

UML installation howto by Teo (matteo.fazi@gmail.com)

Terminologia:

- **Host:** la macchina reale che ospiterà la macchina virtuale
- **Guest:** la macchina virtuale

NB: tutti i passi dell' installazione devono essere fatti da superutente.

Premessa:

prima di iniziare con l'installazione vera e propria occorre controllare che l'host kernel abbia abilitato il supporto per i loopback devices:

```
Device drivers -->
  Block devices -->
    <*> loopback device support
```

Creazione dei filesystems della macchina virtuale

creiamo un po' di directory che ci serviranno:

```
root@host#: mkdir /uml
root@host#: mkdir /uml/filesystems
root@host#: mkdir /uml/files
```

creiamo l'"immagine" del filesystem della macchina virtuale e della partizione di swap:

```
root@host#: cd /uml/filesystems/
root@host#: dd if=/dev/zero of=root_fs count=1 bs=1M seek=1500 #filesystem da 1.5
  GB (seek=1500) virtuali
root@host#: dd if=/dev/zero of=swap_fs count=1 bs=1M seek=500 #swap da 500MB
root@host#: mke2fs -F root_fs
root@host#: mkswap swap_fs
```

Installazione di Gentoo sul filesystem della macchina virtuale

```
root@host#: mkdir /mnt/loop
root@host#: mount root_fs /mnt/loop -o loop
root@host#: cd /uml/files
root@host#: (scaricati lo stage3 della 2004.3) #
  http://www.die.unipd.it/pub/Linux/distributions/gentoo-
  sources/releases/x86/2004.3/
root@host#: (scaricati lo snapshot del portage della 2004.3)
  stesso link di sopra
root@host#: tar -xvvpf LO_STAGE3_SCARICATO -C /mnt/loop # la 'C' è maiuscola
root@host#: tar -xvvpf LO_SNAPSHOT_SCARICATO -C /mnt/loop/usr
root@host#: cp /etc/resolv.conf /mnt/loop/etc
root@host#: chroot /mnt/loop #chroot nel filesystem della macchina virtuale =
  sei nel fs della macchina virtuale
root@/#: passwd
root@/#: emerge quello_che_vuoi (iptables, tcpdump, sysklogd, vixie-cron...)
root@/#: rc-update add IL_SERVIZIO_CHE_VUOI (sshd, sysklogd...) default
root@/#: rc-update del modules
root@/#: rc-update del keymaps
root@/#: rc-update del consolefont
root@/#: rc-update del clock
root@/#: rm /usr/portage/distfiles/*
root@/#: mkdir /mnt/host
```

```
root@/#: nano -w /etc/fstab
```

fstab deve essere cosi:

```
#/dev/BOOT          /boot          ext2           noauto,noatime    1 1
/dev/ubd/0          /              ext3           noatime           0 0
/dev/ubd/1          /              none          swap              sw                0 0
none               /mnt/host     hostfs        LA_TUA_HOME       0 0
#/dev/cdroms/cdrom0 /mnt/cdrom     iso9660       noauto,ro         0 0
#/dev/fd0           /mnt/floppy   auto          noauto            0 0

none               /proc         proc          defaults          0 0
none               /dev/shm     tmpfs        defaults          0 0
```

```
root@/#: nano -w /etc/inittab
```

inittab deve essere cosi: (le righe da modificare sono sottolineate)

```
# /etc/inittab: This file describes how the INIT process should set up
# the system in a certain run-level.
#
# $Header: /home/cvsroot/gentoo-src/rc-scripts/etc/inittab,v 1.6 2003/01/06 21:3
2:43 azarah Exp $

#
# Default runlevel.
id:3:initdefault:

# System initialization, mount local filesystems, etc.
si::sysinit:/sbin/rc sysinit

# Further system initialization, brings up the boot runlevel.
rc::bootwait:/sbin/rc boot

10:0:wait:/sbin/rc shutdown
11:S1:wait:/sbin/rc single
12:2:wait:/sbin/rc nonetwork
13:3:wait:/sbin/rc default
14:4:wait:/sbin/rc default
15:5:wait:/sbin/rc default
16:6:wait:/sbin/rc reboot
#z6:6:respawn:/sbin/sulogin

# TERMINALS
c1:12345:respawn:/sbin/agetty 38400 tty0 linux #attenzione: deve esserci tty0!!!
#c2:12345:respawn:/sbin/agetty 38400 tty2 linux
#c3:12345:respawn:/sbin/agetty 38400 tty3 linux
#c4:12345:respawn:/sbin/agetty 38400 tty4 linux
#c5:12345:respawn:/sbin/agetty 38400 tty5 linux
#c6:12345:respawn:/sbin/agetty 38400 tty6 linux

# What to do at the "Three Finger Salute".
ca:12345:ctrlaltdel:/sbin/shutdown -r now

# Used by /etc/init.d/xdm to control DM startup.
# Read the comments in /etc/init.d/xdm for more
# info. Do NOT remove, as this will start nothing
# extra at boot if /etc/init.d/xdm is not added
# to the "default" runlevel.
x:a:once:/etc/X11/startDM.sh

# End of /etc/inittab
```

```
root@/#: nano -w /etc/securetty
securetty deve essere così: (le righe da aggiungere sono sottolineate)
```

```
vc/0
vc/1
vc/2
vc/3
vc/4
vc/5
vc/6
vc/7
vc/8
vc/9
vc/10
vc/11
vc/12
tty0
tty1
tty2
tty3
tty4
tty5
tty6
tty7
tty8
tty9
tty10
tty11
tty12
```

```
root@/#: echo "NOME_DELLA_MACCHINA_VIRTUALE" > /etc/hostname
root@/#: exit
root@host: umount /mnt/loop
```

Compilazione del kernel della macchina virtuale

```
root@host#: emerge development-sources #ATTENZIONE: assicurarsi che il link
'linux' non punti a questi sorgenti!!!
Se non usi Gentoo (= non hai il comando emerge) scaricati i sorgenti
del kernel vanilla 2.6.xx.xx e scompattali in /usr/src
root@host#: cd /usr/src/linux-2.6.xx.xx
root@host#: make menuconfig ARCH=um
```

assicurarsi che nel kernel siano abilitate le seguenti funzioni:

```
UML Specific Options -->
  <*> Host filesystem
  [*] 2G/2G host address space split
UML Network Devices -->
  [*] tutto selezionato!!!
Filesystems -->
  Pseudo Filesystems -->
    [*] /dev filesystem support (OBSOLETE)
    [*] Virtual memory filesystem support
    [*] tmpfs Extended attributes
```

e il supporto per iptables!!!(se devi lavorarci) Networking support--> ...

```
root@host#: make linux ARCH=um
```

Creazione dei moduli...

```
root@host#: make modules ARCH=um
```

...e loro installazione

```
root@host#: mkdir mnt
root@host#: mount /uml/filesystems/root_fs mnt/ -o loop # mnt/ senza slash
davanti!!!!
root@host#: make modules_install INSTALL_MOD_PATH=`pwd`/mnt ARCH=um
root@host#: umount mnt/
```

Un ultimo sforzo....

```
root@host#: cd /usr/local/bin
root@host#: ln -s /usr/src/linux-2.6.xx.xx/linux linux
```

Speriamo Bene...

```
root@host#: cd /uml/filesystems
root@host#: linux ubd0=root_fs ubd1=swap_fs
```

dovrebbe partire la macchina virtuale.... e dovresti riuscire a loggarti...

Gli ultimi ritocchi...

Quando tutto funziona copia lo script "uml_start" :-) in /usr/local/bin:

```
root@host#: mv PATH_DI/uml_start /usr/local/bin
root@host#: chmod 0777 /usr/local/bin
```

per settare l'ip delle macchine guest:

```
root@guest: ifconfig eth0 xxx.xxx.xxx.xxx
```