

Sistemi Operativi (Laboratorio)

Lorenzo Martignoni

Dipartimento di Informatica e Comunicazione
Università degli Studi di Milano, Italia
lorenzo@security.dico.unimi.it

a.a. 2008/09

Lezione VI: Permessi, Make e primi esperimenti con il kernel

Permessi

Ad ogni file vengono associati dei **permessi**, che definiscono le azioni permesse sui dati del file

- ▶ **Read:** leggere il contenuto del file o directory
- ▶ **Write:** scrivere (cambiare) il file o directory
- ▶ **eXecute** eseguire le istruzioni contenute nel file o accedere alla directory

R	W	X	
1	1	0	6
1	0	1	5
1	0	0	4
1	1	1	7

I permessi possono essere diversi per 3 categorie di utenti del sistema:

- ▶ **User:** il “proprietario” del file
- ▶ **Group:** gli appartenenti al gruppo proprietario
- ▶ **All:** tutti gli altri

Utenti e gruppi

Utenti definiti nel file `/etc/passwd`:

```
root:##root:0:0:Big Brother:/root:
daemon:*:1:1:The Deuce:/etc:
bin:##root:2:0:Binaries:/home/bin:
uucp:*:5:5:UNIX to UNIX copy:/usr/spool/uucp:...
...
nobody:*:9999:99:./tmp:
```

Gruppi definiti nel file `/etc/group`:

```
operator:*:0:
daemon:*:1:
bin:*:2:
other:*:3:
tty:*:4:
...
nogroup:*:99:
```

Agire sui permessi

- ▶ Cambiare il proprietario
 - ▶ `chown utente[:gruppo] file`
- ▶ Cambiare il gruppo
 - ▶ `chgrp gruppo file`
- ▶ Cambiare i permessi
 - ▶ `chmod 755 file`
 - ▶ `chmod +x file`
 - ▶ `chmod a=rw file`
 - ▶ `chmod g-x file`
- ▶ (per creare un utente: `adduser`)

Il proprietario di un processo in esecuzione è normalmente **diverso** dal proprietario del file contenente un programma (e diverso ad ogni esecuzione)

- ▶ effective UID bit: il processo assume come proprietario il proprietario del file del programma
- ▶ SUID root
- ▶ `chmod 4555 file`
- ▶ `chmod u+s file`

Stuart Feldman, 1977 at Bell Labs.

Permette di specificare **dipendenze** fra processi di generazione.

Dipendenze: se cambia questo file, allora il processo di generazione deve essere ripetuto.

```
helloworld: helloworld.o
    cc -o $@ $<
```

```
helloworld.o: helloworld.c
    cc -c -o $@ $<
```

```
.PHONY: clean
```

```
clean:
    rm helloworld.o helloworld
```

Un esempio di Makefile:
`/usr/src/commands/elvis/Makefile`

Iniziamo a dare un'occhiata al codice

Il codice del sistema si trova in `/usr/src`

Per ricompilare il sistema:

```
cd /usr/src/tools  
make clean  
make image  
make hdboot
```

Ci mette 3-5 minuti.

Esercizio: Cambiare Il nome del sistema che appare nella prima riga col copyright

Premendo i tasti F1-F7, F10-F12, Shift+F1-F4 vengono stampate informazioni sulle strutture dati del kernel

- ▶ Identificare i valori del parametro di boot `memory`

Pag. 356 del libro di testo.

Il boot monitor e il setup iniziale

- ▶ Accedere al boot monitor
- ▶ Provare il comando `help`
- ▶ Cosa succede cambiando il parametro di `boot memory=800...` in `memory=900...`?
- ▶ Cambiare qualche campo di `kinfo` (p.es. `release`) modificandone l'inizializzazione in `cstart`

© 2009 Mattia Monga & Lorenzo Martignoni

Creative Commons Attribuzione-Condividi allo stesso modo 2.5
Italia License.

<http://creativecommons.org/licenses/by-sa/2.5/it/>.