

Aislamiento de un Core de la CPU en Linux

Aislar un núcleo (core) de la CPU puede ser útil para tareas de alta prioridad o en sistemas en tiempo real, donde se necesita garantizar el máximo rendimiento y mínima interferencia de otros procesos. A continuación se describe el procedimiento paso a paso.

Características del procesador

Para conocer la topología del procesador, se utiliza el siguiente comando:

```
lscpu
```

Este comando muestra información detallada sobre los cores físicos, threads y CPUs lógicas. Por ejemplo, si se tienen 6 cores físicos y 2 threads por core, habrá un total de 12 CPUs lógicas (threads).

Identificar los cores a aislar

En este ejemplo, se desea aislar el sexto core físico. Asumiendo que los threads se asignan secuencialmente, los identificadores de CPU correspondientes serían 5 y 11.

Paso 3: Modificar la configuración de GRUB

Editar el archivo de configuración de GRUB:

```
sudo nano /etc/default/grub
```

Buscar la línea que comienza con `GRUB_CMDLINE_LINUX` y añadir los siguientes parámetros:

```
isolcpus=5,11 nohz_full=5,11 rcu_nocbs=5,11
```

Ejemplo:

```
GRUB_CMDLINE_LINUX="quiet splash isolcpus=5,11 nohz_full=5,11 rcu_nocbs=5,11"
```

No eliminar los parámetros existentes; simplemente añadir los nuevos.

Actualizar GRUB

Después de guardar los cambios, actualizar GRUB con el siguiente comando:

```
sudo update-grub
```

Reiniciar el sistema

Para aplicar los cambios, es necesario reiniciar el equipo:

```
sudo reboot
```

Ejecutar un proceso en el core aislado

Para ejecutar un proceso en el core aislado (por ejemplo, el thread con ID 5), usar el comando **taskset**:

```
sudo taskset -c 5 <comando>
```

Por ejemplo, para ejecutar un script:

```
sudo taskset -c 5 ./mi_script.sh
```

Verificación

Se puede verificar si los cores están correctamente aislados observando el comportamiento del sistema con herramientas como **htop** o **top** y monitoreando la carga en los cores especificados.

Este procedimiento ayuda a reservar recursos del sistema para tareas específicas, mejorando el rendimiento en sistemas donde la latencia o la predictibilidad son críticas.