System D

The Linux Apocalypse

Philosophy

- Conceptionally, systemd is based on two ideas.
- The prinicpal of least privilege.
- Containerization.
- The cgroup mechanism is used to enforce both these principals.
-

Philospohy Part 2

- cgroup originally was created to create linux containers. It is still used for this.
- Linux containers was trying to create a system similar to Solaris containers, or AIX LVMs.
- The least privilege principal is implemented by the process tracking mechanism in cgroup.

History

- The init.d system has been seen as a problem since its inception.
- It was always intended to be replaced.
- It is a simple extension of the rc.local mechanism.
- It has no policy enforcement mechanisms.

History: An example

- PLAN 9, the 'successor' of unix in the minds of its creators, had a very different init process (which unfortunately they also called init).
- The Plan 9 init process is important, because many of the successors that came afterwards incorporated certain ideas, for example, setting up namespaces, control of privileges.
- Plan 9 and its init system came out in 1989.

The Cheat Sheet

- For a given Application:
- To start:
- systemctl start <app>
- To stop:
- systemctl stop <app>

Systemd Config Example

- This is the postgresql systemd config file

```
[Unit]

Description=PostgreSQL database server

After=network.target

[Service]

Type=forking

User=postgres

Group=postgres

Environment=PGPORT=5432

Environment=PGDATA=/var/lib/pgsql/data

OOMScoreAdjust=-1000
```

Sytemd Config Example

```
ExecStartPre=/usr/bin/postgresql-check-db-dir ${PGDATA}

ExecStart=/usr/bin/pg_ctl start -D ${PGDATA} -s -o "-p ${PGPORT}" -w -t 300

ExecStop=/usr/bin/pg_ctl stop -D ${PGDATA} -s -m fast

ExecReload=/usr/bin/pg_ctl reload -D ${PGDATA} -s

TimeoutSec=300

[Install]

WantedBy=multi-user.target
```

Systemd Mechanisms

- Consoled
- It is intended to replace virtual terminals.
- It was introduced in 2014.
- It works.
- It replaced the tty mechanism.

Systemd Mechanisms

- Logind
- This is an attempt to centralize the back end login system.
- Xdm, gdm, kdm and a host of other login systems bought into this.

Systemd Mechanism

- Networkd
- Networkd is daemon that handles configuration of network interfaces. It handles, static interfaces, brdiged interface, dHCP interfaces for ipv4 and ipv6.

Systemd Mechanism

- Timedated
- My opinion was that this was thrown in as a good idea to quell a majority of people who are annoyed with multiple date time formats, and their incompatibilities.
- Most programs as of yet don't use its facilities.

Systemd Mechanism

- Libudev
- This is the standard library to utilize the udev system.
- This predates systemd by several years. It was created to solve what was seen as a problem with accessibility to udev, and specifically to avoid multiple incompatible methods of accessing udev.

Systemd Mechanism

- Udevd
- This is really the origins of systemd.
- It the device manager for linux. It handles the loading and unloading of devices, the creation of device files in /dev, and the updates of firmware for devices.

Systemd Mechanism

- Journald
- This is the subsystem that caused much controversy.
- The main objection was that it created binary log files in an obtuse and easily obfuscatable format.
- It also added nothing that wasn't there in syslog-ng and rsyslog.

The Problem Children

- There are a few applications that can't (or won't) be ported to systemd.
- The Two big ones are Db2 and Oracle RDBMS
- MQSeries is not systemd friendly.
- RabbitMQ, and other AMQP applications are not systemd ready.

The Problem Children

- DB2 has integrated the user/group system in unix into its own user authentication system, and the enforced principal of least privilege causes problems. It would need to be removed, and integrated using logind.
- Oracle has integrated device management that does not play nice with udevd.

Conclusions

- eh.