

02 marzo 2012

Benvenuti!



MMUL S.a.S. di Raoul Scarazzini & C.
Via Cantore, 11 - 20017 - Rho (MI) - P.IVA 07188550961

<http://www.mmul.it>
info@mmul.it

Chi è MMUL



- MMUL nasce con il portale tecnico
<http://www.miamammauslinux.org>
nel gennaio 2008.
- Nell'ottobre 2010 prende vita la società con l'obiettivo di fornire il meglio ai clienti in materia Linux e software OpenSource.



Alcuni clienti



Perché Linux

Linux



Perché Linux?

Newsgroups: comp.os.minix
From: torva...@klaava.Helsinki.FI (Linus Benedict Torvalds)
Date: 25 Aug 91 20:57:08 GMT
Local: Sun, Aug 25 1991 4:57 pm
Subject: What would you like to see most in minix?
Reply to author | Forward | Print | View thread | Show original | Report this message | Find messages by this author

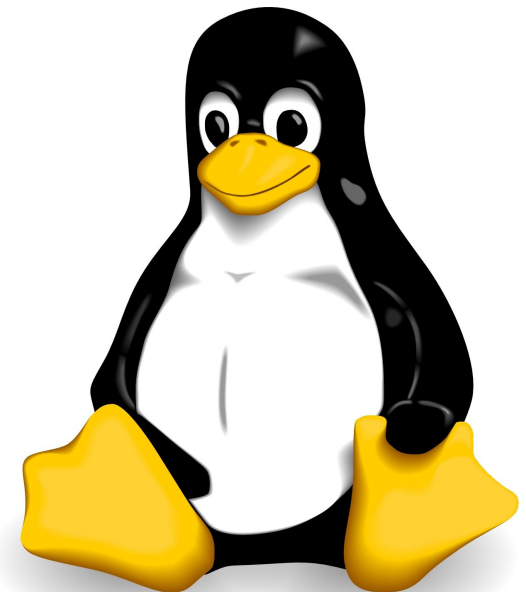
Hello everybody out there using minix -

I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones. This has been brewing since april, and is starting to get ready. I'd like any feedback on things people like/dislike in minix, as my OS resembles it somewhat (same physical layout of the file-system (due to practical reasons) among other things).

I've currently ported bash(1.08) and gcc(1.40), and things seem to work. This implies that I'll get something practical within a few months, and I'd like to know what features most people would want. Any suggestions are welcome, but I won't promise I'll implement them :-)

Linus (torvalds@kruuna.helsinki.fi)

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT protable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-).



Linux è Open Source

- Nessun costo di licenza
- Liberamente scaricabile
- Liberamente modificabile
- Basato su GPL v2 (General Public License)
 - Puoi modificare
 - Puoi rivendere
 - Ma sarà SEMPRE Open Source



Linux nelle sue parole chiave

- Kernel
 - Il cuore del sistema, l'interfaccia verso l'hardware
- GNU (GNU's Not Unix)
 - Il sistema operativo, insieme di software per operare
- Distribuzione
 - Raccolte di software che costituiscono un ambiente operativo
 - Red Hat, Debian, Ubuntu, SuSe



Di cosa si parla oggi

VIRTUALIZZAZIONE



Nascita della virtualizzazione

- Anni '60, laboratori di IBM: nascita del CP-40, precursore del sistema CP-67 e di tutta la "VM Family" di IBM, alla base dei sistemi operativi per i mainframe IBM s370 prima e s390 dopo.
- Tale sistema operativo era basato su virtual machine, virtual memory e time-sharing (multi tasking, più operazioni contemporanee).



Evoluzione della virtualizzazione

- Legge di Gordon Moore (co-fondatore di Intel):

« Le prestazioni dei processori, e il numero di transistor ad esso relativo, raddoppiano ogni 18 mesi. »

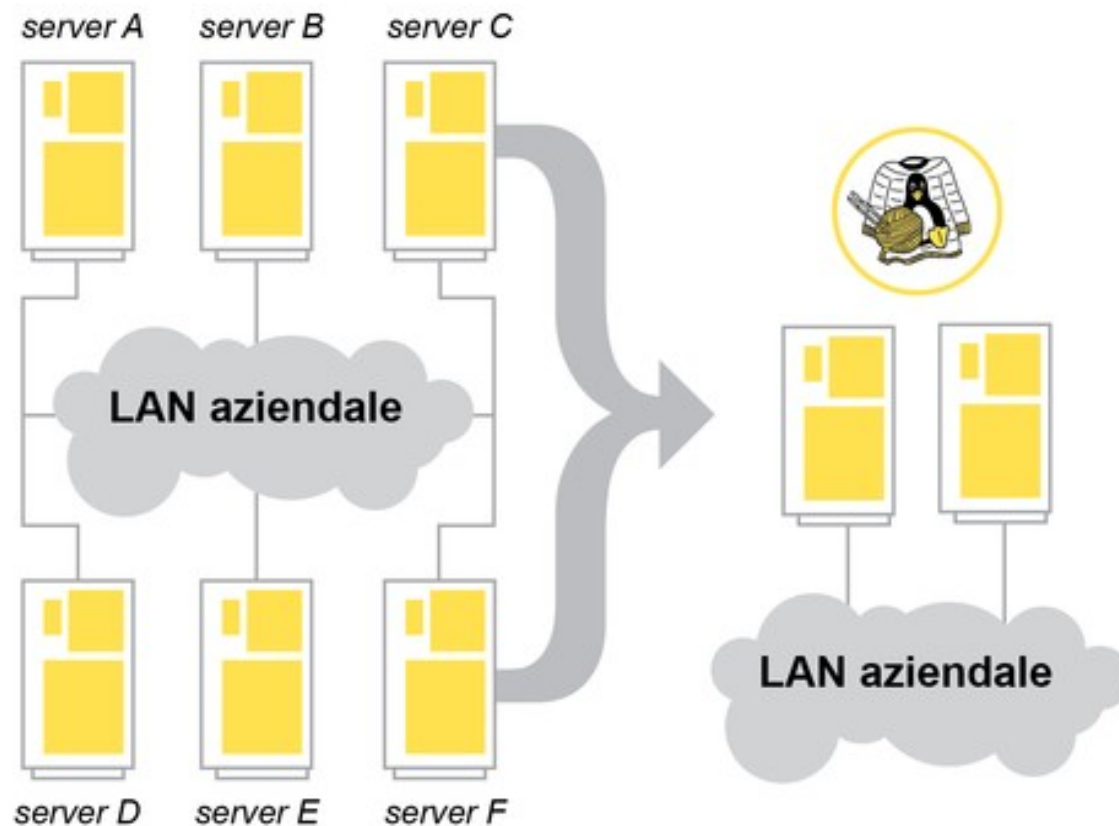
http://it.wikipedia.org/wiki/Legge_di_Moore

- Sistemi sempre più potenti
- Sistemi sempre meno costosi



Cosa significa virtualizzare

- Ottimizzare, consolidare e centralizzare:



Vantaggi e svantaggi

Vantaggi

- Ottimizzazione dell'hardware disponibile
- Gestione centralizzata
- Riduzione costi (in termini di hardware, non licenze)
- Consolidamento

Svantaggi

- Performance insoddisfacenti
- Sovrabbondanza di sistemi virtuali (macchine create inutilmente)
- Virtualizzazioni inutili (non tutto va virtualizzato)



Cosa offre oggi il mercato

- Closed source



- Open source



Questo seminario tratta di KVM

- KVM - Kernel-based Virtual Machine
 - Modulo integrato nel kernel linux (dalla release 2.6.20) che permette di sfruttare le estensioni di virtualizzazione dei processori moderni.
 - Non effettua alcuna emulazione, mette a disposizione l'hardware alle macchine virtuali
 - Necessita di un programma in userspace (libvirt o qemu) per interfacciare la fornitura hardware alle macchine virtuali



Vantaggi e svantaggi di KVM

Vantaggi

- Semplicità di implementazione e funzionamento
- Funzionamento ottimizzato (kernel level)
- Capacità di supportare ogni tipo di sistema operativo

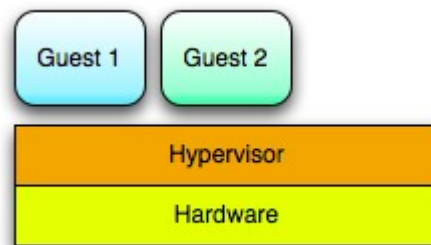
Svantaggi

- Dipendenza da hardware specializzato: non funziona su CPU che non supportino le estensioni per la virtualizzazione.
- Inadatto a computer "anziani" (fortunatamente, tutte le moderne CPU hanno questo supporto)



Come funziona KVM: l'Hypervisor

- **Hypervisor**: alla base di tutti i sistemi di virtualizzazione (KVM compreso)
 - Si inserisce tra i sistemi guest (le VM) e l'hardware:

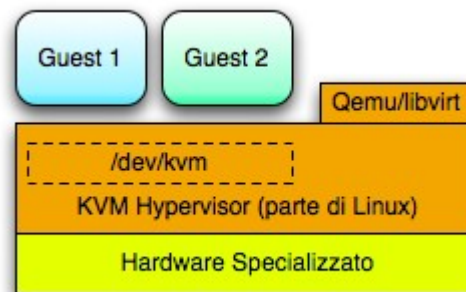


- Tipi di Hypervisor: *Native* (gli originali IBM) e *Hosted* (VirtualBox e VMWare workstation)



L'Hypervisor di KVM nel dettaglio

- **KVM** per funzionare sfrutta una configurazione ibrida *Native/Hosted*:



- **Kernel space** (*Native*): il modulo *KVM*
- **User space** (*Hosted*): *libvirt* e *qemu*



Supporto a KVM

- Per verificare la disponibilità delle estensioni per la virtualizzazione è possibile lanciare il seguente comando:

```
# egrep -c '(vmx|svm)' /proc/cpuinfo  
4
```

Se la risposta è maggiore di uno, allora il sistema supporta KVM.

- Dal filesystem /proc è possibile reperire un numero enorme di informazioni.
- Il numero di cpu, il tipo di architettura, etc.



Installazione del software necessario

- Essendo parte del Kernel, l'unico altro software necessario riguarda lo userspace:

```
# apt-get install qemu-kvm libvirt-bin
```

- Il metodo di installazione dipende dalla distribuzione utilizzata
- Molte distribuzioni Enterprise hanno già l'ambiente virtualizzato pre installato



virsh: lo strumento di gestione

- Capire se le componenti sono correttamente installate:

```
# virsh --connect qemu:///system list
```

```
Id Name
```

```
State
```

```
-----
```

- Il sistema di virtualizzazione è funzionante (nessuna virtual machine è in esecuzione)



network: comunicare con l'esterno

- Diverse tipologie di reti configurabili:
 - Nativa (rete interna logica visibile solo dalla macchina)
 - Bridge (rete esposta sull'interfaccia reale)
 - Opzioni da aggiungere per un'interfaccia bridge:
bridge_ports eth0
bridge_fd 9
bridge_hello 2
bridge_maxage 12
bridge_stp off



virt-install: installare la VM

- Per installare la virtual machine:

```
# virt-install --connect  
qemu:///system -n ubuntu-vm -r 512  
--vcpus=2 --disk  
path=/home/virtuals/ubuntu-  
vm.img,size=5 -c  
/home/virtuals/ubuntu-10.04.2-server-  
amd64.iso --vnc --os-type linux --os-  
variant ubuntu-karmic --accelerate  
--hvm --nonetworks
```



virt-install: opzioni utilizzate

- **--connect qemu:///system** indica a virt-install quale deve essere l'hypervisor target
- **-n** indica il nome della Virtual Machine, in questo caso squeeze-vm
- **-r** indica il quantitativo della ram virtuale in megabytes
- **--vcpus=2** indica il quantitativo di cpu virtuali da assegnare alla VM.
- **--disk-path=/home/virtuals/ubuntu-vm,size=10** indica a virt-install dove creare l'immagine per il disco virtuale e la sua dimensione. Può anche essere usato un dispositivo a blocchi (come ad esempio /dev/sdb1), in questo caso l'opzione size non avrà alcun effetto.
- **-c /path/file.iso** indica a virt-install il path dell'immagine di installazione del sistema guest. È possibile utilizzare anche il cdrom indicando /dev/cdrom
- **--vnc** indica a virt-install di abilitare la console VNC non appena la macchina virtuale è stata creata (INDISPENSABILE per seguire l'installazione).
- **--os-type linux --os-variant debiansqueeze** indicano a virt-install che tipo di sistema operativo verrà installato sul guest
- **--accelerate --hvm** indica a virt-install che il guest dovrà sfruttare l'accelerazione fornita da KVM. Ricordiamo che in realtà Qemu e KVM sono legati a doppio filo e il secondo può essere visto come un acceleratore per Qemu. inoltre la seconda opzione specifica che il guest dovrà essere un sistema completamente virtualizzato (e non paravirtualizzato come succederebbe con Xen)
- **--network=** indica a virt-install il tipo di network relativo alla macchina virtuale (in questo caso, nessuna quindi --nonetworks)



vncviewer: interagire con la VM

- Per interagire con la virtual machine
vncviewer 127.0.0.1:0
- Shell interattiva (con grafica, mouse e tastiera) per controllare la VM



La via facile: Virtualbox

- Complicato?
- Lungo?
- Virtualbox, un'alternativa per l'utilizzo locale e quotidiano



Fine

Conclusioni

