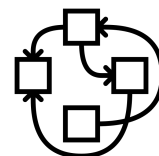


GNU/Hurd - parte II

Entra nel vivo l'esplorazione di GNU/Hurd...



GNU/Hurd Live CD

Provare Hurd senza installazione e' possibile cosi' come fareste con GNU/Linux. Un live cd e' via...

Scarichiamo allora il live cd dal link [0] e avviamolo! Per nostra sfortuna l'ultimo live cd di GNU/Hurd risale al 2007... Di questi live cd ne esistono due versioni, una *tiny* di 5 Mb (compressato) e una di 74 Mb (compressato).

Volendo avviare GNU/Hurd Live con un emulatore possiamo affidarci a **qemu** e lanciare:

```
$ qemu -boot d -cdrom hurd-tiny-cd-20071206.iso
```

Emulare GNU/Hurd

Volendo provare GNU/Hurd senza ricorrere ad un live cd ed evitando (per ora) l'installazione, possiamo approfittare delle immagini ufficiali [1] o meno [2] disponibili in rete.

Anche in questo caso utilizzeremo **qemu** per avviarle. Per avviarle:

```
$ qemu -boot c debian-hurd-k16-qemu.img
```

Installare Debian GNU/Hurd

Ok... Ci siamo stancati di avere la pappa pronta ed evitare l'installazione, vero? Installiamo Debian GNU/Hurd! L'installazione non e' semplice ed intuitiva, ma sicuramente chi mastica un po' di riga di comando e chi ha come motto "Rischio e' il mio secondo nome" allora si divertira' a farlo!

L'installazione di Debian GNU/Hurd **deve** essere effettuata da un altro sistema gia' funzionante (che puo' essere GNU/Linux, BSD ecc..). Questo e' cio' che si dice **cross-installazione**. Visto che siete arrivati fino a qui senza cambiare pagina allora siete dei temerari! Scegliamo quindi di abbandonare **qemu** e installare nativamente...

Supponiamo innanzitutto di voler installare GNU/Hurd sulla partizione che GNU/Linux chiamerebbe `/dev/hda2` e anche di voler installare appunto da quest'ultimo sistema. Creiamo dunque un filesystem **ext2** impostando il proprietario a **hurd**:

```
# mke2fs -o hurd /dev/hda2
```

Impostare il proprietario a **hurd** fa si che si possano sfruttare alcune caratteristiche avanzate, come ad esempio un quarto gruppo di bit per gestire i permessi di utenti sconosciuti.

Ok. Abbiamo creato il filesystem. Montiamolo e decomprimiamoci dentro il tarball del sistema prelevato da [3] preservando il proprietario dei file:

```
# mount /dev/hda2 /mnt/hurd -t ext2
```

```
# cd /mnt/hurd
```

```
# tar -same-owner -xzvpf /home/val/gnu-latest.tar.gz
```

Fatto. Possiamo avviare GNU/Hurd. L'installazione pero' non e' ancora completa... Per

avviare GNU/Hurd dobbiamo accedere alla console di **grub** (il boot loader che il 99 % di voi utilizza...). Prima di procedere oltre e' bene fare attenzione alle regole di nomenclatura utilizzate da linux, hurd e grub relativamente alle partizioni, vi invito dunque a consultare il box 0. Riavviate e non appena appare grub premete **c** per accedere alla console. Digitate ora:

```
grub> root (hd0,1)
grub> kernel /boot/gnumach.gz
-s root=device:hd0s2
grub> module /hurd/ext2fs.static --
multiboot-command-line=${kernel-command-line} --
host-priv-port
grub> module /lib/libd.so.1
/hurd/exec $(exec-task=task-create)
```

Fiuuu... Il primo avvio e' andato... Che faticaccia! Come a scuola copiare dalla lavagna senza capire non ha senso, analizziamo cio' che abbiamo fatto.

Nella prima riga abbiamo chiesto a grub di impostare come partizione di root (hd0,1), seguendo le convenzioni utilizzate dal boot loader. Abbiamo poi caricato GNU Mach istruendolo sul suo root filesystem (ovviamente nel formato a lui gradito). Ora viene il bello... Con le ultime due righe carichiamo il **root filesystem server** ed l'**exec server**.

Nomenclatura di dischi e partizioni

Su grub i dischi ide e scsi sono chiamati (hdN) dove N e' il numero del dispositivo e parte da 0. Su linux si indicano con `/dev/hdX` o `/dev/sdX` dove X indica una lettera che contraddistingue la posizione sul controller. Su GNU Mach si indicano quasi come su linux con `/dev/hdN` o `/dev/sdN` dove N e' un numero che ha lo stesso ruolo di X per linux e che parte da 1.

Vediamo ora come vengono indicate le partizioni! Grub aggiunge, separato da una virgola dal nome del device, un numero che parte da 0. Linux aggiunge al nome del dispositivo un numero che parte da 1, GNU Mach, ispirandosi a BSD, aggiunge una S seguita da un numero che parte da 1.

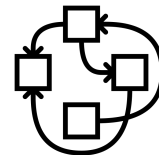
Piccolo esempio:

la prima partizione di un disco IDE situato sul primary master si chiamera' (hd0,0) su grub, `/dev/hda1` su linux, `/dev/hd0S1` su GNU Mach.

Box 0

GNU/Hurd - parte II

Entra nel vivo l'esplorazione di GNU/Hurd...



Finalmente possiamo digitare l'attesissimo comando **boot** e buttarci sul tasto invio! Il primo avvio deve essere eseguito in single user e cio' l'abbiamo fatto aggiungendo -s come parametro di GNU Mach. Senza digitare ogni volta quel putiferio di righe e' bene configurare una volta per tutte grub (io ho utilizzato una versione "tradizionalista" di grub, essendo affezionato al buon vecchio *menu.lst* :-P).

Welcome in Hurd!

Benvenuti a bordo! E' giunta l'ora di completare l'installazione (sempre che sia filato tutto liscio...). Prima di lanciare lo script che terminera' l'installazione e' bene impostare la variabile **TERM**:

```
# export TERM=mach
```

Ora avviamo native-install e completiamo il tutto:

```
# ./native-install
```

Avrete notato che siete root (basta guardare il # nel prompt) e che non avete effettuato alcun login... Tranquilli... Ben presto potrete *settare* anche gli utenti e loro credenziali.

Prima pero' pensiamo a ultimare gli ultimi aspetti basilari del sistema, ad esempio modifichiamo il file */etc/fstab* per aggiungere eventuali altri filesystem... Ricordate di seguire le convenzioni usate da GNU Mach quando inserite il nome di un device all'interno del file... Ogni dispositivo va inoltre creato (in /dev) utilizzando **MAKEDEV**, ad esempio:

```
# cd /dev
```

```
# MAKEDEV hd1s2
```

Ok, possiamo riavviare. Mi raccomando, non e' piu' necessario avviare in single user.

Welcome in Hurd. Again.

Finalmente siamo davanti al prompt di un sistema GNU/Hurd

completamente installato. Abbiamo davanti un prompt di login, digitate login root e proseguite. A questo punto potete digitare i classici passwd o adduser rispettivamente per modificare la password di un utente o aggiungere un nuovo smanettone.

I translator

Dopo tanta fatica possiamo goderci il nostro nuovo sistema operativo e giocare con uno degli aspetti piu' interessanti: i translator! Altro non sono che dei server che realizzano in user space alcune delle funzioni che invece il kernel linux realizza in kernel space e che il kernel GNU Mach non fornisce (essendo un microkernel).

E' utilizzando i translator che possiamo montare i filesystem, che possiamo usare la rete, che possiamo autenticarci ecc... I translator si classificano fra attivi e passivi. Supponiamo di utilizzare un translator attivo per montare una partizione e uno passivo per montarne un'altra. La principale differenza e' che appena attivati i traduttori fra i processi attivi troviamo soltanto quello attivo. Quello passivo viene avviato solo nel momento in cui cerchiamo di accedere alle risorse che mette a disposizione, ad esempio entrando nel mount point della relativa partizione.

I due comandi principali per la gestione dei translator sono settrans e showtrans. Il primo permette di impostare un translator, ma anche di modificarne uno gia' in esecuzione.

Vediamo un piccolo esempio su come configurare la nostra scheda di rete. Le assegneremo l'indirizzo 192.168.1.2, netmask 255.255.255.0 e gateway 192.168.1.1:

```
$ settrans -fgap /servers/socket/2
```

```
/hurd/pfinet -i eth0 -a 192.168.1.2 -g 192.168.1.1 -m 255.255.255.0
```

Voila'! fgap e' in realta' la combinazione di piu' parametri: fg per forzare la chiusura di ogni istanza del translator esistente, ap per rendere il translator sia attivo che passivo. /servers/socket/2 e' il nodo al quale agganciare il translator, una sorta di mountpoint potremmo dire. /hurd/pfinet e' il translator che vogliamo lanciare, quello che gestisce la rete. E' interessante notare che tutti i parametri che seguono /hurd/pfinet sono parametri del translator e non di settrans.

Vediamo un altro esempio, questa volta su come montare una partizione.

```
$ settrans -cgap /mnt /hurd/ext2fs /dev/hd0S4
```

Per smontare:

```
$ settrans -g /mnt/
```

Un traduttore puo' essere lanciato con il parametro -help che fornisce utili indicazioni.

Se avete compreso il funzionamento di un microkernel e dei translator, non vi sara' difficile immaginare che non appena avviato il sistema avremo gia' in esecuzione alcuni di questi server per ottenere le funzioni basilari. Basta un ps -aux per averne conferma. Vediamo ad esempio il server per l'autenticazione, quello per l'accesso al filesystem, ecc...

Di translator ne esistono decine e decine, alcuni utili, altri un po' meno (basti pensare al translator birthday...). Date uno sguardo in /hurd per vedere quali server avete a disposizione.

Nella prossima puntata vedremo come installare nuovi pacchetti utilizzando il famoso gestore apt-get...

Riferimenti:

[0] <http://www.superunprivileged.org/hurd/live-cd/>

[1] <http://ftp.debian-ports.org/debian-cd/K16/debian-hurd-k16-qemu.img.tar.gz>

[2] <http://draketo.de/dateien/hurd/bab-hurd-qemu-2008-10-29.img.tar.bz2>

[3] <http://ftp.debian-ports.org/debian/base/debian-gnu-hurd-current.tar.gz>