

RHSA (Admin 1) Red Hat Enterprise Linux 8 Essentials

Prepared by Eng. Taghreed Elsayed

Lecture 6

جامعة الجميع الذكية

تسعى لتكون الكيان الأول الذي يتبنى مبدأ المجتمع التعليمي
التعاوني ويقدم العلم مقابل العمل لتنمية وبناء المجتمعات.

CONTROLLING SERVICES AND DAEMONS



جامعة الجميع الذكية
EVERYONE'S SMART UNIVERSITY

IDENTIFYING AUTOMATICALLY STARTED SYSTEM PROCESSES

- ❑ **Daemons** are processes that wait or run in the background performing various tasks.
- ❑ Generally, **daemons** start automatically at boot time and continue to run until shutdown or until they are manually stopped. By convention, the names of many daemon programs end in the letter "d".
- ❑ A **service** often refers to one or more daemons, but starting or stopping a service may instead make a one-time change to the state of the system, which does not involve leaving a daemon process running afterward (called oneshot).
- ❑ **systemctl and systemd** units The systemctl command is used to manage different types of systemd objects, called units. A list of available unit types can be displayed with systemctl -t help.



Service states

- ❖ The status of a service can be viewed with `systemctl status name.type`.
- ❖ If the unit type is not provided, `systemctl` will show the status of a service unit, if one exists.

```
[root@serverX ~]# systemctl status sshd.service
sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled)
   Active: active (running) since Thu 2014-02-27 11:51:39 EST; 7h ago
 Main PID: 1073 (sshd)
   CGroup: /system.slice/sshd.service
           └─1073 /usr/sbin/sshd -D

Feb 27 11:51:39 server0.example.com systemd[1]: Started OpenSSH server daemon.
Feb 27 11:51:39 server0.example.com sshd[1073]: Could not load host key: /et...y
Feb 27 11:51:39 server0.example.com sshd[1073]: Server listening on 0.0.0.0 ....
Feb 27 11:51:39 server0.example.com sshd[1073]: Server listening on :: port 22.
Feb 27 11:53:21 server0.example.com sshd[1270]: error: Could not load host k...y
Feb 27 11:53:22 server0.example.com sshd[1270]: Accepted password for root f...2
Hint: Some lines were ellipsized, use -l to show in full.
```

Several keywords indicating the state of the service can be found in the status output:

KEYWORD:	DESCRIPTION:
loaded	Unit configuration file has been processed.
active (running)	Running with one or more continuing processes.
active (exited)	Successfully completed a one-time configuration.
active (waiting)	Running but waiting for an event.

KEYWORD:	DESCRIPTION:
inactive	Not running.
enabled	Will be started at boot time.
disabled	Will not be started at boot time.
static	Can not be enabled, but may be started by an enabled unit automatically.

LISTING UNIT FILES WITH `systemctl` :

NOTE Notice that the `systemctl` command will automatically paginate the output with `less`.

❑ 1. List all service units on the system

```
[student@serverX ~]$  
systemctl list-units --type=service
```

❑ 2. List all socket units, active and inactive, on the system.

```
[student@serverX ~]$  
systemctl list-units --type=socket --all
```

❑ 3. Explore the status of the chronyd service. This service is used for network time synchronization (NTP).

3.1. Display the status of the chronyd service. Note the process ID of any active daemons.

CONTROLLING SERVICES AND DAEMONS

```
[student@serverX ~]$  
systemctl status chronyd
```

3.2. Confirm that the listed daemons are running.

```
[student@serverX ~]$  
ps -p PID
```

4. Explore the status of the sshd service. This service is used for secure encrypted communication between systems.

4.1. Determine if the sshd service is enabled to start at system boot.

```
[student@serverX ~]$  
systemctl is-enabled sshd
```



4.2. Determine if the sshd service is active without displaying all of the status information.

```
[student@serverX ~]$  
systemctl is-active sshd
```

4.3. Display the status of the sshd service.

```
[student@serverX ~]$  
systemctl status sshd
```

5. List the enabled or disabled states of all service units.

```
[student@serverX ~]$  
systemctl list-unit-files --type=service
```



CONTROLLING SYSTEM SERVICES

➤ To control system daemons and network services using `systemctl`.

✓ STARTING AND STOPPING SYSTEM DAEMONS ON A RUNNING SYSTEM

→ Changes to a configuration file or other updates to a service may require that the service be restarted. A service that is no longer used may be stopped before removing the software. A service that is not frequently used may be manually started by an administrator only when it is needed.

❑ Managing services on a running system.

- ❑ 1. View the status of a service.

```
[root@serverX ~]#  
systemctl status sshd.service
```

2. Verify that the process is running.

```
[root@serverX ~]#  
ps -up PID
```

3. Stop the service and verify the status.

```
[root@serverX ~]#  
systemctl stop sshd.service  
[root@serverX ~]#  
systemctl status sshd.service
```

4. Start the service and view the status. The process ID has changed.

```
[root@serverX ~]#  
systemctl start sshd.service  
[root@serverX ~]#  
systemctl status sshd.service
```

5. Stop, then start, the service in a single command

```
[root@serverX ~]#  
systemctl restart sshd.service  
[root@serverX ~]#  
systemctl status sshd.service
```

6. Issue instructions for a service to read and reload its configuration file without a complete stop and start. The process ID will not change.

```
[root@serverX ~]#  
systemctl reload sshd.service  
[root@serverX ~]#  
systemctl status sshd.service
```



Unit dependencies

- ❑ For example, a file placed into the print spool directory will cause the cups print service to be started if it is not running.

```
[root@serverX ~]# systemctl stop cups.service
Warning: Stopping cups, but it can still be activated by:
  cups.path
  cups.socket
```

- To completely stop printing services on a system, stop all three units. Disabling the service will disable the dependencies.
- The **systemctl list-dependencies UNIT** command can be used to print out a tree of what other units must be started if the specified unit is started.
- Depending on the exact dependency, the other unit may need to be running before or after the specified unit starts.
- The **--reverse** option to this command will show what units need to have the specified unit started in order to run.

ENABLING SYSTEM DAEMONS TO START OR STOP AT BOOT

- ❖ Starting a service on a running system does not guarantee that the service will be started when the system reboots.
- ❖ **Similarly, stopping** a service on a running system will not keep it from starting again when the system reboots.
- ❖ Services are started at boot time when links are created in the appropriate **systemd** configuration directories.
- ❖ These links are created and removed with **systemctl** commands.

✓ **Enabling and disabling services.**

1. View the status of a service.

```
[root@serverX ~]#  
systemctl status sshd.service
```

2. Disable the service and verify the status. Note that disabling a service does not stop the service.

```
[root@serverX ~]#  
systemctl disable sshd.service  
[root@serverX ~]#  
systemctl status sshd.service
```

3. Enable the service and verify the status.

```
[root@serverX ~]#  
systemctl enable sshd.service  
[root@serverX ~]#  
systemctl is-enabled sshd.service
```



SUMMARY OF **systemctl** COMMANDS

TASK:	COMMAND:
Stop a service on a running system.	<code>systemctl stop <i>UNIT</i></code>
Start a service on a running system.	<code>systemctl start <i>UNIT</i></code>
Restart a service on a running system.	<code>systemctl restart <i>UNIT</i></code>
Reload configuration file of a running service.	<code>systemctl reload <i>UNIT</i></code>
Completely disable a service from being started, both manually and at boot.	<code>systemctl mask <i>UNIT</i></code>
Make a masked service available.	<code>systemctl unmask <i>UNIT</i></code>
Configure a service to start at boot time.	<code>systemctl enable <i>UNIT</i></code>
Disable a service from starting at boot time.	<code>systemctl disable <i>UNIT</i></code>
List units which are required and wanted by the specified unit.	<code>systemctl list-dependencies <i>UNIT</i></code>
View detailed information about a unit state.	<code>systemctl status <i>UNIT</i></code>

USING **systemctl** TO MANAGE SERVICES

➤ **The chronyd service is disabled and no longer running on the system.**

1. Observe the results of **systemctl restart** and **systemctl reload** commands.
 - 1.1. Display the status of the **sshd** service. Note the process ID of the daemon

```
[student@serverX ~]$  
systemctl status sshd
```

- 1.2. Restart the **sshd** service and view the status. The **process ID** of the daemon has changed.

```
[student@serverX ~]$  
sudo systemctl restart sshd  
[student@serverX ~]$  
systemctl status sshd
```

2. Verify that the **chronyd** service is running.

```
[student@serverX ~]$  
systemctl status chronyd
```

3. Stop the **chronyd** service and view the status.

```
[student@serverX ~]$  
sudo systemctl stop chronyd  
[student@serverX ~]$  
systemctl status chronyd
```

4. Determine if the **chronyd** service is enabled to start at system boot.

```
[student@serverX ~]$  
systemctl is-enabled chronyd
```

5. Reboot the system, then view the status of the **chronyd** service.

```
[student@serverX ~]$  
systemctl status chronyd
```

6. Disable the **chronyd** service so that it does not start at system boot, then view the status of the service.

```
[student@serverX ~]$  
sudo systemctl disable chronyd  
[student@serverX ~]$  
systemctl status chronyd
```



7. Reboot the system, then view the status of the chronyd service.

```
[student@serverX ~]$  
systemctl status chronyd
```



جامعة الجميع الذكية
EVERYONE'S SMART UNIVERSITY

