

Laboratorio di Sistemi Operativi
a.a. 2003-2004

Esercitazione 1

Renzo Davoli
Alberto Montresor

Esercitazioni

- ♦ **Danno diritto a 2+2 punti bonus (in totale) che verranno "aggiunti" alla valutazione del progetto**
- ♦ **Non hanno valutazione "completa" e "immediata"**
 - ♦ li guarderemo più avanti (quando avremo tempo) (ovviamente prima che il corso finisca!)
 - ♦ non hanno valutazione approfondita
 - ♦ 2 punti se funziona ed è ben scritto
 - ♦ 1 punto se funziona ma è un casino
 - ♦ 0 punti se non funziona
- ♦ **E' un meccanismo per incentivarvi a studiare durante l'anno**
- ♦ **Le esercitazioni vanno svolte in gruppo!**

Esercitazione 1

- ♦ **Scopo:**
 - ♦ iniziare a prendere confidenza con UML e scripting
- ♦ **Passi**
 - ♦ creazione di uno *script*
 - ♦ verifica se sono state eseguite variazioni su un file system, basato su file di configurazione
 - ♦ creazione di un *file di configurazione* di esempio
 - ♦ creazione di uno *script di test*
 - ♦ deve illustrare il funzionamento dello script, modificando uno o più file e lanciando lo script apposito
 - ♦ inserimento del materiale prodotto in un file system UML
 - ♦ creazione di un file system COW basato su UML

Lo script principale

- ◆ **Alla prima esecuzione**
 - ◆ Crea un "database" contenente informazioni sul file system analizzato
 - ◆ Per ogni file:
 - ◆ diritti di accesso, dimensione
 - ◆ checksum del contenuto
 - ◆ Per ogni directory
 - ◆ diritti di accesso, dimensione
 - ◆ file e directory contenute
- ◆ **Formato di questo database:**
 - ◆ a vostra scelta
 - ◆ deve semplificarvi la vita nelle scelte successive

Opzioni da linea di comando

- ◆ **Flag**

- ◆ **-c [dir], --create [dir]**

Crea un nuovo database nella posizione specificata (se presente), o nella directory corrente (altrimenti). Sovrascrive un database pre-esistente

- ◆ **-t [dir], --test [dir]**

riporta eventuali variazioni rispetto al database nella posizione specificata (se presente), senza aggiornare il database

- ◆ **-u [dir], --update [dir]**

- ◆ come test, riporta le variazioni

- ◆ se non ci sono state variazioni marcate error, aggiorna il database

- ◆ **-f configfile**

utilizza il file di configurazione specificato

File di configurazione

- ♦ **Può assumere una forma di questo tipo:**

```
# Commento da ignorare
```

```
/bin/                error
```

```
/etc/                error
```

```
/etc/passwd         warning
```

```
/var/www            warning
```

- ♦ **Significato:**

- ♦ Se vi sono modifiche in `/bin/` o `/etc/`, con eccezione di `/etc/passwd`, riporta un errore e blocca l'eventuale aggiornamento
- ♦ Se vi sono modifiche in `/etc/passwd` o in `/var/www`, riporta un warning

Ulteriori dettagli

- ◆ **Nel caso di --update**
 - ◆ L'aggiornamento deve essere atomico
 - ◆ Ovvero, non aggiornate il database un elemento dopo l'altro, per poi scoprire all'ultimo momento che l'ultimo elemento genera un error (per cui il database non andava aggiornato)
- ◆ **Output dello script**
 - ◆ Bisogna riportare:
 - ◆ ogni file creato e rimosso
 - ◆ ogni file modificato o per i quali sono stati modificati i diritti
 - ◆ lo stato prima e dopo la modifica
 - ◆ la severità del problema (warning, error)

Lo script di testing

- ◆ **Uno script che**

- ◆ lancia lo script principale in modalità create
- ◆ modifica/rimuovi/inserisci alcuni file
 - ◆ almeno un file con severità warning in una directory con severità error
 - ◆ almeno un file con severità warning
 - ◆ almeno un file in una directory non controllata
- ◆ lancia lo script principale in modalità test
- ◆ modifica alcuni file
 - ◆ almeno un file con severità error
- ◆ lancia lo script principale in modalità update
- ◆ riporta il file system nello stato originale

File system UML

- ◆ **Vi forniamo un file system da montare tramite UML**
 - ◆ presso `/home/students/COURSES/labso/UML`
 - ◆ Due utenti:
 - ◆ root con password `piripicchio`
 - ◆ reno con password `piripicchio`
- ◆ **Per creare un file system in modalità cow:**
 - ◆ `linux ubd0=iso04mz50.cow, filesystem-del-corso`
 - ◆ dove `filesystem-del-corso` è il file system fornito da noi
 - ◆ dove `iso04mz50` è il numero del vostro gruppo
 - ◆ verrà creato un file system in modalità cow
- ◆ **Per montare un file system in modalità cow precedentemente creato:**
 - ◆ `linux ubd0=iso04mz50.cow`

File system UML

- ♦ **Il file .cow dovrà riflettere le seguenti variazioni**
 - ♦ Lo script principale e lo script di testing devono essere inseriti all'interno della directory /root
- ♦ **Cosa faremo per eseguire il test?**
 - ♦ Facciamo partire user-mode linux, utilizzando il file system che ci avete mandato montato in modalità COW
 - ♦ Facciamo login come root
 - ♦ Eseguiamo lo script di test, come indicato dal file README (vedi dopo)
 - ♦ Verifichiamo i risultati

Commenti

- ♦ **In italiano o in inglese (italiano corretto o inglese leggibile)**
- ♦ **Dovete inserire la quantità minima di commenti che renda chiaro il funzionamento del programma.**
 - ♦ tanti commenti sono un costo per un'azienda:
+commenti = +tempo per leggerli
 - ♦ pochi commenti sono un costo per un'azienda:
-commenti = +tempo per comprendere il programma
- ♦ **I commenti non sono valutati in base alla quantità bensì in base alla qualità (chiarezza)**

Commenti

- ◆ **Commenti interni**

- ◆ 1 commento all'inizio del file che indichi cosa fa lo script e qual è il suo "usage"
- ◆ 1 commento prima di ogni funzione per descrivere le sue funzionalità e i suoi parametri
- ◆ 1 commento per ogni punto criptico

- ◆ **File README**

- ◆ deve descrivere sia le modalità di utilizzazione (dal punto di vista di un utente) che le modalità di implementazione (dal punto di vista dello sviluppatore)
- ◆ deve indicare lo script che deve essere lanciato
- ◆ in formato .txt, .html, .pdf

Cosa, come, quando

◆ Consegnate

- ◆ un file con estensione .tgz (tar gzipped) e nome uguale all'identificativo del gruppo
- ◆ contenente solo una directory con nome uguale all'identificativo del gruppo
- ◆ la directory deve contenere
 - ◆ tutti gli script che avete scritto
 - ◆ file AUTHORS (lista autori) (non scrivete il nome su ogni file)
 - ◆ file README
 - ◆ il file .cow con il nome uguale all'identificativo del gruppo

```
% ls
lso04mz50.tgz
% tar zxvf lso04mz50.tgz
% ls
lso04mz50 lso04mz50.tgz
% cd lso04mz50
% ls
README AUTHORS checkfile
testscript lso04mz50.cow
```

Cosa, come, quando

- ◆ **Cosa non consegnare:**

- ◆ dischetti o materiale cartaceo
- ◆ file di prova, materiale non correlato all'esercizio

- ◆ **Come consegnare:**

- ◆ copiando il file .tgz in una directory del file system trusted che verrà indicata in seguito nel newsgroup

- ◆ **Quando consegnare**

- ◆ entro 7 aprile 2004, ore 23.59