

Systemd is the init system used in modern Linux distributions for managing services, processes, logs, and boot configurations. Below is a cheat sheet with essential systemd commands.

Table of Contents



- [Managing Services](#)
- [Checking System Boot and Performance](#)
- [Managing System Startup \(Targets\)](#)
- [Managing Systemd Unit Files](#)
- [Viewing System Logs \(Journalctl\)](#)
- [Example: Create a Custom Systemd Service](#)
 - [Create a new service file](#)
 - [Add the following configuration](#)
 - [Enable and Start the Service](#)
- [Other Useful Commands](#)
- [Summary](#)

Managing Services

Command	Description
<code>systemctl start <service></code>	Start a service.
<code>systemctl stop <service></code>	Stop a service.
<code>systemctl restart <service></code>	Restart a service.
<code>systemctl reload <service></code>	Reload configuration without restarting.
<code>systemctl status <service></code>	Check service status.
<code>systemctl enable <service></code>	Enable service to start at boot.
<code>systemctl disable <service></code>	Disable service from starting at boot.
<code>systemctl is-enabled <service></code>	Check if a service is enabled.
<code>systemctl mask <service></code>	Prevent a service from starting.
<code>systemctl unmask <service></code>	Unmask (re-enable) a masked service.

Example: Restart and Enable Apache

```
sudo systemctl restart apache2  
sudo systemctl enable apache2
```

Checking System Boot and Performance

Command	Description
<code>systemctl list-units -type=service</code>	List all running services.
<code>systemctl list-units -failed</code>	Show failed services.
<code>systemctl list-timers</code>	Show scheduled system timers (cron alternative).
<code>systemd-analyze</code>	Show boot time analysis.
<code>systemd-analyze blame</code>	List services causing slow boot.

Example: Check Slow Boot Services

```
systemd-analyze blame
```

Managing System Startup (Targets)

Systemd uses “targets” instead of traditional runlevels.

Command	Description
<code>systemctl get-default</code>	Show current default target.
<code>systemctl set-default multi-user.target</code>	Set system to boot into multi-user mode (no GUI).
<code>systemctl set-default graphical.target</code>	Set system to boot into GUI mode.
<code>systemctl isolate rescue.target</code>	Switch to rescue mode (single-user mode).

Example: Boot into Text Mode

```
sudo systemctl set-default multi-user.target
```

Managing Systemd Unit Files

Systemd services are defined in unit files stored in:

- /etc/systemd/system/ (custom services)
- /lib/systemd/system/ (system-installed services)

Command	Description
<code>systemctl cat <service></code>	Show service unit file.
<code>systemctl edit -full <service></code>	Edit service unit file.
<code>systemctl daemon-reload</code>	Reload systemd after editing unit files.

Example: Edit and Reload a Service

```
sudo systemctl edit --full myservice
sudo systemctl daemon-reload
sudo systemctl restart myservice
```

Viewing System Logs (Journalctl)

Systemd uses journald to store logs.

Command	Description
<code>journalctl -xe</code>	Show recent logs with errors.
<code>journalctl -u <service></code>	View logs for a specific service.
<code>journalctl -since "1 hour ago"</code>	View logs from the last hour.
<code>journalctl -b</code>	Show logs from the last boot.
<code>journalctl -disk-usage</code>	Check journal log size.

Example: View Apache Logs

```
journalctl -u apache2 --since "yesterday"
```

Example: Create a Custom Systemd Service

Create a new service file

```
sudo nano /etc/systemd/system/myapp.service
```

Add the following configuration

```
[Unit]
Description=My Custom Application
After=network.target

[Service]
ExecStart=/usr/bin/python3 /home/user/myapp.py
Restart=always
User=user
Group=user

[Install]
WantedBy=multi-user.target
```

Enable and Start the Service

```
sudo systemctl daemon-reload
sudo systemctl enable myapp.service
sudo systemctl start myapp.service
sudo systemctl status myapp.service
```

Other Useful Commands

Command	Description
hostnamectl	Show system hostname and OS details.
timedatectl	Show and set system time & timezones.
localectl	Display system locale settings.
loginctl	Manage user sessions.

Summary

- systemctl manages services and targets.
- journalctl handles logs.
- Systemd replaces cron for scheduled tasks (using timers).
- Custom unit files allow custom services.

Systemd is a powerful tool—this cheat sheet helps simplify managing your Linux system!