

21

Come fare a leggere questi dati generati da un programma windows?

A questo punto siamo in possesso di tutti gli elementi per sfruttare appieno le potenzialità di Linux: sappiamo destreggiarci (ma abbastanza) con i comandi di base della shell, abbiamo un'interfaccia grafica in costante miglioramento che ci aiuta, e sappiamo gestire i pacchetti con degli strumenti piuttosto potenti.

Tuttavia, pensare di abbandonare completamente il rassicurante Microsoft può essere difficile per chi ci ha passato praticamente tutto il proprio tempo informatico, e dunque questa lezione sarà interamente dedicata a descrivere due modi per usare (o almeno provare ad usare) programmi Windows in ambiente Linux.

C'è tutta una serie di programmi, quali programmi di automazione ufficio (Word, Excel, Power Point e simili) che ha degli equivalenti totalmente "free" capaci di leggere gli stessi files, e risolvono il problema alla base. Vedremo nel prossimo modulo di lezioni quale è l'offerta, e cosa ci possiamo aspettare in questo campo.

Ci sono invece numerosissimi programmi che siamo abituati ad usare, o che ci vengono consigliati continuamente da amici, che vorremmo, probabilmente, provare anche sotto Linux (in realtà, tenendo sia Windows che Linux sulla stessa macchina si risolve agevolmente il problema, ma voglio convincervi a lasciare Windows per sempre, come ho fatto io da qualche tempo).

Per risolvere questo problema ci sono due possibilità: un emulatore Windows, o una macchina virtuale per installare Windows ed averlo in una finestra Linux.

Le due soluzioni sono filosoficamente (e praticamente) molto diverse.

La prima ha come scopo fornire un programma Linux che sostituisca Windows. Tale programma deve conoscere il formato dei files eseguibili Windows, e far loro credere di essere il sistema operativo: per ogni chiamata fatta dai programmi stessi, l'emulatore dovrebbe rispondere correttamente (ad esempio, se il programma chiede di disegnare un punto sullo schermo, l'emulatore lo deve soddisfare rivolgendo la richiesta a Linux). Teoricamente, se i programmatori coinvolti nel progetto riusciranno mai nella titanica impresa di riscrivere tutte le librerie, questo permetterebbe di avere una piattaforma totalmente "free" per l'esecuzione di software per sistemi Microsoft. Tuttavia ci sono dei problemi legali (alcune librerie usate da Windows sono protette da copyright e non possono quindi essere riscritte, per uno strano concetto dei diritti d'autore americani), e pratici (non si sa ancora come riprodurre determinate chiamate, visti i segreti che le circondano), per cui la soluzione scelta attualmente è di usare le librerie dinamiche (i files con estensione dll) contenuti in Windows. Quando un programma fa una richiesta per un'operazione che l'emulatore non sa come servire, cerca tra le librerie Windows che conosce se l'operazione è offerta,

ed in caso affermativo le passa il controllo. Legalmente questo presuppone che ci sia una versione di Windows installata sullo stesso computer, che l'utente l'abbia comprata e ne sia il legittimo proprietario, andando un po' contro all'idea stessa di emulatore.

La seconda soluzione invece è più radicale: invece di simulare un sistema operativo, che deve dare una risposta a tutte le possibili richieste di tutti i programmi, si può simulare un computer intero! Una macchina virtuale infatti non è altro che un computer "finto", cioè un programma che fa da intermediario tra sistema operativo (che è già un'astrazione di un computer, per quanto detto nella prima lezione), ed altri programmi (tra cui sistemi operativi), mettendo a disposizione un disco, della memoria, un processore, delle periferiche, eccetera. Basterà dunque simulare delle componenti hardware reali, cioè offrire delle funzionalità uguali a quelle dell'hardware reale, per riuscire nell'impresa.

Come vedremo più avanti, esiste una macchina virtuale perfettamente funzionante per Linux (ma anche per Windows), che permette di avere un altro sistema operativo come applicazione. Ovviamente, bisogna comprare l'altro sistema per usarlo, contrariamente all'idea dell'emulatore!

Esiste un emulatore Windows particolarmente famoso per Linux, chiamato Wine. Per scaricarlo è sufficiente usare Red Carpet (e dovrebbe essere anche distribuito nei cd della RedHat), mentre per usarlo è necessario aver montato la directory contenente Windows come si è visto in una delle scorse lezioni. Bisogna modificare il file .wine/config, presente nella home directory, con un editor come emacs. Nella sezione intitolata "[Drive C]", bisogna mettere come path la directory in cui abbiamo installato Windows (nel mio caso /win) al posto di quella indicata.

L'emulatore ha qualche problema nel caso abbiate NTFS come filesystem Windows (nel qual caso provate ad usarlo come root, modificando il file /root/.wine/config e copiandolo dalla vostra home se non ci fosse), ma in generale è capace di eseguire piccoli programmi.

Su un computer con Windows 98, sono riuscito ad eseguire il solitario (Figura 1), il software di masterizzazione Nero (Figura 2), winzip (Figura 3) e, con mia grandissima sorpresa, Internet Explorer, con plug-in quali Flash e Java (Figura 4)!

Tempo fa, con un wine che mi ero ricompilato, ero riuscito a far partire numerose applicazioni che con questa versione installata dalla Red Hat non vogliono proprio funzionare. Se quindi voleste sapere davvero se una certa applicazione può funzionare con wine, dovete scaricare i sorgenti e ricompilarlo!

E' chiaro che per tutto quello che sono riuscito ad eseguire (tranne Internet Explorer, che comunque è partito dopo mille errori ed era un po' lento) relegano il wine alla stregua di un software giocattolo, anche se non è da escludere che in futuro vengano rilasciate versioni funzionali che ci permetteranno di valutare se un software Windows è utile, anche senza avere il sistema operativo Microsoft.

Diverso è il caso della macchina virtuale. Forse l'unica che davvero funziona è la vmware, che si trova all'indirizzo <http://www.vmware.com>.

E' un programma commerciale, che bisogna acquistare, anche se i produttori offrono una versione di prova gratuita per 30 giorni.

Possiamo dunque scaricare il pacchetto rpm della macchina virtuale, dopo esserci registrati al sito (mettete almeno l'indirizzo di posta elettronica corretto, perché vi invieranno un codice di registrazione senza il quale non potrete usare il programma).

Installare il programma è semplice come sempre:

```
rpm -i Vmware*rpm
```

Purtroppo non è finita qui: abbiamo bisogno di installare il pacchetto gcc (si trova con Red Carpet molto semplicemente), ed i sorgenti del kernel.

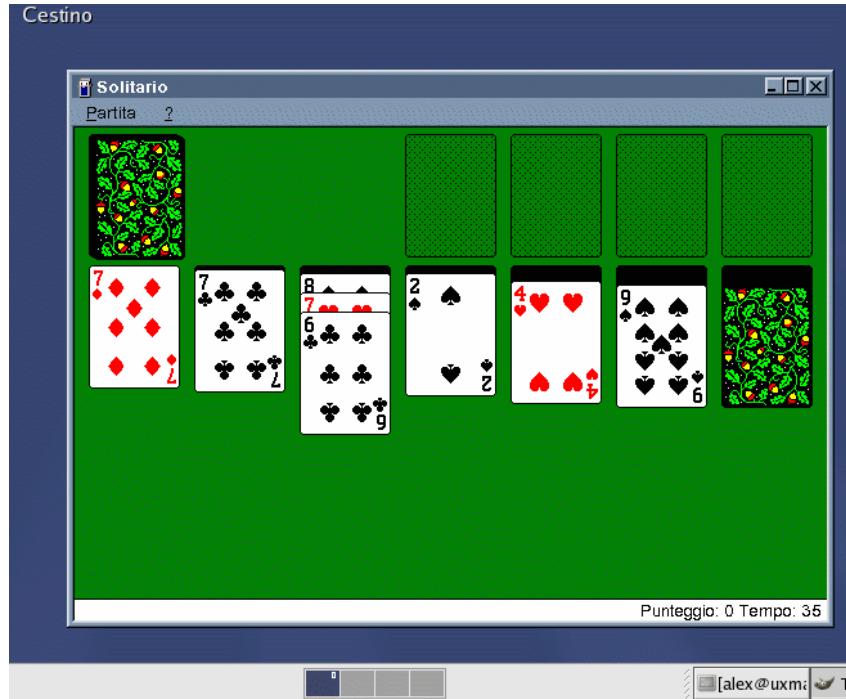


Figura 1: Wine esegue il solitario

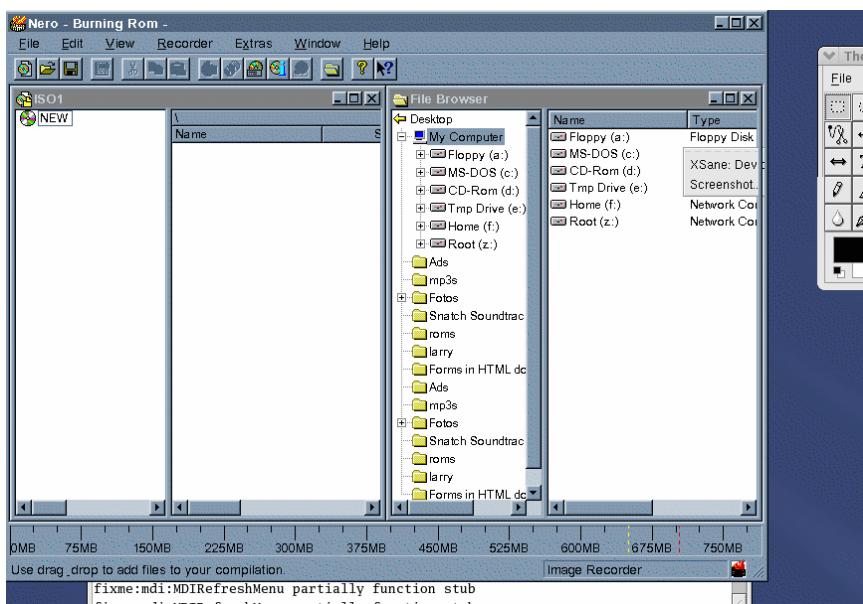


Figura 2: Nero su wine

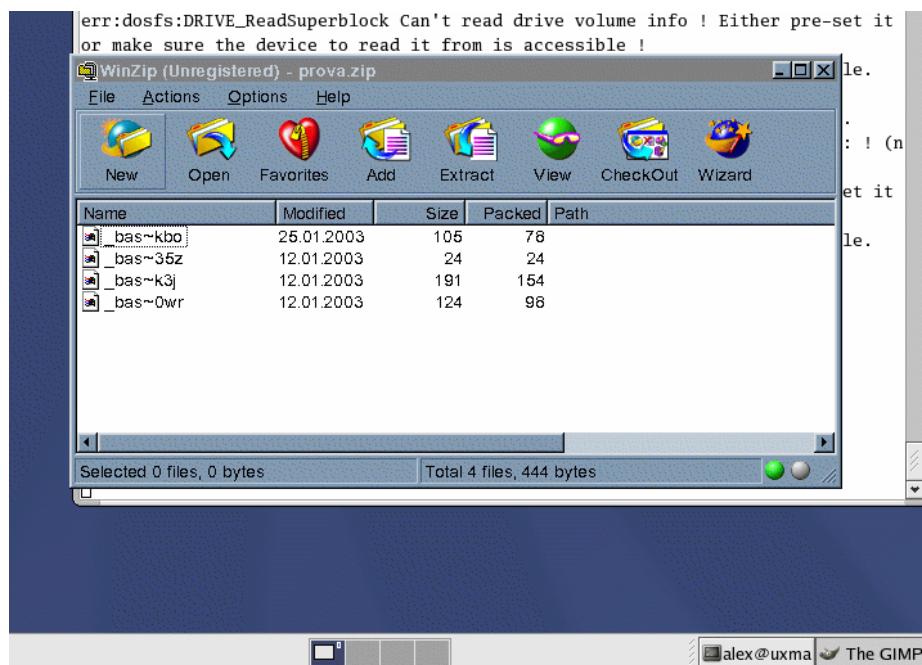


Figura 3: Winzip su wine

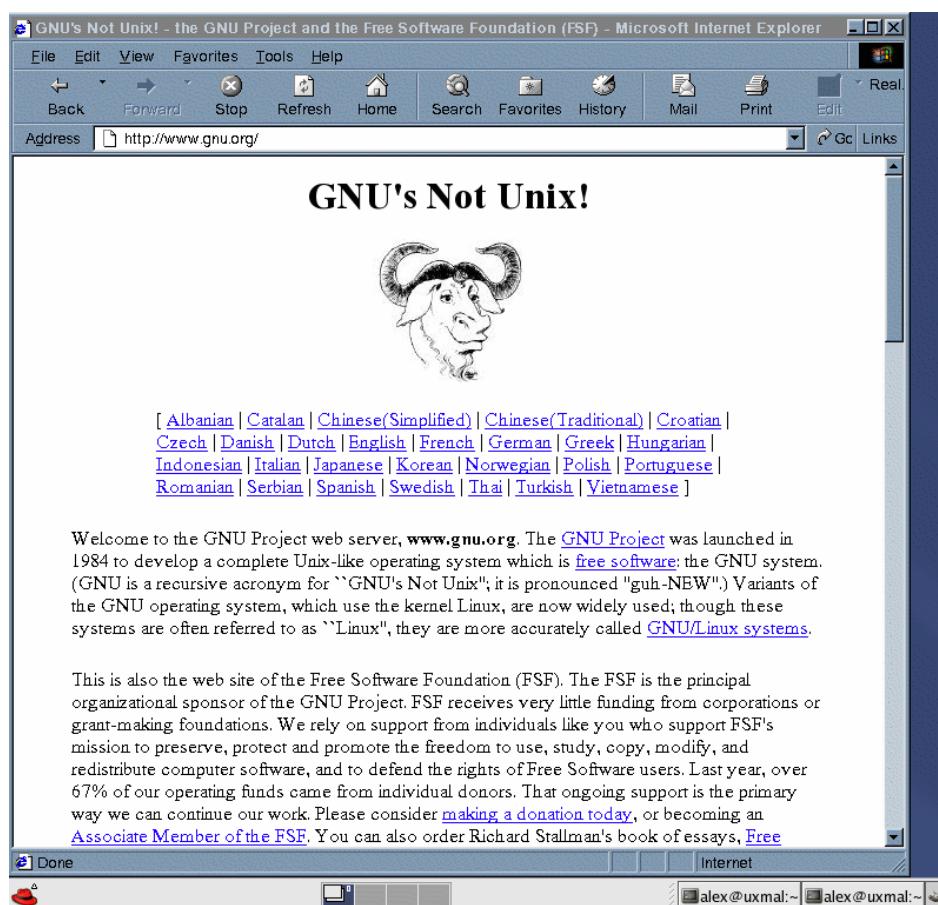


Figura 4: Internet Explorer gira su Linux!

Per sapere quali sorgenti scaricare, è necessario digitare il comando
 uname -a

Tra le informazioni restituite, oltre al tipo di sistema operativo ed al nome del computer, c'è anche la versione del kernel. Nel mio caso è 2.4.18-19.8.0, e quindi ho dovuto cercare un pacchetto chiamato kernel-source-2.4.18-19.8.0 (più qualcosa come i386 o simile). Si può andare sul sito ftp della RedHat (<ftp://ftp.redhat.com>) oppure cercare direttamente con google, per fare prima.

Installato questo, dobbiamo invocare il comando 'vmware-config.pl' (sempre come root), per poter configurare la macchina virtuale.

Dovremo dapprima confermare la solita licenza d'uso, in cui si garantisce che accettiamo la non responsabilità dei produttori del software per un suo cattivo funzionamento (tutto in inglese, ovviamente), quindi dirgli che abbiamo un compilatore, e confermare la directory che contiene i moduli (di solito trova tutto automaticamente).

Bisogna poi decidere se vogliamo la rete nella macchina virtuale (è sempre meglio rispondere di sì), e di che tipo: quando vi viene chiesto se volete rete di tipo host-only, rispondete di sì, perché è la più facile da installare e, a meno che non abbiate particolari esigenze, ci permetterà di usufruire di Internet anche dalla macchina virtuale.

A parte questo, potete confermare tutto quello che vi viene proposto.

Una volta terminato, chiamate la macchina virtuale con il comando 'vmware' (da utente normale). Dovrebbe apparirvi la schermata che vi richiede il numero seriale (dovreste averlo ricevuto per posta), per poi passare alla configurazione.

Potete usare il "configuration wizard" per semplificarvi la vita, e dire di installare una "standard virtual machine". Selezionate poi il sistema operativo operativo che intendete installare, confermate la destinazione dei files, decidete la dimensione del disco virtuale (è lo spazio disco che destinerete al Windows, quindi pensateci un po'), e scegliete il tipo di rete: la rete NAT vi permette di condividere un'unica connessione tra Linux e macchina virtuale, risparmiandovi un po' di grattacapi.

Ci siamo! Premete su "Power on", ed avrete un computer che fa il boot in una finestra! Potete entrare nel bios (con [F2]) e scegliere di fare il boot da cd, quindi inserire il cd di Windows, e far partire l'installazione!

All'interno della macchina virtuale, qualsiasi programma Windows funzionerà senza nessun problema, permettendoci così di usare quei programmi che riteniamo indispensabili.

Vi ho descritto una possibilità: ovviamente è equivalente a tenere Windows sul computer insieme a Linux, anche se la macchina virtuale dà un grosso vantaggio: se arriva un virus e vi mette scompiglio, sarà solo in una macchina virtuale. Il computer vero e proprio non ne risentirà assolutamente!