Operating System fundamentals

Services and Daemons

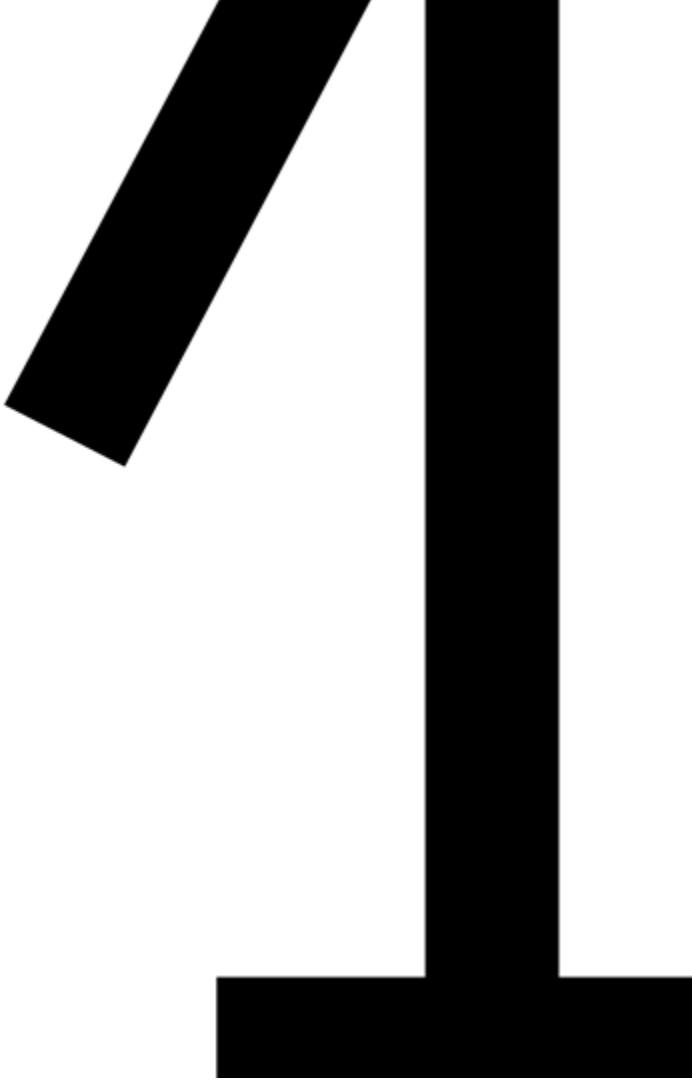
KdG Karel de Grote Hogeschool

Contents

Course text

- Chapter 14 Control Services and Daemons
  - (RedHat chapter 9)
  - Identify Automatically Started System Processes
  - Control System Services

Services and daemons



Services and daemons

- What happens when somebody uses ssh to login into a server?
  - connection with which port?
  - who “listens” to this port?
- other examples
  - webserver
  - printer server
  - file server
  - …
- the names of these **daemons** end often in “d” (sshd, lpd, …)
- a **service** is 1 or more daemons working together

Services and daemons

- How to start a service?
  - automatically by systemd
  - manually using systemctl (interacts with systemd)
- How to manage services?
  - with systemctl
  - with configuration files

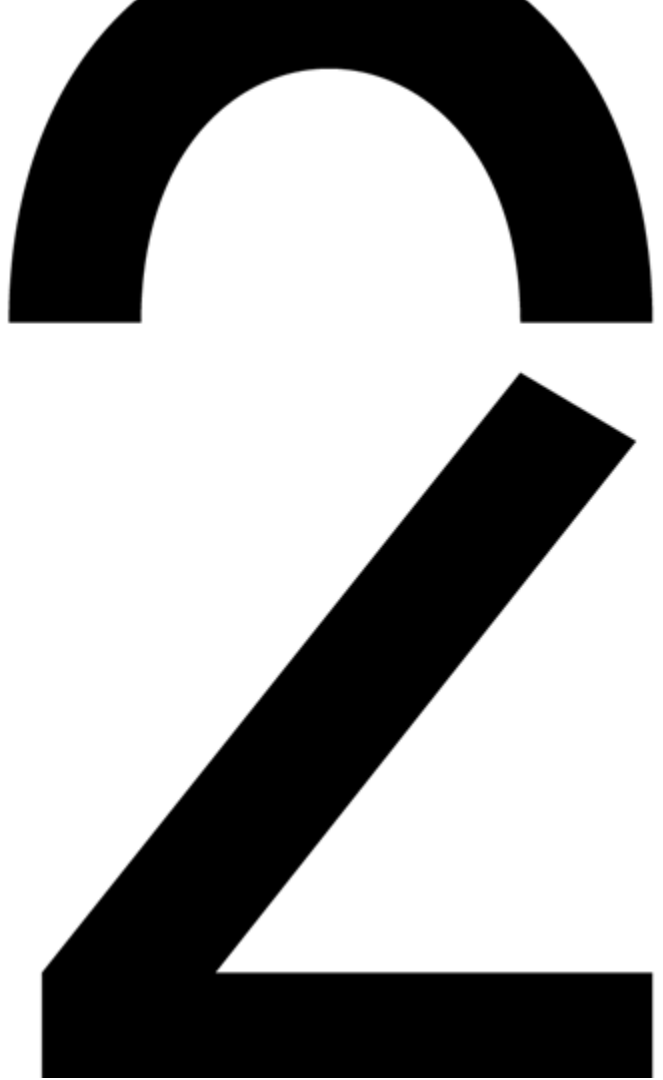
Systemd

- systemd is started by the kernel (PID = 1)
- systemd starts all services
  - in parallel (faster boot time)
  - starts the daemons
  - manages dependencies between services
  - manages the processes

Units

- define the objects that systemd should manage
- are configured using configuration files (“unit files”)
- have a name and a type
- types
  - service units: handles system services (e.g. webservice)
  - socket units: respond to socket connections (e.g. incoming ssh)
  - path units: respond to changes in files or directories (e.g. print a file when it is added to a certain folder)
- units are controlled through the systemctl command

Requesting information



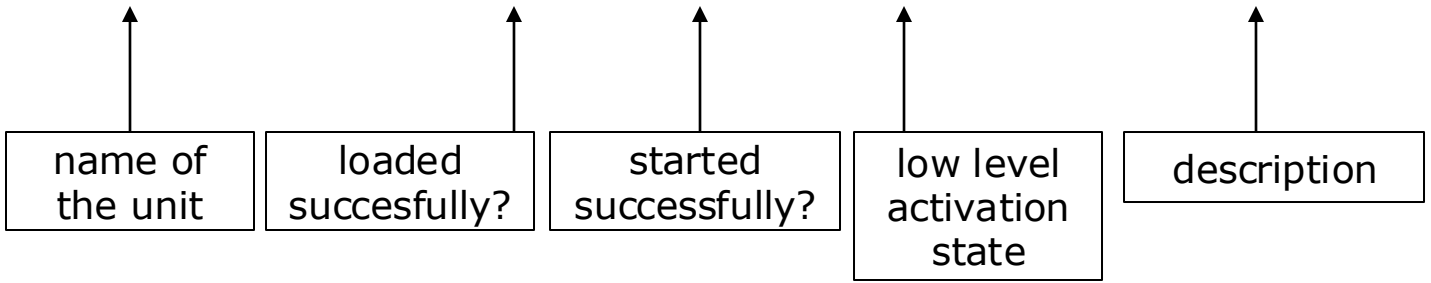
KdG
Karel de Grote Hogeschool

Service units overview

only show services

also show the non-active

```
[root@host ~]# systemctl list-units --type=service -all
UNIT                            LOAD      ACTIVE   SUB     DESCRIPTION
  atd.service                   loaded    active   running Job spooling tools
  auditd.service                loaded    active   running Security Auditing ...
  auth-rpcgss-module.service    loaded    inactive dead    Kernel Module ...
  chronyd.service               loaded    active   running NTP client/server
  cpupower.service              loaded    inactive dead    Configure CPU power ...
  crond.service                 loaded    active   running Command Scheduler
  dbus.service                  loaded    active   running D-Bus System Message Bus
● display-manager.service       not-found inactive dead    display-manager.service
```



Service units states

- a service can be in different states
  - loaded: configuration loaded into memory
  - active: the services processes are running
  - inactive: the processes are not running
  - enabled: the service automatically starts at boot time
  - disabled: the service does not automatically start at boot time
  - static: de service does not automatically start but could be started by another service that depends on this
  - …

Request service state

```
[root@host ~]# systemctl status sshd.service
● sshd.service - OpenSSH server daemon
     Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2022-03-14 05:38:12 EDT; 25min ago
       Docs: man:sshd(8)
             man:sshd_config(5)
   Main PID: 1114 (sshd)
      Tasks: 1 (limit: 35578)
     Memory: 5.2M
        CPU: 64ms
     CGroup: /system.slice/sshd.service
             └─1114 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Mar 14 05:38:12 workstation systemd[1]: Starting OpenSSH server daemon...
Mar 14 05:38:12 workstation sshd[1114]: Server listening on 0.0.0.0 port 22.
Mar 14 05:38:12 workstation sshd[1114]: Server listening on :: port 22.
Mar 14 05:38:12 workstation systemd[1]: Started OpenSSH server daemon.
...output omitted...
```

Request service state

- You can check a service state using:
  - **systemctl is-active** sshd.service
  - **systemctl is-enabled** sshd.service
  - **systemctl is-failed** sshd.service

Manage services

Manage services

- starting and stopping
  - **systemctl start** sshd
  - **systemctl stop** sshd.service
- restarting and reloading
  - **systemctl restart** sshd.service (stops and starts the service)
  - **systemctl reload** sshd.service (reloads configuration)
  - **systemctl reload-or-restart** sshd.service
    (reloads configuration and restarts if the reload fails)

Manage services

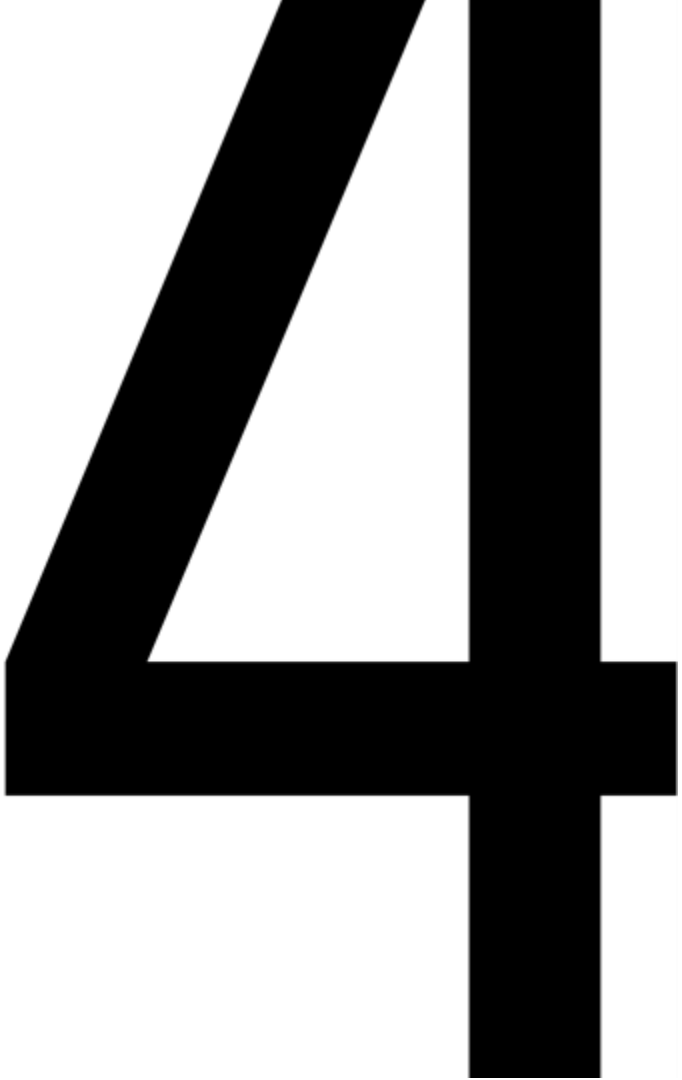
- start and stop at boot time
  - **systemctl enable** sshd.service (start at boot time)
  - **systemctl enable --now** sshd.service (start at boot time and also now)
  - **systemctl disable --now** sshd.service
    (don't start at boot time and stop it immediately)
- completely block a service
  - **systemctl mask** sshd.service
    (service cannot be started automatically nor manually)
  - **systemctl unmask** sshd.service

Dependencies between services

```
[root@host ~]# systemctl list-dependencies
sshd.service
sshd.service
● ├─system.slice
● ├─sshd-keygen.target
● │ ├─sshd-keygen@ecdsa.service
● │ ├─sshd-keygen@ed25519.service
● │ └─sshd-keygen@rsa.service
● └─sysinit.target
...output omitted...
```

KdG
Karel de Grote Hogeschool

Exercises



Exercises

- KdG
  - …
- RedHat
  - ch09s02
  - ch09s04
  - ch09s05

KdG Karel de Grote Hogeschool