

# Debian default init: sysvinit -> systemd

佐々木洋平

[uwabami@gfd-dennou.org](mailto:uwabami@gfd-dennou.org)

EPNetFaN 2015 座学編

2015-06-11



はじ  
めに



# はじめに/宣伝

- 今日は「Debian 8」の話は(あまり)しません
- OSC 2015 Hokkaido
  - 今週末, 札幌コンベンションセンター
  - セッション
    - 6/13(土) 11:00 - 11:45
    - タイトル 「8,9,10 = Jessie,Stretch,Buster」
    - 詳細: 「OSC 2015 Hokkaido」で検索

そんな  
こんなで

今日の  
お題

**systemd**



# お品書き

- init の役割
- これまでの init: SysV init
- これからの init: systemd
- systemd でできること/できないこと(?)



# お品書き

## ■ **init** の役割

■ これまでの init: SysV init

■ これからの init: systemd

■ systemd でできること/できないこと(?)

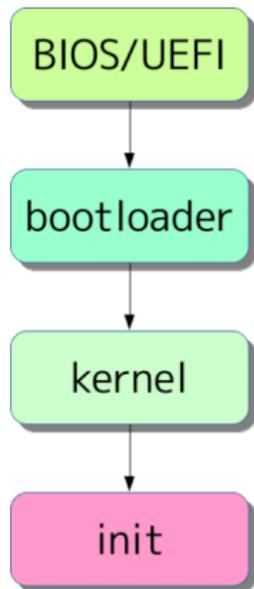
**init** の

**役割**



# Linux の boot sequence

1. 電源投入⇒BIOS or UEFI が起動
2. BIOS or UEFI ⇒ bootloader を起動
3. bootlader が kernel を起動
4. kernel が **init** を起動





# init とその役割

- init: kernel が最初に呼ぶプログラム(の総称)
- init がすること
  - ファイルシステムのマウント, ネットワーク設定, サービスの開始
  - サービスの管理(起動, 停止, 再起動)
  - カーネルからのイベントの受け取りとユーザランドでの処理の実行(例: Ctrl-Alt-Delete ⇒ 再起動)



# お品書き

- init の役割
- **これまでの init: SysV init**
- これからの init: systemd
- systemd でできること/できないこと(?)

**sysv  
init**



# SysV init

## SysV init = SystemV 由来の init

- ✓ init が初回に行なうプロセスの総称
- ✓ /sbin/init バイナリそのもの(文脈に注意)

## SysV init 由来の概念

- ✓ **runlevel**: 0(halt), 1(single user mode), 6(reboot), 等



# Debian の sysvinit(1)

## sysvinit の設定ファイル群と初回起動時の挙動

- ✓ /etc/inittab: runlevel **N**one でシステムを初期化
- ✓ /etc/rcS.d: ランレベル **S**ingle でハードウェアを初期化
- ✓ /etc/rc**X**.d: 2-5 のいずれかの runlevel でサービスを起動 (通常は runlevel 2 で動作する)



# Debian の sysvinit(2)

## Squeeze の sysvinit

- ✓ 単純な SysV init スタイル. sh script を**逐次実行**

## Wheezy の sysvinit

- ✓ LSB(Linux Standard Base) ヘッダーで定義された依存関係に従い, sh script を**並列に実行** ⇒ 高速に起動
- ✓ 依存関係が間違っていると挙動不審



# LSB ヘッダの例: dovecot

```
#!/bin/sh
### BEGIN INIT INFO
# Provides:          dovecot
# Required-Start:    $local_fs $remote_fs $network $syslog $time
# Required-Stop:     $local_fs $remote_fs $network $syslog
# Should-Start:      postgresql mysql slapd winbind nslcd
# Should-Stop:       postgresql mysql slapd winbind nslcd
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: Dovecot init script
# Description:       Init script for dovecot services
### END INIT INFO
```



# 脱線: サービスの起動/停止

- 多くの場合スクリプトを直接叩きがち
- なるべく service を使うこと

```
# /etc/init.d/dovecot stop <-- これは駄目  
# service dovecot stop <-- こっちの方が良い
```

- 何故=>**環境変数, ファイルシステムが同じ保証は無い**



# SysV init じゃ駄目なの？

1. SysVinit によるサービス起動の複雑さ
  - 環境の切り替え, 他のサービスへの依存...etc.
  - ⇒ **うまくやる**のが難しい(できなくはないけれど)
2. ファイルベースの PID 管理
  - ⇒ **SysV init は PID を管理していない.**
  - ファイルに吐いているだけ.
  - PID の重複チェックは? 孫プロセスは?
3. sh に起因するアレコレ: **遅い**



# お品書き

- init の役割
- これまでの init: SysV init
- **これからの init: systemd**
- systemd でできること/できないこと(?)

**systemd**



# systemd とは?

## systemd: not system"D" or "S"ystemd

- ✓ (メイン)開発者: Lennart Poettering@Red Hat
- ✓ License: LGPL-2.1+

## 特徴

- ✓ cgroups によるプロセス管理, 積極的な並列動作
- ✓ sh より簡便な設定, 詳細なドキュメント...etc.



# cgroups とは?

- Control Groups の略
- Linux Kernel の新機能
  - プロセスのグループ化, リソース制限, 名前ベースの分離
  - LXC(Linux Container), Docker  
= cgroups + namespace での分離 (+ 管理ツール)



# systemd のプロセス管理

## 全ての「サービス」は独自の cgroup に所属

- ✓ 「サービス」に関連する全てのプロセスを管理可能  
(SysV init では PID ファイルでしか管理できなかった)
- ✓ 「サービス」の「リソース」「権限」も管理可能  
(リソースの制限は ulimit 等で指定するしかなかった)



# systemd-{cgls,cgtop}

## systemd-cgls

- ✓ cgroup の階層構造を表示

## systemd-cgtop

- ✓ cgroup のリソース消費を表示



# systemd: サービス管理

- 個々の「サービス」は「Unit」ファイルで記述
- Unit で定義された「サービス」を systemctl で操作

```
# systemctl start dovecot    <-- service dovecot start
# systemctl start dovecot.service <-- (.service) は省略可能)
# systemctl stop dovecot     <-- service dovecot stop
# systemctl restart dovecot  <-- service dovecot restart
```



# systemd: runlevel

- systemd で runlevel に相当する概念は **targets**
  - 例: /etc/systemd/system/graphical.target.wants
- 各サービスをターゲット下に symlink を作成して管理

```
# systemctl enable/disable SERVICE <-- symlink の作成/削除  
# systemctl mask/unmask SERVICE <-- /dev/null への symlink
```



# bootup(7)



# boot の解析: systemd-analyze

- ユニット定義が正しければ積極的に並列動作: 高速
- boot 時の解析コマンド: systemd-analyze

```
# systemd-analyze <-- 起動時間の表示
# systemd-analyze blame <-- 遅いユニットの表示
# systemd-analyze plot <-- グラフ化
# systemd-analyze critical-chain <-- 一番時間がかかっている
                                ユニットの解析
```



# システムの状態表示: systemctl

```
% systemctl list-units          <-- systemctl だけでも良い  
% systemctl -t service         <-- サービスのみの表示  
% systemctl --failed           <-- 起動に失敗したサービスの表示  
% systemctl status             <-- 全体の状況を表示  
% systemctl status SERVICE     <-- 特定のサービスの表示  
# systemctl status SERVICE -l  <-- 詳細表示
```



# ユニットファイルの書き方

- atd を例に
- sysvinit の場合: /etc/init.d/atd
- systemctl では?
  - systemd.unit(5)
  - systemd.service(5)
  - systemd.exec(5)



# 例: atd.service

```
[Unit]
Description=Deferred execution scheduler
Documentation=man:atd(8)

[Service]
ExecStart=/usr/sbin/atd -f
IgnoreSIGPIPE=false

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



# Unit ファイルの置き場所

- パッケージの場合:  
⇒ /lib/systemd/system/ 以下に置かれる
- 個別に設定したい or 修正したい場合:  
⇒ /etc/systemd/system/ 以下に置く



# SysV init の sh は?

- (今のところ) systemd-sysv が対応
  - LSB ヘッダの解析
  - init-functions の置き換え
- 複雑な事をしないサービス(デーモン)は, そのまま動く
  - とはいえ, sh を調整するのは苦痛
  - Unit -> sh の変換は容易に可能なので...



# これからのinit: systemd

- systemd
  - cgroups によるプロセス管理
  - 積極的な並列動作
  - 「Unit」による簡便な設定
  - 詳細なドキュメント



# 脱線: 他の OS は?

- Solalis: **SMF** Service Management Facility
  - カーネル空間でのプロセス管理: contract
- Mac OS X (>= 10.4): **launchd**
  - init, inetd, atd, crond 等の機能を単一のバイナリで提供
  - 2006年8月に Apache 2.0 で公開
- 両者とも, XML で設定ファイルを記述



## 脱線: その2

- Debian で sysvinit を使い続けるには?  
⇒ **syvinit-core** をインストールすると良い  
互換性維持のため, systemd-shim も入れておく  
と吉
- ただし, GNOME を使うなら systemd 推奨
  - Desktop の場合には systemd 使った方が...
- LXC や Docker を使うなら systemd の方が良い?
  - jessie on wheezy は今のところ駄目(注意)



# お品書き

- init の役割
- これまでの init: SysV init
- これからの init: systemd
- **systemd** でできること/できないこと(?)



# All your cgroups are belong to us

- inittab -> consoled, logind, locale
- network -> systemd-networkd
- syslog -> journald
- ntp -> timedated, timesyncd
- cron -> timer.targets



# systemd components

## systemd Utilities

systemctl journalctl notify analyze cgls cgtop loginctl nspawn

## systemd Daemons

systemd  
journald networkd  
logind user session

## systemd Targets

bootmode basic multi-user graphical user-session  
shutdown reboot dbus telephony display service  
dlog logind user-session tizen service

## systemd Core

manager unit login namespace log  
service timer mount target multiseat inhibit  
systemd snapshot path socket swap session pam cgroup dbus

## systemd Libraries

dbus-1 libpam libcap libcryptsetup tcpwrapper libaudit libnotify

## Linux Kernel

cgroups autofs kdbus

["http://ja.wikipedia.org/wiki/Systemd"](http://ja.wikipedia.org/wiki/Systemd)



# 醜聞

- Fscck SystemD and its developers and its users:  
<https://lists.debian.org/debian-ctte/2014/02/msg00376.html>
- Linus 様が systemd にぶちきれ:
- 「魚は頭から腐る」--systemdの開発者がトーバルズ氏を公然と批判:  
<http://cpplover.blogspot.jp/2014/04/linussystemd.html>
- 「魚は頭から腐る」--systemdの開発者がトーバルズ氏を公然と批判:  
<http://japan.zdnet.com/article/35054778/>

個人的  
見解



# まとめ

- Debian 8 から default の init は systemd になりました
  - 高速起動, 管理・設定しやすい方式に
  - 柔軟なプロセス/リソース管理が可能に
    - sysvinit も残ります(選択の自由).
  - ~~RHEL7がバグを踏み潰してくれでしょう~~
- 別の init が表れる可能性もあるでしょう



# 参考文献

- Debian リファレンス:  
<https://www.debian.org/doc/manuals/debian-reference/index.ja.html>
- Wikipedia: systemd  
<http://ja.wikipedia.org/wiki/Systemd>
- Fscck SystemD and its developers and its users:  
<https://lists.debian.org/debian-ctte/2014/02/msg00376.html>
- Linus 様が systemd にぶちきれぬ:  
<http://cpplover.blogspot.jp/2014/04/linussystemd.html>
- 「魚は頭から腐る」--systemdの開発者がトールバルズ氏を公然と批判:  
<http://japan.zdnet.com/article/35054778/>