



Linux Virtuale

Come si può virtualizzare sotto Linux

Micky Del Favero
micky@linux.it

BLUG - Belluno Linux User Group
Linux Day 2007 - Belluno 27 ottobre 2007



Virtualizzazione, ovvero?

Per virtualizzazione si intende la creazione di una versione virtuale di una risorsa normalmente fornita fisicamente. Qualunque risorsa hardware o software può essere virtualizzata: sistemi operativi, serveri, memoria, spazio disco, sottosistemi. Un tipico esempio di virtualizzazione è la divisione di un disco fisso in partizioni logiche.

Meccanismi più avanzati di virtualizzazione permettono la ridefinizione dinamica tanto delle caratteristiche della risorsa virtuale, tanto della sua mappatura su risorse reali.

La virtualizzazione permette l'ottimizzazione delle risorse e la capacità di far fronte a esigenze specifiche secondo il più classico paradigma dell'on demand. *Wikipedia*



Vantaggi

Semplicità di gestione I dispositivi virtuali non cambiano al cambiare dell'hardware.

Sicurezza Codice maligno su macchine configurate a dovere non crea danni.

Affidabilità Macchine virtuali possono essere facilmente migrate su server fisici diversi.

Consolidamento Anche sistemi diversi possono essere virtualizzati su un unico server fisico.

Risparmio Un'unica macchina anziché cinque.



Come fare?

Software di virtualizzazione

Un software che mappa dispositivi virtuali in reali al fine di presentare al sistema operativo ospite una macchina virtuale su cui girare.

Hardware virtualizzabile governato da un software (*hypervisor*)

CPU con apposite istruzioni gestite dall'hypervisor e una mappa dei dispositivi virtuali in quelli reali gestita dal software presenta al sistema operativo ospite una macchina virtuali su cui girare, il vantaggio di questa soluzione sta nel fatto che l'ospite non ha bisogno di modifiche e gira a velocità pressoché identica a quella che avrebbe se girasse nella macchina reale.



Hardware Virtuale?

Nuovi processori basati sulla tecnologia VT *Vanderpool* di Intel o AMD-V *Pacifica* di AMD espongono ad un *hypervisor* le istruzioni necessarie per permettere ad esso di far girare macchine virtuali governate da sistemi operativi **non** modificati senza incorrere in significanti perdite di prestazioni.

Xen e KVM usano questa tecnologia.



Soluzioni Libere

Alcune soluzioni libere disponibili per Linux:

vserver

uml

qemu

xen

lguest

kvm



Soluzioni Libere

Linux Virtuale

vserver

Un sistema di partizionamento basato sui *Security Contexts* che permette di creare più *Virtual Private Servers (VPS)* simili ad un server normale che condividono le stesse risorse hardware e lo stesso kernel.

uml

User Mode Linux: un sistema di virtualizzazione con la capacità di emulare più risorse di quante ne abbia la macchina fisica e con la capacità di limitare l'accesso alle risorse fisiche.

Linux Virtuale – p. 7



Soluzioni Libere

Linux Virtuale

qemu

Un emulatore di processore e virtualizzatore libero. Può emulare vari tipi di CPU diverse (x86, x86_64, ARM, SPARC, SPARC64, PowerPC, PowerPC64, MIPS, m68k (Coldfire), SH-4, Alpha) e con un modulo del kernel apposito: *kqemu*, ottenere buone prestazioni anche in assenza di hardware fisico con estensioni per la virtualizzazione (Intel VT Vanderpool o AMD-V Pacifica).

Linux Virtuale – p. 8



Soluzioni Libere

xen

Un monitor di macchine virtuali libero che supporta l'esecuzione di più sistemi operativi ospiti con livelli di prestazione quasi pari alla macchina reale su hardware con estensioni per la virtualizzazione (Intel VT Vanderpool o AMD-V Pacifica). Solamente per domU, è incluso nel kernel vanilla dal 2.6.23 (il kernel della virtualizzazione!).

Iguest

Un hypervisor minimo per il kernel, scritto da Rusty Russel per sviluppatori e utenti che vogliono sperimentare con la virtualizzazione con il minimo possibile di complessità. È molto performante: 10 volte più veloce di qemu base. Dal 2.6.23 fa parte del kernel vanilla.

Linux Virtuale – p. 9



Soluzioni Libere

kvm

Kernel-based Virtual Machine: una soluzione per virtualizzazione completa per hardware x86 con estensioni per la virtualizzazione (Intel VT Vanderpool o AMD-V Pacifica). Consiste di due parti, una in kernel-space consistente di due moduli: *kvm.ko* il core dell'infrastruttura di virtualizzazione e *kvm-intel.ko* per hardware Intel o *kvm-amd.ko* per hardware Amd, e una in user-space consistente in una versione modificata di *qemu*. Con kvm è possibile far girare più macchine virtuali in cui girino versioni non modificate di Linux e di altri sistemi anche non liberi. È incluso nel kernel vanilla dal 2.6.20, prima soluzione per la virtualizzazione ad essere inclusa nel kernel ufficiale.

Linux Virtuale – p. 10



Alcune applicazioni

Linux Virtuale

Server virtuale.

Sistema di controllo di un robot.

Lan virtuale.

Linux Virtuale – p. 11



Server virtuale

Linux Virtuale

Un server in tutto e per tutto uguale ad uno reale, ma che gira in una macchina virtuale.

```
kvm -k it -m 512 -net nic,${mac} \  
-net tap,script=${ifup} \  
/export/img/kiwi.qcow
```

`${ifup}` è uno script che aggiunge ad un bridge l'interfaccia `tap` di `kvm` per permettere alla macchina virtualizzata di avere un collegamento con la lan.

Dal punto di vista dell'utente la macchina è indistinguibile da una reale.

Vantaggi: backup dell'intera macchina banale e migrazione su hardware diverso indolore: basta copiare semplicemente un file.

Linux Virtuale – p. 12



Sistema di controllo di un robot

Consolidamento di un sistema di controllo di un robot in un server Linux con qemu.

```
qemu -k it -m 256 -localtime -snapshot \  
-net nic,macaddr=52:54:00:12:34:09 \  
-serial /dev/ttyS0 /export/img/bmek-A.qcow
```

Il sistema sopra descritto è attualmente in fase di test e sembra poter sostituire una macchina reale senza problemi. Girando su un disco read-only la macchina risulta inattaccabile da virus, malware, etc, nonostante il sistema operativo che la governa sia un colabrodo.

È facile prevedere in futuro molte applicazioni che non possono essere riscritte o portate su sistemi operativi attuali e/o sicuri continueranno a girare in una macchina emulata con le caratteristiche della macchina per cui sono nate.



Lan virtuale (1)

Lan virtuale di macchine virtuali.

```
vde_switch -s /tmp/switch1 -tap tap0
```

```
vdeq kvm -k it -m ${mem1} \  
-net vde,sock=/tmp/switch1 \  
-net nic,macaddr=${mac1} \  
/export/img/limone.qcow
```

```
vdeq kvm -k it -localtime -m ${mem2} \  
-net vde,sock=/tmp/switch1 \  
-net nic,macaddr=${mac2} \  
/export/img/WXP.qcow
```



Lan virtuale (2)

WXP.qcow = *Warning! may eXPlode!*

Nell'esempio due macchine, una governata da Linux e una da un sistema pericoloso, sono collegate fra loro e ad Internet attraverso uno switch virtuale *vde_switch*.

Vantaggi:

- studio
- monitoraggio
- gestione semplificata
- risparmio energetico



Domande?

Grazie per l'attenzione.

