalmente, o CSS 11000 configura o `uIndicatedLen` para 0 e observa um término de conexão. Por último, o método de requisição deve ser levado em consideração para determinar a existência/não existência de um corpo de entidade. Se o método do pedido era CABEÇA, não deve haver um corpo de entidade.

O campo do tipo de conteúdo deve ser terminado com `0x0d0a<CR><LF>` e não `<LF><LF>`.

O CSS11000 espera ver o pacote de resposta em um pacote, não dois.

O CSS 11000 pode processar uma resposta de manutenção de atividade que está dividida em dois pacotes. Quando o primeiro quadro chega, o CSS11000 deve assegurar-se de que tenha o encabeçamento de resposta HTTP inteiro assim que pode começar a processar a resposta de `keepalive`.

O encabeçamento de resposta HTTP deve ser permitido medir pacotes e deve terminar em

0x0d0a0d0a <CR><LF><CR><LF>. Contudo, se há uns encontros com server que responderam somente ao encabeçamento de resposta HTTP que terminou com <CR><LF>, um CSS11000 que siga padrões do RFC2068 apoiará qualquer um. Para apoiar a fragmentação do cabeçalho da resposta e de um ou outro terminal, se o 0x0d0a <CR><LF> é o terminal do segmento, significando é os últimos dois caracteres no pacote, a seguir supõe-se que este é o terminal do encabeçamento de resposta HTTP.

Os server da fusão a frio terminam o primeiro pacote (segmento) com 0x0d0a <CR><LF>, assim o CSS11000 supõe que esta é a extremidade do encabeçamento de resposta HTTP e tenta o processar. Como o restante do cabeçalho está realmente no segundo pacote, ele falhará, indicando que o serviço nunca será ativado. Idealmente, a resposta deve ocorrer em um pacote mas se dois pacotes são usados o CSS11000 precisa o seguinte:

O primeiro pacote não pode ser terminado com <CR><LF>. É necessário combinar todos os dados em um pacote ou então, a aplicação precisa garantir que a divisão entre pacotes 1 e 2 não ocorra nos caracteres <CR><LF>. Os dados que chegam em dois pacotes podem encontrar problemas com a maneira como o CSS 11000 irá analisar os dois pacotes.

Outro problema de URI de manutenção de atividade e ColdFusion

Quando são feitos um HEAD ou um GET em um arquivo que não existe no servidor ColdFusion, este responde com um código HTTP de 200 OK e uma mensagem 404 de objeto não encontrado, no corpo do HTML.

Por exemplo, está aqui um pedido do HTTP HEAD:

```
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
```

Está aqui uma resposta típica se o arquivo de /keepalive.cfm não está atual.

```
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
HTTP: Line 1: HTTP/1.1 200 OK
```

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 1: HTTP/1.1 200 OK

HTTP: Line 6: Error Occurred While Processing Request

HTTP: Line 7: Error Diagnostic Information

HTTP: Line 8: An error has occurred.

HTTP: Line 9: .0 404 Object Not Found

Utilizando manutenções de atividade em script como uma solução

Você pode usar manutenções de atividades em scripts no CSS 11000 usando WebNS 4.10 e superior (qualquer compilação). Busca para a corda não encontrada de 404 objetos na resposta. Por padrão, o CSS 11000 usará manutenções de atividades com scripts compactadas para verificar a integridade dos serviços padronizados, como SMTP e Netbios. O CSS 11000 permite que os clientes personalizem seus próprios scripts, embora eles não sejam suportados pelo TAC Cisco. Os clientes devem depurar e fazer Troubleshooting dos seus próprios scripts.

Note: Ajuste a configuração de servidores de aplicativos de terceiros para que eles funcionem com um CSS 11000, utilizando manutenções de atividades de HTTP.

Informações Relacionadas

- [Cisco CSS 11000 Series Content Services Switches](#)
 - [Sustentação do produto da Distribuição de vídeo e conteúdo](#)
 - [Comandos do modo de configuração de manutenção de atividade](#)
 - [Suporte Técnico e Documentação - Cisco Systems](#)
-