日曜研究室

技術的な観点から日常を綴ります

[xv6 #1] Chapter 0 – Operating system interfaces

テキストの7,8ページ

概要

xv6はユーザー空間とカーネル空間に分かれている。
ユーザー空間のプロセスからOSのインターフェースを通してカーネルが提供するサービスを呼び出すことをシステムコールという。
そしてこれはプロセスがユーザー空間とカーネル空間で交互に実行される事を示す。
カーネルはCPUにハードウェア実装された保護機構を使って、ユーザー空間のプロセスが自分自身のメモリ領域へのみアクセス可能なようにする。

xv6のシステムコール一覧

```
01 fork()  プロセスの生成
02 exit()  現在のプロセスを終了
03 wait()  子プロセスを待つ
04 kill(pid)   pidのプロセスを終了
05 getpid()    現在のプロセスのpidを返す
06 sleep(n)    単位時間 x nだけ寝る
07 exec(filename, *argv)   ファイルを読み込んで実行する
08 sbrk(n) プロセスのメモリをnバイト増やす
09 open(filename, flags)   ファイルを開く（flagsは読み書きフラグ）
10 read(fd, buf, n)    開いてるファイルからnバイト読み込んでbufに入れる
11 write(fd, buf, n)   開いてるファイルにbufの内容をnバイト書き込み
12 close(fd)   開いてたファイルを解放する
13 dup(fd) fdを複製する
14 pipe(p) パイプを生成してpのfdを返す
15 chdir(dirname)  現在のディレクトリを変更する
16 mkdir(dirname)  新しいディレクトリを作成する
17 mknod(name, major, minor)   デバイスファイルを作成する
18 fstat(fd)   開いてるファイルの情報を返す
19 link(f1, f2)    ファイルf1の別名f2を作成
20 unlink(filename)    ファイルを削除する
```

シェルはユーザー空間で実行される通常のプログラムであってカーネルの一部ではない。
内部でシステムコールを使ってるだけである。

だからシェルを入れ替えるのは簡単である。
世の中のUnixベースのシステムは多くのシェルの選択肢を持つ。
xv6のシェルはUnix Bourne shellを元にしたシンプルなものである。

感想

Chapter 0のさわりって事もあって、OSのコード書いたり読んだりしたことなくても何となく知ってる内容でした。
読み違えてなければxv6のシステムコールは上記で全てなので、僕でも中身を理解できそうかもと思えてきました。
一日2ページぐらいのペースでテキストを読み進められればいいなぁ(*´∀｀*)ﾎﾟﾜ

カテゴリー: 技術 | タグ: xv6 | 投稿日: 2012/2/6 月曜日 [http://peta.okechan.net/blog/archives/1215] |