

Netkit: il laboratorio virtuale per lo studio di reti

Sandro Doro

ITIS “C.Zuccante” - Venezia–Mestre
Corso Serale Sirio

4 settembre 2010



Programma

- 1 La virtualizzazione
 - UML
 - QEMU
 - Applicazioni
- 2 Netkit
 - Linee guida
 - Esperienze riproducibili
 - VisualNetkit
- 3 Netkit4TIC
 - Struttura
 - Esperienze riproducibili
 - La nuova release v3.0
 - Direzione futura



Programma

- 1 La virtualizzazione
 - UML
 - QEMU
 - Applicazioni
- 2 Netkit
 - Linee guida
 - Esperienze riproducibili
 - VisualNetkit
- 3 Netkit4TIC
 - Struttura
 - Esperienze riproducibili
 - La nuova release v3.0
 - Direzione futura



La virtualizzazione

Introduzione

- Nel settore dei computer la virtualizzazione è una tecnica per nascondere le caratteristiche fisiche delle risorse computazionali.
- Da alcuni anni si stanno diffondendo progetti il cui scopo è quello di simulare altri sistemi, sia hardware che software.
- La virtualizzazione di una intera macchina apre nuovi scenari nella sperimentazione con il software.
- Questa idea non è nuova poichè risale all'epoca d'oro dei mainframe.



Elenco implementazioni

http://en.wikipedia.org/wiki/Virtual_machine

L'elenco è molto lungo e quindi citeremo solo una parte:

- Xen (<http://www.cl.cam.ac.uk/Research/SRG/netos/xen/>)
- VMware (<http://www.vmware.com/>)
- Virtual Server (<http://tinyurl.com/2okxgcg>) Virtual PC (<http://tinyurl.com/2jr7a7>)
- QEMU (<http://fabrice.bellard.free.fr/qemu/>)



Supporto Hardware

http://en.wikipedia.org/wiki/Intel_Virtualization_Technology

La Intel e la AMD hanno sviluppato indipendentemente delle estensioni alla architettura x86 per la virtualizzazione:

- La tecnologia Intel VT (IVT) è disponibile sui processori Pentium 4 6x1/2, Pentium D 9x0, Xeon 3xxx/5xxx/7xxx, Core Duo (escluso T2300E) e Core Duo 2 (non tutti).
- La tecnologia AMD Virtualization (AMD-V) è disponibile sui processori K8 AMD a partire dallo stepping F in avanti.



User Mode Linux

<http://www.user-mode-linux.org>

Il progetto è ospitato sul sito sourceforge.net e il suo ideatore e autore è Jeff Dike. Si è laureato al MIT e in seguito ha lavorato presso la Digital fino al 1993. Nel decennio successivo ha lavorato come indipendente ed è diventato un Linux kernel developer nel 1999. Dal 2004 lavora presso la Intel.

Contribuisce al progetto anche un giovane italiano: Paolo Giarrusso.

A partire dalla versione 2.6.0 UML è parte integrante del kernel di Linux.



User Mode Linux - caratteristiche

- Permette di eseguire multipli sistemi Linux (indicati come guest) in un normale sistema Linux (indicato come host) ossia da un pc con installato GNU/Linux posso “creare” un altro sistema GNU/Linux anche di una diversa distribuzione.
- È un sistema di virtualizzazione di tipo emulativo ossia ha lo scopo di riprodurre accuratamente le **funzionalità** di un sistema reale ma con una limitata velocità.

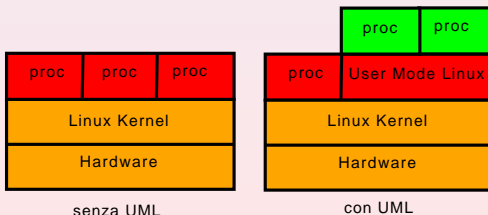


User Mode Linux - come lavora

Un programma che necessita dell'uso dell'hardware (scheda video, tastiera, ecc) richiede il servizio al kernel. Nel caso di utilizzo di UML la richiesta viene fatta al kernel User Mode Linux:

Modalita' di funzionamento

\$Id: how-works.xml,v 1.1 2008-09-19 04:41:40 doros Exp \$



QEMU processor emulator

<http://fabrice.bellard.free.fr/qemu/>

Il progetto è ideato e coordinato dal francese Fabrice Bellard. È un emulatore multiplatforma che permette di eseguire del codice Linux compilato per una particolare CPU su di un processore x86: per esempio codice SPARC o ARM. Per avere una maggiore velocità deve essere compilato anche il modulo acceleratore kqemu.



Applicazioni

I contesti in cui si utilizza la virtualizzazione di un intero sistema operativo sono ad esempio:

- Ambiente di testing di nuovo software
- Ambiente di simulazione di protocolli di rete
- Costruzione di Honeynet virtuali
- Costruzione di Virtual Host per Hosting Service



UML - Ambiente di testing di nuovo software (1)

Il progetto OpenSWAN (IPsec per Linux):

- si basa su righe di codice estremamente complicate per la comunicazione in rete
- cifra tutto il traffico di rete
- Utilizza un sistema di chiavi dinamico che rende praticamente impossibile ad un osservatore di capire cosa viene trasmesso
- occorrono circa 6 sistemi per testare completamente il codice prodotto e quindi è impossibile o troppo costoso tenere un tale sistema disponibile solo per questo scopo



UML - Ambiente per testing di nuovo software (2)

Il progetto BorpLAN (<http://sourceforge.net/projects/borplan>):

- si tratta di una applicazione “web based” realizzata come stage estivo A.S. 2005 da Simone Veronese ex studente dell’ITIS “C.Zuccante” di Mestre (VE) e finanziata dalla Cassa di Risparmio di Venezia. Lo scopo della applicazione è costruire un sistema di controllo capillare degli accessi alla rete scolastica tramite l’uso di un browser e di packet filter.
- per testare completamente il codice prodotto occorrono almeno 2 router, 2 pc client e un web server e quindi è difficile e costoso mettere a disposizione una tale struttura. Inoltre nel caso di bugs il sistema deve risultare pronto per l’accettazione e la verifica del bug stesso e per la successiva risoluzione.



Ambiente di sperimentazione di protocolli di rete

Il progetto UMTS/TIC corsi C1 e C2 (2003/2004 e 2006/2007) è un progetto del MIUR per la formazione del personale interno (ATA e docenti) sulla infrastruttura tecnologica.

Si sono verificati alcuni problemi:

- mancanza di un laboratorio creato appositamente e sempre accessibile ai corsisti per effettuare le loro prove
- mancanza di un sufficiente numero di computer
- impossibilità di effettuare le esercitazioni a casa



Laboratorio di sistemi/informatica

Negli istituti tecnici a indirizzo informatica (ABACUS/Sirio) nell'ultimo anno di corso, a volte anche prima, si studiano le reti e le applicazioni web. Per l'assimilazione effettiva da parte degli studenti è opportuno:

- mettere a disposizione **per ogni studente** un gruppo di sistemi da configurare e amministrare
- avere lo stesso sistema **a casa**



Playgroud per sperimentazione 3GPP

- Playgroud è inteso come tecnologia per ambienti di test dove si può giocare (play) con le ultime novità tecnologiche.
- 3GPP ossia Third Generation Partnership Project è un accordo di collaborazione, formalizzato nel dicembre 1998, fra enti che si occupano di standardizzare sistemi di telecomunicazione in diverse parti del mondo (da Wikipedia). In tale ambienti si sperimenta l'IPv6 mobile e il sistema IP Multimedia System (IMS).



Costruzione di Honeynet (vasi di miele)

Gli Honeynet sono dei sistemi installati sulle reti e servono per attirare eventuali intrusi e per studiare le loro mosse. Queste reti di sistemi sono altamente controllate e quando uno dei sistemi è attaccato esso cattura tutta l'attività dell'intruso. Un articolo che spiega come costruirli è: "Know Your Enemy: Defining Virtual Honeynets" tradotti in italiano in una serie di tre articoli(1, 2, 3)



Costruzione di Virtual Host per Hosting Service

La vendita di servizi di hosting è un mercato in forte crescita e spesso chi li utilizza desidera passare dalla soluzione di delega della gestione dei propri servizi al pieno possesso degli stessi per ottenere più flessibilità. La soluzione più vantaggiosa è sicuramente l'acquisto di un sistema virtuale.



Programma

- 1 La virtualizzazione
 - UML
 - QEMU
 - Applicazioni
- 2 **Netkit**
 - Linee guida
 - Esperienze riproducibili
 - VisualNetkit
- 3 Netkit4TIC
 - Struttura
 - Esperienze riproducibili
 - La nuova release v3.0
 - Direzione futura



Cos'è Netkit - <http://www.netkit.org>

Netkit è il risultato del lavoro di alcune persone del Networks Research Group dell'Università di Roma 3 e del Linux User Group LUG Roma 3. Il software è composto da:

- Un insieme predefinito di comandi per il setup di macchine virtuali
- Un filesystem con preinstallato il software necessario per le sperimentazioni
- Un kernel UML linux



Linee guida

Netkit è stato concepito come un ambiente a basso costo per esperimenti di rete. All'interno del suo ambiente possono essere “creati” e interconnessi dei completi router, switch e host. Tali apparati sono virtuali ma possono operare con molte delle caratteristiche possedute da quelli reali.



Considerazioni generali

- Basato su User Mode Linux
- Ogni apparato di rete è una linux box
- Linux supporta quasi tutti i protocolli di rete e può essere configurato come bridge/switch o come router



Struttura

- Le varie istanze che simulano gli apparati di rete sono create all'interno dello stesso host
- Le varie istanze sono interconnesse in domini di collisione (hub/switch)
- Il ruoli dei nodi sono configurabili




Esperienze riproducibili

- Esperienze base: rete minimale con due host, tabelle di routing, protocollo ARP, protocollo RIP.
- Esperienze applicative: configurazione di DNS e Mail server.
- Esperienze avanzate: esperienze su switch e STP.
- Esperienze su routing interdominio (bgp): routing tra Autonomous System.



Visual Netkit

- <http://code.google.com/p/visual-netkit/>
- È un ambiente grafico scritto in C++ e Qt4 che permette di configurare un laboratorio Netkit in modo semplice ed intuitivo.
- Alla base del progetto la sua architettura a plugin che permette di abilitare funzionalità o servizi dei nodi della rete in funzione delle necessità degli utenti.
-  YouTube Video



Programma

- 1 La virtualizzazione
 - UML
 - QEMU
 - Applicazioni
- 2 Netkit
 - Linee guida
 - Esperienze riproducibili
 - VisualNetkit
- 3 **Netkit4TIC**
 - **Struttura**
 - **Esperienze riproducibili**
 - **La nuova release v3.0**
 - **Direzione futura**



Netkit4TIC - <http://www.tic.fdns.net/tic/html/lab.html>

Questo progetto è nato nel 2003 dalla necessità nei corsi UMTS/TIC C2 di un laboratorio di sperimentazione “permanente”.

Le aree di interesse, intersecate tra loro, si possono dividere in:

- problematiche di rete: lo stack TCP/IP e la progettazione in generale di reti
- configurazione servizi: Web, DNS, directory services, file share, Kerberos, Terminal Server
- problematiche sulla sicurezza: cifratura, certificati elettronici, VPN



Struttura

È composto da due parti:

- 1 Live-DVD: è una distribuzione GNU/Linux in grado di “eseguire” le esperienze senza bisogno di installazione. Contiene:
 - una versione di Knoppix elaborata per ottenere migliori prestazioni con UML.
 - un kernel UML e un filesystem con preinstallato una parte della distribuzione GNU/Linux Debian Lenny
 - un kernel UML e un filesystem con la distribuzione LEAF/Bering con librerie uClibc per sistemi embedded.
 - una versione leggermente personalizzata di Netkit
- 2 Internet: dal sito <http://www.tic.fdns.net/tic/html/lab.html> sono scaricabili le esperienze virtuali (qualche KiB) in formato archivio compresso (tgz).



lo stack TCP/IP e la progettazione di reti

Esperienze riproducibili

- bridge, Bridge+STP, VLAN, proxyARP
- tabella di routing (statica), protocollo OSPF, Controllo LAB
- Qualità del Servizio (QoS)
- Esercizi: NetPractice



Web, DNS, directory services, Kerberos, cluster HA, file share, LTSP

Esperienze riproducibili

- file sharing: Samba, Samba Enterprise e HA Samba
- directory service: OpenLDAP
- Kerberos V e Single Sign On
- HA-Cluster, AFS, XFS
- Terminal Server
- Monitoraggio reti via SNMP



Sicurezza

Esperienze riproducibili

- Firewall
 - a singola area interna
 - con area interna e area smilitarizzata (DMZ)
 - con due aree perimetrali e backbone interno
- PKI: OpenCA una infrastruttura Open Source
- Protocolli SSH e SSL
- VPN
- Sistemi di rilevazione intrusioni (NIDS)



Novità della versione 3.0 (YouTube Video)

Host

- Visual Netkit

Guest Kernels

- Linux 2.6.22 UML di Netkit
- Linux 2.6.26 UML di Debian Lenny (DFSG)

Guest Filesystem

- nodo Debian Lenny di 2GiB (occupato al 51%)



selezione nodo Lenny

v-command

- selezione del kernel:
-k \$NETKIT_HOME/kernel/lenny

l-command

- selezione del kernel:
vm[k]=\$NETKIT_HOME/kernel/lenny



selezione nodo Bering

v-command

- selezione del kernel:
-k \$NETKIT_HOME/kernel/ub.krnl
- selezione del software da installare sul Bering:
-append=LRP=root,config,etc, ... ,customvm

l-command

- selezione del kernel:
vm[k]=\$NETKIT_HOME/kernel/ub.krnl
- selezione del software da installare sul Bering:
vm[append]=LRP=root,config,etc, ... ,customvm



Open-Source Linux

aggiunta connessione di rete reale

parametri

- default netkit
- `-append="ethX=tuntap,tapY"`



Nella prossima versione 4.0

Host

- Utilizzazione di un più recente Knoppix (5.3.1 o 6.0) e con patch SKAS3 o meglio ancora SKAS4

Guest

- abbandono della distribuzione LEAF/Bering sostituita in toto dalla nuova Debian Lenny in grado oramai di girare sulle box di tipo WRAP/ALIX.



Sommario

Riassunto

- Lo studio e la progettazione di reti e di servizi di rete è un elemento essenziale del nostro futuro.
- La modalità dell'apprendimento virtuale è in grande espansione.
- Esiste una via economica e valida: Netkit4TIC

Problematiche aperte

- Adozione limitata a: Mestre, Padova, Vicenza, Firenze, Palermo e Messina
- Costruzione di una comunità per la costruzione, collezione e condivisione di esperienze virtuali

