

1章 システムアーキテクチャー

1-1. ハードウェア設定の決定と構成

1-1-1. デバイスファイルと/proc/ディレクトリ

- デバイスファイル (/dev/)
 - ハードウェア情報(/dev/hdaなど)
- /proc/ディレクトリ
 - カーネル情報(/proc/cpuinfoなど)

```
/proc/cpuinfo #CPUに関する情報
/proc/meminfo #メモリに関する情報
/proc/cmdline #起動時のカーネルオプション
```

- udev
 - udevはハードウェアの状態に応じて、動的にデバイスファイルを生成する。/etc/udev/rules.d/ディレクトリにルールファイルを配置して、自動生成される際のルールを構成することができる。

1-1-2. カーネルモジュールの操作

```
lsmod #現在読み込まれているカーネルモジュール一覧を表示する
insmod ファイル #カーネルモジュールを読み込み、有効にする
rmmod ファイル #カーネルモジュールを削除し、無効にする
modprobe [ option ] モジュール名 #カーネルモジュールを読み込む、もしくは削除(-r)する。依存関係にあるモジュールも操作対象とする
modinfo モジュール名 #指定したカーネルモジュールの情報を表示する
```

1-1-3. PCIデバイスとUSBデバイスの操作

- PCIデバイス
 - マザーボードに差し込んだもの

```
lspci [ option ] #認識しているPCIデバイスの一覧を表示する
# -vで詳細な情報
lsusb [ option ] #認識しているUSBデバイス一覧を表示する
# -vで詳細な情報
```

1-2. システムのブート

1-2-1. Linuxの起動フロー

電源投入 -> ファームウェア(BIOS) -> ブートローダ -> カーネル -> init/systemd -> ログインプロンプト

- 電源投入
- ファームウェア(BIOS)
 - キーボードなど必要最低限のデバイスを認識する
 - 機器の初期設定をする
 - 以前はBIOSというファームウェアが標準だった
 - UEFIが後継として使われる(GUI操作・GPT形式(part)のディスクからの起動サポート)
- ブートローダ
 - OS(カーネル)の選択
 - 主要なブートローダはGRUB
 - 起動時にメッセージを出力する(リンクバッファに出力)

```
dmesg [ option ] #リンクバッファの内容を表示する
```

1-3. ランレベル/ブートターゲットの変更とシステムのシャットダウン

1-3-1. initとsystemd

- プログラムを自動起動させたりする仕組み
- サービスと呼ぶ
- initはUNIX系で古くから利用されている仕組み、systemdは最近のディストリビューション

1-3-2. init

- ランレベル
 - システム起動状態を表す値
 - 0~6の数値で管理されている

```
cat /etc/inittab #デフォルトのランレベルが記載されている
init ランレベル #ランレベルを切り替える
telinit [ option ] ランレベル #GUI環境で起動する (-q : initの設定ファイルを再読み込み)
runlevel #現在起動中のランレベルと1つ前のランレベルを表示する
```

1-3-3. サービスの制御スクリプト

- /etc/init.d/ディレクトリには、サービスを制御するスクリプトが格納
 - Sで始まる: そのランレベルで起動させるスクリプト
 - Kで始まる: そのランレベルで起動させないスクリプト

```
/etc/init.d/ スクリプト名 引数
```

- start
- stop
- restart
- status

```
service サービス名 引数 #RH系ディストリビューションでSysV init環境におけるサービスを制御する
```

- サービスを自動起動させるには、シンボリックリンクのファイル名を変更する
- CentOSでは「chkconfig」コマンドが用意されていて、自動起動の設定を行うことができる

```
[root@localhost rc5.d]# pwd
/etc/rc5.d
[root@localhost rc5.d]# ls -l
lrwxrwxrwx. 1 root root 20 11月 4 2022 K50netconsole -> ../init.d/netconsole
lrwxrwxrwx. 1 root root 17 11月 4 2022 S10network -> ../init.d/network
```

```
chkconfig [ option ] サービス名 on|off
# --listでサービスの自動起動の状態を一覧表示
```

1-3-4. systemd

- 並列処理による高速な起動と終了処理
 - initは順次起動
- cgroupsによるプロセス管理
- 設定ファイルによるシステムの管理
 - /etc/init.d/のシェルスクリプトでの管理からの変更
- ユニットによる様々な処理の管理

```
systemctl サブコマンド [ ユニット名 ]
```

- start
- stop
- restart
- status
- **enable** #自動起動の有効化
- **disable** #自動起動の無効化
- is-enabled #自動起動の状態確認
- get-default #既定のターゲットを表示
- **set**-default #既定のターゲットを変更
- isolate #現在のターゲットを変更 (systemctl isolate rescue.target)
- list-unit-files #ユニットの状態を表示 ◎

- systemdでは様々な要素をユニットとして管理している
- 管理対象となっているユニットは「/usr/lib/systemd/system/」ディレクトリに配置
 - .service
 - .target (複数のユニットがとりまとめられている、システム起動時に利用)

1-3-5. システムのシャットダウンと再起動

```
shutdown option タイミング ["メッセージ"]
-h : シャットダウン
-r : 再起動
-k : テスト用に処理(実際はシャットダウンしない)
-c : シャットダウン・再起動の予約をキャンセル
-f : 次回起動時にfsck(整合性確認)を行わない
-F : 次回起動時にfsck(整合性確認)を行う

- now : 今すぐ実行
- 18:00 : 18時に実行
- +5 : 5分後に実行
```

- 互換性コマンド

```
#シャットダウン
init 0
halt
poweroff

#再起動
init 6
reboot
systemctl reboot
```

```
wall メッセージ #他ユーザへの警告メッセージなどを送ることができる
```