Gestión de Servicios con systemd

Uno de los aspectos principales gestionados mediante systemd son los **servicios**, o daemons, del sistema. Estos son todos los procesos que se inician automáticamente durante el arranque del mismo.

En el pasado la gestión de este aspecto recaía sobre **sysvinit**, un **modelo** de gestión de inicio de procesos clásico y estándar en los sistemas GNU/Linux y UNIX. Sin embargo este sistema adolece de una serie de carencias que systemd puede gestionar de modo más eficiente.

Como todo en systemd tenemos que definir units correspondientes a los elementos a administrar, en este caso los servicios. El modo de definición de units es a través de unit files, por tanto el aspecto fundamental es la definición del archivo de unit file correspondiente.

Gestión de inicio de units

- Inicio y parada de servicio de unit tipo service

```
systemctl start nginx.service
```

- Inicia servicio

```
systemctl stop nginx.service
```

- Para servicio

```
systemctl restart nginx.service
```

- Reinicia servicio

```
systemctl reload nginx.service
```

- Recarga configuración de servicio

```
systemctl reload-or-restart nginx.service
```

Se usaría este comando en lugar del anterior cuando no sepamos si el servicio puede recargar su configuración

- Enmascarar units

```
systemctl mask nginx.service
```

Con este comando enmascaramos la unit nginx.service lo que hará que esa unit no pueda ser iniciada. Puede ser un estado de inhabilitación temporal sin llegar a deshabilitarla por completo. Para volver a activarla

```
systemctl unmask nginx.service
```

- Habilitar y deshabilitar unit de servicio para que inicie durante el arranque del sistema

```
systemctl enable nginx.service
```

Habilita la unit correspondiente. Creará un link simbólico al service file, en /lib/systemd/system o /etc/systemd/system, en la ubicación en la que systemd ubica los archivos de inicio, por lo general en /etc/systemd/system/nombre_target.target.wants, ubicación que especifica para un target (runlevel) asociado los servicios que se inician en el mismo.

```
systemctl disable nginx.service
```

Deshabilita la unit

Compilando y definiendo servicio systemd para mariadb-server

Vamos a ver un ejemplo de creación de unit file para un sistema mariadb-server que compilaremos desde las fuentes. De este modo podremos definir y gestionar el servicio correspondiente

Ejecutamos los comandos

```
apt build-dep mariadb-server
cd /tmp
apt source mariadb-server
tar -xzf mariadb-10.1_10.1.26.orig.tar.gz
cd mariadb-10.1.26
BUILD/autorun.sh
./configure
make
make install
```

Ejecutamos los siguientes comandos para poder ejecutar mysql. Primero cambiamos el propietario del directorio de instalación de mysql

```
chown -R mysql /usr/local/mysql
```

a continuación creamos el directorio de datos para el servidor:

```
/usr/local/mysql/scripts/mysql_install_db --user=mysql
```

Tras el proceso de compilación y configuración anterior dispondremos de mariadb-server en nuestra máquina. Sin embargo no tenemos ningún sistema de gestión del servicio por defecto.

Para poder gestionar el servicio, y por tanto poder arrancar automática y manualmente el servidor mariadb, necesitamos un unit file de tipo service.

Definiendo unit file para mariadb-server

Creamos el unit file correspondiente en el archivo **/etc/sytemd/system/mariadb.service** con el siguiente contenido:

```
[Unit]
Description=MariaDB database server
After=network.target
After=syslog.target
[Service]
Type=simple
PrivateNetwork=false
User=mysql
Group=mysql
CapabilityBoundingSet=CAP_IPC_LOCK
PermissionsStartOnly=true
ExecStart=/usr/local/mysql/bin/mysqld_safe --basedir=/usr/local/mysql --datadir=/usr/local/mysql/data
ExecReload=/bin/kill -HUP $MAINPID
ExecStop=/bin/kill -QUIT $MAINPID
Restart=on-abort
RestartSec=5s
UMask=007
PrivateTmp=false
LimitNOFILE=16364
[Install]
WantedBy=multi-user.target
Alias=mysql.service
Alias=mysqld.service
```

tras definir el unit file activamos el servicio en systemd

```
systemctl enable mariadb.service
```

Al reiniciar la máquina debería arrancar automáticamente el servidor mariadb.

Para detener el servidor

```
systemctl stop mariadb.service
```

para arrancarlo

```
systemctl start mariadb.service
```

etc.

Volver

JavierFP 17:44 11 dec 2017 (CET)