

LPIC2 2021 Sesión 4 (2021-02-11) - systemd, journal

Curso

- Documentación relacionada 202:
 - Presentaciones/2020/201/202 - Arranque del Sistema.pdf
 - /Material Practicas LPIC-2/LPIC-201/3-System Startup/SystemD/Resumen Systemd-SystemV.pdf

SystemD

- levanta los servicios de manera paralela
- <https://wiki.archlinux.org/index.php/systemd>
- `systemd-cgls`: procesos en arbol
- `systemd-cgtop`
- `systemd-analyze`
 - `blame`: tiempos de arranque de los servicios, para ver quien es el lento
 - `plot` > `/var/www/html/plot.html`

Units

- Units:
 - `service`
 - `socket`
 - `device`
 - `mount`
 - `automount`
 - `target (runlevels)`
 - `snapshot`
- `cgroups`: gestión en caliente de CPU, RAM, DISCO
- **service** es unit por defecto (se puede omitir en los comandos)
- `systemctl <accion> <unidad>`
 - `start`
 - `stop`
 - `enable`: activa en el arranque
 - `enable -now`
 - `disable`
 - `reload`
 - `status`

targets

- los runlevels en systemV
- `cat /etc/inittab`
- `runlevelX.target` o `<NOMBRE>.target`
 - 0: `halt?`
 - 1: `rescue.target`
 - 2:
 - 3:
 - 4: `multi-user.target`

- 5: graphical.target
- 6: reboot.target
- `systemctl get-default`
- `systemctl set-default <TARGET>.target`
- `systemctl isolate <TARGET>.target` → init o telinit en systemV

snapshot

mask

- para e impide ejecución de un servicio (o unidad?)
- `systemctl mask nombre.service`
- `systemctl unmask nombre.service`

laboratorio1

- Material Practicas LPIC-2/LPIC-201/3-System Startup/SystemD/Laboratorio 1 Gestion de servicios con systemctl:
 - `systemctl list-units --type service`: servicios activos
 - `systemctl list-units --type service --all`: servicios activos e inactivos
 - `systemctl start|stop|restart|try-restart|reload|status|is-enabled|is-active nombre.service`: operaciones con servicios
 - `try-restart`: solo hace restart si está arrancado
 - `systemctl enable|disable nombre.servicio`: activar en arranque
 - `systemctl kill [-s SIGKILL] nombre.servicio`: parar servicios
 - `systemctl reboot|halt|poweroff`: reboot, halt
 - `systemctl mask|unmask`: enmascara un servicio, impide que se ponga en marcha

Laboratorio 2

- Material Practicas LPIC-2/LPIC-201/3-System Startup/SystemD/Laboratorios-2 Systemd:, página 9
 - units propias: **/etc/systemd/system/**
 - no funciona **mask**
 - **/usr/lib/systemd/system/**: unidades distribuidos con paquetes instalados
 - **/run/systemd/system/**: unidades creadas en tiempo de ejecución. Tiene tiene precedencia sobre el directorio anterior
 - **/etc/systemd/system/**: unidades creadas y administradas por el administrador del sistema
 - `systemctl cat <unit>`: muestra el contenido y ubicación del archivo
 - `systemctl edit --full <unit>`: editar la unidad
 - `systemd-delta`: ver cambios (extended) en las unidades
 - `systemctl daemon-reload`: al crear una UNIT, para que systemctl se entere.
 - Si no ponemos sección **[Install]** en nuestro fichero de unit, podemos asignarlo a un target creando un enlace simbólico en **/etc/systemd/system/multi-user.target.wants/**
 - `systemctl enable supervisamem.service` → Created symlink from `/etc/systemd/system/multi-user.target.wants/supervisamem.service` to `/etc/systemd/system/supervisamem.service`
 - https://www.doc-developpement-durable.org/file/Projets-informatiques/cours-&-manuels-informatiques/RedHat/Red_Hat_Enterprise_Linux-7-System_Administrators_Guide-en-US.pdf
 - archivos `.service` personalizados

```

▪ cd /opt/scripts
  chmod 700 supervisamem

```

```
chmod 700 /etc/init.d/iniciar-supervisamema
```

```
cd /etc/systemd/system
vi supervisamem.service
```

```
[Unit]
Description=Daemon of axample
[Service]
ExecStart=/etc/init.d/iniciar-supervisamem start
ExecStop=/etc/init.d/iniciar-supervisamem stop
Type=forking
[Install]
WantedBy=multi-user.target
```

```
systemctl daemon-reload
systemctl start supervisamem.service
tail -f /var/log/supervisamem.log
systemctl stop supervisamem

systemctl enable suspervisamem.service

systemd-cgtop
```

Journal

- Material Practicas LPIC-2/LPIC-201/3-System Startup/SystemD/Laboratorio journalctl.pdf
- journalctl
 - -xb: message explanation + current boot
 - -b [-n] | --list-boots: boots registrados
 - -f en tiempo real
 - -k filtra mensajes kernel
 - -n <n> los últimos <n>
 - _COMM=<programa>
 - _PID=<>
 - _UID=<>
 - -u <service>
 - filtrado por fechas:
 - --since
 - --until
 - **30 min ago**
 - **yesterday**
 - **2021-02-11 21:27**
 - -p: en función de su prioridad según 7 niveles
 - 0: emerg
 - 1: alert
 - 2: crit
 - 3: err
 - 4: warning
 - 5: notice
 - 6: info
 - 7: debug

- /dev/sda: mensajes de discos
- -disk-usage: espacio ocupado por los registros
- usermod -aG adm <usuario>: acceso a journal
- ≡ /var/log/message o /var/log/syslog
- persistencia: /var/log/journal (en centos7 no lo está)
 - usa el 10% de la partición como máximo
 - se puede cambiar en /etc/systemd/journald.conf
 - journalctl --vacuum-size=2G: limpia registros hasta 2GB
 - journalctl --vacuum-size=2years: limpia registros hasta 2 años

Tips & Tricks

*ctl

- loginctl

Ejecutar en systemd un script en Bash para que arranque como demonio al encender la máquina

- `vi /usr/bin/mail_arraque.sh`

- [mail_arraque.sh](#)

```
#!/bin/bash
echo se encendio el equipo $(hostname) a las $(date +%d-%m-%Y---%X) |
mail usuario@correo.es
```

- `chmod +x /usr/bin/mail_arraque.sh`
`vi /etc/systemd/system/mail_arraque.service`

- [mail_arraque.service](#)

```
[Unit]
Description= Envio hora de arranque
After=network.target

[Service]
ExecStart=/usr/bin/mail_arraque.sh
Type=oneshot
RemainAfterExit=true

[Install]
WantedBy=default.target

##RemainAfterExit=true hará que el servicio permanezca de acuerdo con
systemd, por lo que solo intentará ejecutarse la primera vez que ejecute
systemctl start, pero no la segunda.
```

- ```
systemctl daemon-reload
systemctl enable mail_arranque.service
systemctl start mail_arranque.service
```
- antes se ponía en: **/etc/profile.d**

## TODO

- revisar PDF journal

From:

<https://miguelangel.torresegea.es/wiki/> - **miguel angel torres egea**

Permanent link:

<https://miguelangel.torresegea.es/wiki/info:cursos:pue:lpic2-2021:s4>

Last update: **18/09/2022 23:45**

