Filename: comptia-linuxxk0004-8-2-1-managing_services_with_systemd
Show Name: CompTIA Linux+ (XK0-004)
Topic: Managing System Components
Episode Name: Managing Services with Systemd
Description: In this episode, Zach and Don explore systemd, the program responsible for managing hardware and services in many popular Linux distros. They highlight how to determine if you are running systemd and then show how to leverage commands like systemctl to configure and support services under systemd.

Managing Services with Systemd

[?] How are services controlled under Linux?

- SysVinit
  - Series of scripts that run at boot time
  - Runlevel determines the scripts
  - Upstart was a short-lived upgrade to SysVinit
- systemd
  - Replaces old init system
  - Improvements
    - Runs tasks in parallel
    - Uses cgroups to track and isolate processes
    - Provides better support for hot-plug hardware

[?] How do we know if we are using systemd?

- Almost all modern distros use systemd
  - Exceptions
    - Slackware
  - Process ID #1
    - `ps aux`
    - `init`
    - `systemd`
  - `ls -l /sbin/init`

[?] So systemd starts right after the kernel, and then what happens?

- Daemons
  - Services/Applications
  - Run in the background
- Unit Files
  - Define the service
  - `/lib/systemd/system/`
- Can be overridden
  - `/etc/systemd/system/`

[?] Do we have to create unit files for new services?

- `systemctl`
  - Manages installing/removing unit files
  - Also allows starting/stopping daemons

- Example

    1. `sudo yum install httpd`
    2. `sudo systemctl start httpd`
    3. `sudo systemctl status httpd`
    4. `sudo systemctl enable httpd`
    5. `sudo systemctl disable httpd`

[?] How does systemd keep track of which services to run?

- Targets
    - Define a collection of units to execute
    - Establishes dependencies and a hierarchy
    - `/lib/systemd/system/graphical.target`

[?] Can we change between targets?

- Switch to CLI only
    - `systemctl isolate multi-user.target`
- Switch back to GUI
    - `systemctl isolate graphical.target`