

Control groups (*Cgroups*)

Contenidos

1. ¿Qué son los Cgroups ?
2. Un ejemplo de uso
3. Implementación
4. Uso de Cgroups
5. Uso de Cgroups con systemd
6. Ejemplos de parámetros de uso de recursos configurables mediante Cgroups
7. Bibliografía

1. ¿Qué son los *Cgroups*?

- *Cgroups* es la denominación de una funcionalidad de los sistemas Linux mediante la cual se pueden administrar los recursos que se asignan a conjuntos de procesos.
- Los procesos que forman parte de estos conjuntos poseen una relación jerárquica.
- Esta administración de recursos permite un control más fino del uso de los recursos del sistema:
 - Limitar el uso de los recursos utilizados por los procesos.
 - Priorizar el uso de los recursos.
 - Denegar el uso de recursos.
- El uso de *Cgroups* permite un uso más inteligente de los recursos con el fin de que el sistema funcione de una forma más eficiente y más segura.

2. Un ejemplo de uso

- El servicio **httpd** atiende cada solicitud de servicio creando un proceso hijo. Si no se controla el consumo de recursos que hace:
 - Por cada conexión de solicitud de servicio crearía un proceso hijo que consumiría recursos del sistema.
 - A medida que el número de conexiones de solicitud de servicio aumenta, el conjunto de procesos existentes generados por el servicio **httpd** iría acaparando cada vez más recursos del sistema.
 - A partir de un cierto número de conexiones, debido a este acaparamiento de recursos, se resentiría el rendimiento de todos los servicios que se ejecutan en el sistema, incluido el servicio **httpd**, que tardaría cada vez más en atender las solicitudes de servicio (por ejemplo, se tardaría cada vez más en descargar un contenido solicitado).
 - Si siguen aumentando las solicitudes de servicio, el propio sistema se haría inoperativo debido a que no sería capaz de atender a la carga de tareas que debe ejecutar.
- Sin embargo, si se controlan los recursos que el servicio **httpd** puede utilizar, estableciendo límites de uso mediante *Cgroups*, entonces el fenómeno anterior no se produciría. **La funcionalidad de Cgroups permite llevar a cabo este control en la asignación de recursos.**
- *Cgroups* no sólo se utiliza para evitar el acaparamiento de recursos por parte de un proceso, también se puede utilizar para garantizar que los procesos posean la cantidad de recursos mínima requerida para que se ejecuten adecuadamente.

3. Implementación

- La funcionalidad de *Cgroups* está soportada directamente por el núcleo del sistema operativo.
- Cada tipo de recurso que es capaz de controlar *Cgroups* tiene asociado un controlador que es cargado dinámicamente por **systemd**.
- Ejemplos de controladores disponibles en Red Hat:
 - **blkio**: permite limitar los accesos de E/S a disco.
 - **cpu**: controla el uso de la CPU a través de planificador de CPU del sistema.
 - **cpuset**: permite asignar CPUs individuales y áreas de memoria física.
 - **memory**: permite limitar el uso de la memoria.
 - **devices**: habilita o deshabilita el acceso a dispositivos.
 - **net_cls**: permite controlar el tráfico de red.

4. Uso de *Cgroups*

- Se puede manejar de dos maneras:
 - Usando el administrador de servicios del sistema: **systemd**.
 - Usando las herramientas que proporciona el paquete **libcgroup**.
- En distribuciones basadas en Red Hat, se recomienda utilizar **systemd** debido a que:
 - El paquete **libcgroup** se considera obsoleto.
 - Su uso puede producir conflictos con el manejo de **Cgroups** que hace **systemd**.

5. Uso de *Cgroups* con **systemd**

- Con **systemd**, los *Cgroups* se asocian a:
 - Unidades de tipo *Service*. Están formadas por conjunto de procesos que son iniciados por **systemd** a partir de unos archivos de configuración. Cada unidad de este tipo tiene un nombre con el formato *nombre.service*.
 - Unidades de tipo *Scope*. Están formadas por procesos que son iniciados por un proceso del sistema (no **systemd**) usando la función **fork()**. Por ejemplo: los procesos que se ejecutan pertenecientes a la sesión de un usuario. Cada unidad de este tipo tiene un nombre con el formato *nombre.scope*.
 - Unidades de tipo *Slice*. Unidades organizadas jerárquicamente (árbol) en la que se ubican unidades de los tipos anteriores. Cada unidad de este tipo tiene un nombre con el formato *nombre.slice*.

5. Uso de *Cgroups* con **systemd**

- El *Cgroup* que posee una unidad puede ser:
 - **Transitorio:**
 - Se crea desde la línea de órdenes usando la orden **systemd-run**.
 - La unidad y la configuración del *Cgroup* se puede administrar usando la orden **systemctl**.
 - El *Cgroup* permanece hasta la finalización de todos los procesos que forman parte de la unidad.
 - **Permanente:**
 - Se crea desde el momento que **systemd** inicia la unidad.
 - La configuración del *Cgroup* se especifica en los ficheros de configuración del servicio.
 - La unidad y la configuración del *Cgroup* se puede administrar usando la orden **systemctl**.

6. Ejemplos de parámetros de uso de recursos configurables mediante Cgroups

- Parámetros de CPU
 - CPUShare: establece la duración de la rodaja de uso de la CPU cuando se le asigna a algún proceso de la unidad.
 - CPUQuota: porcentaje máximo de CPU que puede consumir la unidad.
 - AllowedCPUs: establece qué CPUs pueden utilizar los procesos de la unidad.
- Control de tareas
 - TasksMax: establece el número máximo de tareas que pueden crearse en la unidad.
- Parámetros de memoria
 - MemoryLimit: establece la cantidad máxima de memoria que puede consumir la unidad.
 - MemoryMin: establece la cantidad mínima de memoria que el sistema debe intentar mantenerle a la unidad.
 - MemorySwapMax: establece el tamaño máximo del área swap que puede consumir la unidad.
- Parámetros de E/S
 - IOReadBandwidthMax: limita en ancho de banda máximo que puede consumir la unidad en operaciones de lectura de disco.
 - IOWriteBandwidthMax: limita en ancho de banda máximo que puede consumir la unidad en operaciones de escritura de disco.

7. Bibliografía

Resource Management Guide. Using cgroups to manage system resources on RHEL 7

https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/resource_management_guide/index