

Comprender show resources CLI en la aplicación SA 5G

Contenido

[Introducción](#)

[Cómo funciona](#)

[Descripción del resultado](#)

[Explicación adicional del lenguaje de programación Go](#)

Introducción

Este documento describe cómo entender el resultado de la CLI **show resources** en la aplicación SA 5G de Cisco.

Cómo funciona

La CLI **show resources** enumera la información de recursos para todos los grupos de dispositivos (principalmente los grupos de protocolos y de servicios) del clúster. La información de recursos se recopila periódicamente de cada grupo de dispositivos y se presenta a la CLI.

Ejemplo de resultado

```
[unknown] smf# show resources
TOTAL USED DISK
NODE POD USAGE GO GC
CPU MEMORY MEMORY IN ROUTINES GC PAUSE
POD INSTANCE USAGE IN MB IN MB Kbps COUNT COUNT IN NS
```

```
-----
cache-pod-1 2 32011 85 0 172 29684 85
cache-pod-2 2 32011 83 0 172 29627 83
gtpc-ep-0 2 32011 70 0 69 29088 70
li-ep-0 5 32011 51 0 56 29095 51
oam-pod-1 2 32011 98 0 157 29095 98
smf-nodemgr-0 7 32011 94 0 213 29096 94
smf-protocol-0 3 32011 135 0 206 29092 135
smf-rest-ep-0 4 32011 125 0 203 29091 125
smf-service-0 3 32011 321 0 247 34958 321
smf-udp-proxy-0 1 32011 82 0 88 29083 82
```

Descripción del resultado

INSTANCIA POD	Nombre del grupo
CPU USAGE	Uso de CPU del POD
MEMORIA TOTAL DE NODOS EN MB	Memoria total del nodo de trabajo donde se ejecuta el grupo de dispositivos
MEMORIA DE POD UTILIZADA EN MB	Memoria utilizada por el grupo de dispositivos
USO DE DISCO EN Kbps	Velocidad de E/S de disco en Kbps

CONTAR RUTINAS

Número de Go Routine

GC COUNT

Recuento rutinario de recolección de basura

GC PAUSE IN NS

Pausa rutinaria de recolección de basura en NanoSegundos

Explicación adicional del lenguaje de programación Go

Go Routine es una especie de "hilo" en el lenguaje de programación Go. Se utiliza para algunos fines, como un mejor rendimiento, etc. En un escenario problemático, se puede ver una fuga rutinaria de Go donde las Rutinas Go no están correctamente terminadas y las nuevas Rutinas Go siguen creando. El impacto de la fuga de Go Routine es similar a la pérdida de memoria. Esto se puede identificar consultando CONTEO DE RUTINAS GO en el resultado de show resources. Si se observa un aumento espectacular o constante, existe la posibilidad de una fuga de Go Routine.

La recolección de elementos no utilizados es una función de administración de memoria integrada en Go. Se ejecuta de forma continua y limpia los objetos que ya no se utilizan para guardar el uso de la memoria. GC COUNT y GC PAUSE IN NS se pueden utilizar para monitorear si la recolección de elementos no utilizados funciona sin ningún problema. Por ejemplo, demasiados GC cuentan para un grupo de dispositivos específico o una pausa GC demasiado larga puede indicar algún problema.