

Stolpervirtualisierung mit Xen

VON STEFAN SCHÄFER

Viele Server sind inzwischen virtuelle Xen-Maschinen. Dafür sind die Kernel von NetBSD und Linux bereits vorgesehen. Benötigt man eine Windows-VM, wird es aber schwierig, wenn man für die Treiber kein Geld ausgeben will.



Wer schon einmal unter der freien Version von Xen eine vollvirtualisierte Windows-Maschine betrieben hat, kennt das Problem: Sie ist viel zu langsam! Daß man von einem virtualisierten System keine Wunder erwarten darf, ist selbstverständlich und nicht das Problem. Die Zugriffe auf Netzwerkressourcen und die virtuelle Festplatte sind allerdings nicht akzeptabel. Die Ping-Antwortzeiten und der Datendurchsatz haben sich mit Xen 3.1 zwar gebessert, gut sind sie indes noch lange nicht. Es gibt gute Gründe, vollvirtualisierte Windows-Systeme zu betreiben, insbesondere wenn Branchen-Software serverseitig einen proprietären SQL-Server voraussetzt. Hier kommt Xen ins Spiel und man steht vor den beschriebenen Problemen. Die Gründe sind auch schnell gefunden: Die Netzwerkkarte und der Festplattencontroller müssen emuliert werden, da ein Durchreichen der realen Hardware an vollvirtualisierte Systeme nicht möglich ist.

Xen kennt zwei unterschiedliche Methoden der Virtualisierung. Bei der Paravirtualisierung wird auf den Xen-Server, im allgemeinen Xen-Host ge-

nannt, ein speziell modifiziertes Gast-System (Xen-Guest) als virtuelle Maschinen aufgesetzt. Sowohl der Kernel des Hosts als auch die Kernel der Gäste sind an Xen angepaßt. Ihnen eingepflanzt wurde der sogenannte Xen-Bus, eine schnelle Kernel-Kernel-Bus-Verbindung, über die es möglich ist, beliebige Hardware des physisch existenten Servers an die paravirtualisierten Maschine durchzureichen.

Techniken

Die Kontrolle darüber übernimmt die sogenannte Domäne 0 (kurz dom0), in der der Xen-Daemon selbst läuft. Auch sie ist letztlich nur eine virtuelle Maschine, allerdings in exponierter Stellung. Bei der Paravirtualisierung wird also keine Hardware emuliert, es wird auf reale Hardware zugegriffen.

Etwas anders sieht es allerdings bei den Netzwerkkarten aus, sie sind virtuelle Geräte, sowohl auf Seiten der Domäne 0 als auch bei den Gästen. Allerdings wird auch hier keine Hardware emuliert, sondern auf Techniken zurückgegriffen, die oh-

nehin im Kernel vorhanden sind. Beim Start des Xen-Daemons werden in der dom0 eine Reihe virtueller Netzwerkkarten für die Verbindungen zu den virtuellen Gästen (domU genannt) initialisiert. Diese virtuellen Netzwerkkarten und erste physisch vorhandene werden über eine virtuelle Netzwerkbrücke miteinander verbunden. Die Verbindung zwischen virtuellem Gerät in dom0 und den korrespondierenden real erscheinenden Geräten in den domUs erfolgt wiederum über den Xen-Bus.

Gänzlich anders sieht es bei der Vollvirtualisierung aus. Hier wird den virtuellen Gast-Maschinen mit Hilfe der speziellen Prozessor-Erweiterungen moderner AMD- und Intel-Prozessoren ein vollständiger PC vorgegaukelt und außer Xen wird eine Version von Qemu benötigt.

Installiert man auf einer vollvirtualisierten Maschine ein Windows, fällt auf, daß nach der Installation für mindestens ein PCI-Device der Treiber fehlt. Sucht man nach der Kennung, erhält man die Vendor-ID 5853. Das ist das »Xen Platform Device«. Auch die vollvirtualisierten Maschinen bekommen also den Xen-Bus zu Ge-

sicht, können aber nichts damit anfangen. Solche Treiber gibt es auch von Citrix, Novel und Red Hat; sie tragen den Namen »PV1 Driver«, sind proprietär und nicht frei verfügbar. Es gibt auch einen OpenSource-Treiber. Für den Download muß man aber dessen Name und Versionsnummer kennen.

Die Hauptseite des Entwicklers unter www.meadowcourt.org produziert nur eine Fehlermeldung. Die gerade aktuelle Version läßt sich aber per Versuch und Irrtum ermitteln. Zum Redaktionsschluß dieser Ausgabe paßte der Link www.meadowcourt.org/WindowsXenPV-0.8.4.zip.

Diese Datei befindet sich auch auf der freeX-DVD. Durch Variation der Versionsnummer kann man die jeweils neueste Version herausfinden. Für die Installation erwartet das Treiber-Paket ein .NET-2-Framework und den aktuellen Windows-Installer. Die Installation ist dann recht einfach. Die Zip-Datei wird entpackt und die *install.bat* aufgerufen. Zum Schluß muß das System neu gestartet werden.

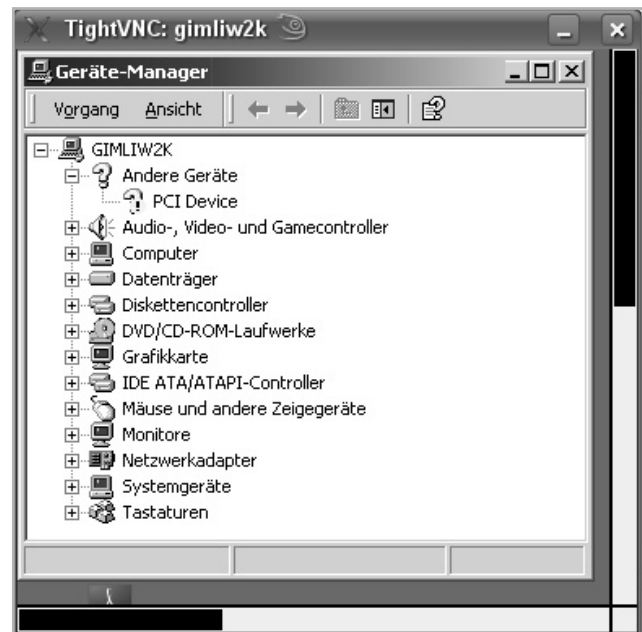
Leider funktioniert das Beschriebene erst ab Windows XP, leider aber nicht mit der Netzwerkkarte!

Philosophisches

Auch der Eintrag *type=netfront* (wie vielfach empfohlen) in der vif-Definition der zugehörigen Konfigurationsdatei der VM ändert daran nichts. James Harper, dem Entwickler des Open-Source-Treibers, ist die Misere nicht anzulasten. Er wird die noch verbliebenen Probleme mit seinen Treibern früher oder später lösen – eher früher, wenn man sieht, in welchen Abständen neue Releases der Treiber erscheinen.

Die Probleme liegen an einer anderen Stelle und treten immer an den Berührungspunkten zwischen Open Source und kommerzieller Software auf. Mit großem Marketingrummel wird ein Produkt freigegeben, natürlich nicht zur Gänze, sondern nur die große entwicklungsintensive Codebasis, die dann Freiwillige auf der ganzen Welt weiterpflegen dür-

Bild 1:
Ein unbekanntes
PCI-Gerät



fen. Die Sahnehäubchen, mit denen sich das Geld verdienen läßt, verbleiben, wo sie sind: gut behütet und geschützt.

Funktionierende PV-Treiber etwa gibt es längst, nur eben nicht umsonst. Hieraus einen echten Vorwurf zu stricken, fällt nicht leicht. Schließlich haben diese Firmen ja auch die Arbeit geleistet und wollen sich diese auch vergüten lassen. Überdies haben sie ja auch Unmengen mühsam erarbeiteten Code der Allgemeinheit geschenkt.

Auf der anderen Seite gewinnen derartig interessante Projekte wie Xen in kurzer Zeit viele Freunde, freie Entwickler opfern ihre Freizeit zum Wohle des Projekts und leisten damit einen ganz und gar nicht unerheblichen Beitrag an der Sache. Einen Beitrag, den die kommerziellen Initiatoren abschöpfen können, ohne je dafür etwas bezahlen zu müssen. Ungerecht würde die Bewertung, wenn ich nicht eine dritte Gruppe in die Betrachtung mit einbeziehen würde: Den kommerziellen Open-Source-Anwender, also jemand, der gerne die Open-Source-Basis und die Sahnehäubchen kostenlos hätte, um damit Geld zu verdienen.

Sind wir reine Nutznießer? Meines Erachtens nein. Ich jedenfalls bringe meine Erfahrungen mit den Projekten wieder in diese ein und stelle die Ergebnisse meiner Arbeit übli-

cherweise in Form funktionierender Konfigurationsdateien, die ich zum Download anbiete, der Allgemeinheit kostenlos zur Verfügung. Andere machen es genauso. Vor allem das Zurückfließenlassen praktischer Erfahrungen ist schließlich eine der Säulen, auf denen funktionierende Open-Source-Projekte stehen. Das wiederum kommt ja nicht zuletzt auch den Projekt-Initiatoren zugute. Überdies stellen kleine Open-Source-Software verwertende Unternehmen wie meines keine nennenswerte Konkurrenz zu Branchenriesen wie Citrix, Novell oder Red Hat dar, wir fischen in unterschiedlichen Gewässern. Bleibt einzig die Frage, wer den größten Gewinn dabei macht. ♦

