

Resolución de problemas de uso elevado de CPU/memoria en dispositivos Kubernetes

Contenido

[Introducción](#)

[1. Alerta de problemas de CPU/memoria alta en Pod](#)

[1.1. Alerta para la CPU](#)

[1.2. Alerta para memoria](#)

[2. Perfiles por proceso de Kubernetes](#)

[2.1. Perfiles de CPU \(/debug/prof/profile\)](#)

[2.2. Perfiles de memoria \(/debug/prof/heap\)](#)

[2.3. Perfiles de rutina \(/debug/prof/goroutine\)](#)

[2.4. Encontrar puerto de prueba en un POD de Kubernetes](#)

[3. Datos que deben recopilarse del sistema](#)

[4. Comprensión de las salidas de registro de pruebas recopiladas](#)

[4.1. Salida de lectura de perfiles de memoria \(/debug/prof/heap\)](#)

[5. Grafana](#)

[5.1. Consulta de CPU](#)

[5.2. Consulta de memoria](#)

Introducción

Este documento describe cómo resolver problemas de CPU o memoria en la plataforma de implementación nativa en la nube (CNDP) que se utiliza como función de administración de sesiones (SMF) o función de control de políticas (PCF).

1. Alerta de problemas de CPU/memoria alta en Pod

Es importante entender la alerta para tener un buen comienzo en la resolución de este problema. Una explicación de todas las alertas predeterminadas que están preconfiguradas se encuentra en [este link](#).

1.1. Alerta para la CPU

Aquí, hay una alerta predeterminada activa que se activa denominada `k8s-pod-cpu-usage-high` .

Verá que está relacionado con un grupo de dispositivos denominado: `smf-udp-proxy-0` y es un contenedor: `k8s_smf-udp-proxy_smf-udp-proxy-0_smf`

Verá que este contenedor está en el espacio de nombres: `smf`

```
alerts active detail k8s-pod-cpu-usage-high 36fbd5e0bbce
severity major
type "Processing Error Alarm"
startsAt 2024-02-23T12:45:44.558Z
source smf-udp-proxy-0
summary "Container: k8s_smf-udp-proxy_smf-udp-proxy-0_smf of pod: smf-udp-proxy-0 in namespace: smf has
labels [ "name: k8s_smf-udp-proxy_smf-udp-proxy-0_smf" "namespace: smf" "pod: smf-udp-proxy-0" ]
```

En el maestro de Kubernetes, busque la vaina afectada ingresando este comando:

```
master $ kubectl get pods smf-udp-proxy-0 -n smf
```

1.2. Alerta para memoria

Aquí, hay una alerta predeterminada activa que se activa denominada `container-memory-usage-high` .

Puede ver que está relacionado con un grupo de dispositivos denominado: `grafana-dashboard-sgw-765664b864-zwxct` y es un contenedor: `k8s_istio-proxy_grafana-dashboard-sgw-765664b864-zwxct_smf_389290ee-77d1-4ff3-981d-58ea1c8eabdb_0`

Este contenedor está en el espacio de nombres: `smf`

```
alerts active detail container-memory-usage-high 9065cb8256ba
severity critical
type "Processing Error Alarm"
startsAt 2024-04-25T10:17:38.196Z
source grafana-dashboard-sgw-765664b864-zwxct
summary "Pod grafana-dashboard-sgw-765664b864-zwxct/k8s_istio-proxy_grafana-dashboard-sgw-765664b864-zwxct"
labels [ "alertname: container-memory-usage-high" "beta_kubernetes_io_arch: amd64" "beta_kubernetes_io_c" ]
annotations [ "summary: Pod grafana-dashboard-sgw-765664b864-zwxct/k8s_istio-proxy_grafana-dashboard-sgw-765664b864-zwxct" ]
```

En el maestro de Kubernetes, busque la vaina afectada ingresando este comando:

```
master $ kubectl get pods grafana-dashboard-sgw-765664b864-zwxct -n smf
```

2. Perfiles por proceso de Kubernetes

2.1. Perfiles de CPU (`/debug/prof/profile`)

La creación de perfiles de CPU sirve como técnica para capturar y analizar el uso de CPU de un programa Go en ejecución.

Toma muestras de la pila de llamadas periódicamente y registra la información, lo que le permite analizar dónde pasa el programa la mayor parte de su tiempo.

2.2. Perfiles de memoria (`/debug/prof/heap`)

La creación de perfiles de memoria proporciona información sobre la asignación de memoria y los patrones de uso de la aplicación Go.

Puede ayudarle a identificar pérdidas de memoria y a optimizar la utilización de la memoria.

2.3. Perfiles de rutina (`/debug/prof/goroutine`)

La creación de perfiles de Goroutine proporciona información sobre el comportamiento de todas las Goroutine actuales mediante la visualización de sus rastros de pila. Este análisis ayuda a identificar las Goroutines atascadas o con fugas que pueden afectar el rendimiento del programa.

2.4. Encontrar puerto de prueba en un POD de Kubernetes

Comando:

```
master:~$ kubectl describe pod <POD NAME> -n <NAMESPACE> | grep -i pprof
```

Ejemplo de salida:

```
master:~$ kubectl describe pod udp-proxy-0 -n smf-rcdn | grep -i pprof
PPROF_EP_PORT: 8851
master:~$
```

3. Datos que deben recopilarse del sistema

Durante el tiempo del problema y la alerta activa en el entorno de ejecución común (CEE), recopile los datos que cubren el tiempo antes y durante/después del problema:

CEE:

```
cee# show alerts active detail
cee# show alerts history detail
cee# tac-debug-pkg create from yyyy-mm-dd_hh:mm:ss to yyyy-mm-dd_hh:mm:ss
```

Nodo maestro CNDP:

General information:

```
master-1:~$ kubectl get pods <POD> -n <NAMESPACE>
master-1:~$ kubectl pods describe <POD> -n <NAMESPACE>
master-1:~$ kubectl logs <POD> -n <NAMESPACE> -c <CONTAINER>
```

Login to impacted pod and check top tool:

```
master-1:~$ kubectl exec -it <POD> -n <NAMESPACE> bash
root@protocol-n0-0:/opt/workspace# top
```

If pprof socket is enabled on pod:

```
master-1:~$ kubectl describe pod <POD NAME> -n <NAMESPACE> | grep -i pprof
master-1:~$ curl http://<POD IP>:<PPROF PORT>/debug/pprof/goroutine?debug=1
master-1:~$ curl http://<POD IP>:<PPROF PORT>/debug/pprof/heap
master-1:~$ curl http://<POD IP>:<PPROF PORT>/debug/pprof/profile?seconds=30
```

4. Comprensión de las salidas de registro de pruebas recopiladas

4.1. Salida de lectura de perfiles de memoria (/debug/prof/heap)

This line indicates that a total of 1549 goroutines were captured in the profile. The top frame (0x9207a9) shows that the function `google.golang.org/grpc.(*addrConn).resetTransport` is being executed, and the line number in the source code is `clientconn.go:1164`.

Cada sección que comienza con un número (por ejemplo, 200) representa un seguimiento de pila de una Goroutine.

```
goroutine profile: total 1549
200 @ 0x4416c0 0x415d68 0x415d3e 0x415a2b 0x9207aa 0x46f5e1
# 0x9207a9 google.golang.org/grpc.(*addrConn).resetTransport+0x6e9 /opt/workspace/gtpc-ep/pkg/m
```

The first line in each section shows the number of goroutines with the same stack trace. For example, there are 200 goroutines with the same stack trace represented by memory addresses (0x4416c0, 0x415d68, and more.). The lines that start with # represent the individual frames of the stack trace. Each frame shows the memory address, function name, and the source code location (file path and line number) where the function is defined.

```
200 @ 0x4416c0 0x45121b 0x873ee2 0x874803 0x89674b 0x46f5e1
# 0x873ee1 google.golang.org/grpc/internal/transport.(*controlBuffer).get+0x121 /opt/workspace/g
# 0x874802 google.golang.org/grpc/internal/transport.(*loopyWriter).run+0x1e2 /opt/workspace/g
# 0x89674a google.golang.org/grpc/internal/transport.newHTTP2Client.func3+0x7a /opt/workspace/g

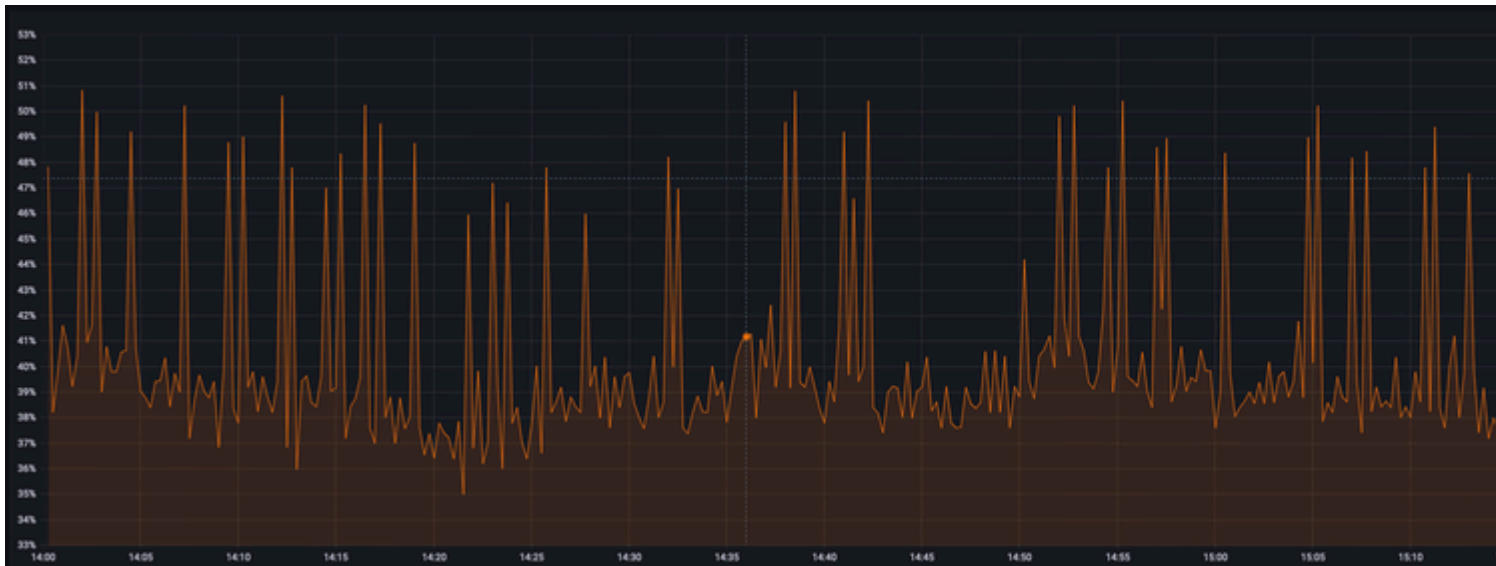
92 @ 0x4416c0 0x45121b 0x873ee2 0x874803 0x897b2b 0x46f5e1
# 0x873ee1 google.golang.org/grpc/internal/transport.(*controlBuffer).get+0x121 /opt/workspace/g
# 0x874802 google.golang.org/grpc/internal/transport.(*loopyWriter).run+0x1e2 /opt/workspace/g
# 0x897b2a google.golang.org/grpc/internal/transport.newHTTP2Server.func2+0xca /opt/workspace/g
```

5. Grafana

5.1. Consulta de CPU

```
sum(cpu_percent{service_name=~"[[microservice]]"}) by (service_name,instance_id)
```

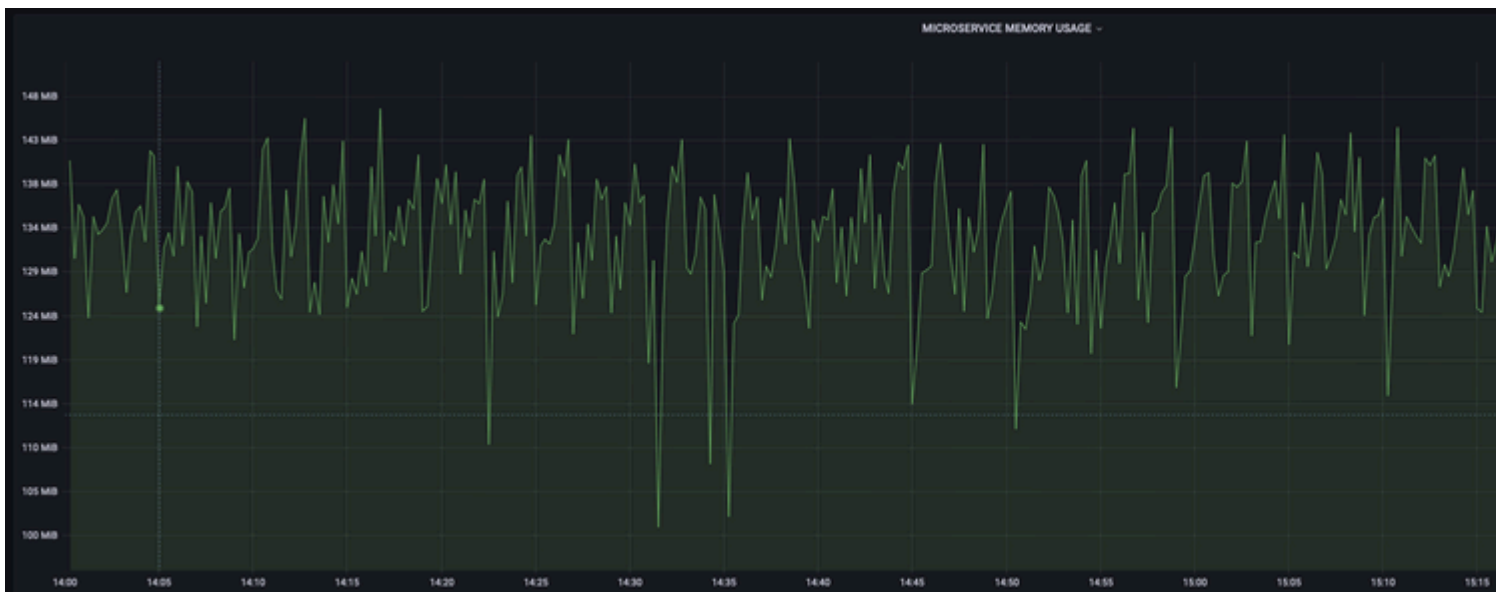
Ejemplo:



5.2. Consulta de memoria

```
sum(increase(mem_usage_kb{service_name=~"[[microservice]]"}[15m])) by (service_name,instance_id)
```

Ejemplo:



Acerca de esta traducción

Cisco ha traducido este documento combinando la traducción automática y los recursos humanos a fin de ofrecer a nuestros usuarios en todo el mundo contenido en su propio idioma.

Tenga en cuenta que incluso la mejor traducción automática podría no ser tan precisa como la proporcionada por un traductor profesional.

Cisco Systems, Inc. no asume ninguna responsabilidad por la precisión de estas traducciones y recomienda remitirse siempre al documento original escrito en inglés (insertar vínculo URL).