

Systemd

[Linux]

- <https://fedoraproject.org/wiki/Systemd>
- <http://d.hatena.ne.jp/enakai00/20130914/1379146157>
- <http://alpha.mixi.co.jp/2013/12063/>

Linux の起動処理は、これまで `init/upstart` と呼ばれる仕組みで行われていました。Red Hat Enterprise Linux 7 (RHEL7) では、これが、`systemd` と呼ばれるまったく新しい仕組みに置き換わります。

Unit

- `systemd` では、「Unit」という単位で処理を管理
- `rc.sysinit` やサービス起動スクリプトが実施していた処理の内容は、すべて、Unit として定義
- Unit は、「target」「mount」「service」「device」など、役割によってタイプがわかれている
 - ・設定ファイル名の末尾「.target」「.service」で判別ができます
- Unit の定義ファイルは `/usr/lib/systemd/system/` ディレクトリ配下にあります。

Unit のタイプ

| タイプ | 説明 |
|-----------|---|
| mount | 指定のファイルシステムをマウントする |
| automount | オートマウント処理を実施する (<code>automountd</code> の代替的な機能) |
| service | 指定のバイナリを実行する (主にはデーモンの起動に使用する) |
| socket | <code>systemd</code> が Socket を Listen して、接続があるとプロセスに受け渡す (<code>xinetd</code> の代替的な機能) |
| path | 指定のファイルが作成されると、指定されたサービスを起動する |
| device | <code>udev</code> から通知されたデバイスを表す |
| target | 複数の Unit をまとめるために使用する |

`systemctl` コマンドによる Unit の確認

- 「`systemctl list-units`」コマンドを利用すると、現在の設定で稼働している Unit を確認することができます

稼働サービスの一覧

```
# systemctl list-units --type=service
```

on/off の設定

- `chkconfig` による on/off に対応

on

```
# systemctl enable docker.service
```

off

```
# systemctl disable docker.service
```

ターゲットの設定

- /usr/lib/systemd/system/ 配下の設定ファイルを確認

runlevel 5 に相当する「graphical.target」に対しては、「multi-user.target」が前提として定義されていますので、multi-user.target で有効化された service は、自動的に graphical.target でも有効化されます

例

- runlevel 3 と 5 で有効化したい WantedBy に multi-user.target を指定
- runlevel 5 だけで有効化したい WantedBy に graphical.target を指定

```
# cat /usr/lib/systemd/system/docker.service
[Unit]
Description=Docker Application Container Engine
Documentation=http://docs.docker.io
After=network.target
Requires=docker.socket

[Service]
Type=notify
EnvironmentFile=-/etc/sysconfig/docker
ExecStart=/usr/bin/docker -d --selinux-enabled -H fd://
Restart=on-failure
LimitNOFILE=1048576
LimitNPROC=1048576

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

サービスの確認

```
# systemctl status docker.service
docker.service - Docker Application Container Engine
Loaded: loaded (/usr/lib/systemd/system/docker.service; enabled)
Active: active (running) since 日 2014-07-13 23:04:55 JST; 30min ago
Docs: http://docs.docker.io
Main PID: 3072 (docker)
CGroup: /system.slice/docker.service
        3072 /usr/bin/docker -d --selinux-enabled -H fd://

 7月 13 23:04:53 higeoyaji docker[3072]: [ba0e5300.init_networkdriver()] creating new bridge for
docker0
 7月 13 23:04:54 higeoyaji docker[3072]: [ba0e5300.init_networkdriver()] getting iface addr
 7月 13 23:04:54 higeoyaji docker[3072]: [ba0e5300] -job init_networkdriver() = OK (0)
 7月 13 23:04:55 higeoyaji docker[3072]: Loading containers: : done.
 7月 13 23:04:55 higeoyaji docker[3072]: [ba0e5300.initserver()] Creating pidfile
 7月 13 23:04:55 higeoyaji docker[3072]: [ba0e5300.initserver()] Setting up signal traps
 7月 13 23:04:55 higeoyaji docker[3072]: [ba0e5300] -job initserver() = OK (0)
 7月 13 23:04:55 higeoyaji docker[3072]: [ba0e5300] +job acceptconnections()
 7月 13 23:04:55 higeoyaji docker[3072]: [ba0e5300] -job acceptconnections() = OK (0)
 7月 13 23:04:55 higeoyaji systemd[1]: Started Docker Application Container Engine.
```

再起動

```
# systemctl restart docker.service
```

設定の再読み込み

```
# systemctl reload [サービス]
```

該当の service が reload オプションに対応している必要あり

例

- ・ start/reload の際に実行されるコマンドが指定されています (「ExecStart=」および「ExecReload=」オプション)

```
[Service]
EnvironmentFile=/etc/sysconfig/ssh
ExecStartPre=/usr/sbin/ssh-keygen
ExecStart=/usr/sbin/ssh -D $OPTIONS
ExecReload=/bin/kill -HUP $MAINPID
```

chkconfig/service コマンドとの共存

/etc/init.d/ 以下に旧来のサービススクリプトがある場合、そちらを `chkconfig/service` コマンドで操作することも可能です。CentOS7 では、次のサービスは systemd に移行せずに残っています。

```
# chkconfig --list
```

注記：この出力は SysV サービスのみであり、ネイティブな systemd のサービスは含まれていません。systemd services. SysV 設定のデータはネイティブな systemd の設定によって上書きされます。systemd サービスを一覧表示するには 'systemctl list-unit-files' を使用してください。特定のターゲットにおいて有効化されているサービスを確認するには、'systemctl list-dependencies [target]'。

```
iprdump      0:off  1:off  2:on   3:on   4:on   5:on   6:off
iprinit      0:off  1:off  2:on   3:on   4:on   5:on   6:off
iprupdate    0:off  1:off  2:on   3:on   4:on   5:on   6:off
netconsole   0:off  1:off  2:off  3:off  4:off  5:off  6:off
network      0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

/etc/init.d/ 以下にスクリプトがないサービスに対して、`chkconfig/service` コマンドを実行した場合は、自動的に対応する `systemctl` コマンドが発行されます。

各フェーズにおける起動時間

```
# systemd-analyze
Startup finished in 390ms (kernel) + 7.498s (initrd) + 24.687s (userspace) = 32.576s
```

仮想環境種類の判別

```
# systemd-detect-virt
microsoft
```