

Pasando dispositivos PCI del anfitrión por medio de un huésped KVM

SERVICIOS HUÉSPEDES

Ahora, el rápido y ligero KVM es capaz de pasar hardware PCI físico por medio de un sistema huésped.

POR OLIVER RATH, HANS-PETER MERKEL Y MARKUS FEILNER

Ramona D'Vioia, 123RF

El popular sistema de virtualización KVM que se encuentra en muchos sistemas Linux proporciona ahora acceso al hardware PCI real. Este acceso a los dispositivos PCI puede proporcionar mejor rendimiento que las técnicas de emulación de hardware convencionales, y en algunos casos, podría ser la mejor solución y la más práctica para conectarse con un dispositivo PCI poco usual que, independientemente del motivo, no se encuentre accesible desde el huésped por medio de la emulación. Para lograr el mejor soporte posible para este acceso a PCI, será mejor disponer de los últimos módulos del kernel

Un Poco de Historia

KVM comenzó como una variante de QEMU, y el desarrollo de KVM y QEMU continuó en paralelo durante algún tiempo.

El plan de los desarrolladores era que la gente se interesara en QEMU por la licencia GPL y luego ganar dinero por medio de la licencia comercial del módulo KQEMU, que aceleraba el emulador, haciéndolo casi cinco veces más rápido. Desafortunadamente, la presentación del hardware Vanderpool con el hardware para la virtualización lo dejó obsoleto, o al menos eso fue lo que dijo en su blog Avi Kivity de Qumranet [2], uno de los desarrolladores perteneciente a la empresa que se encontraba tras KVM.

para KVM (o el último kernel), pero como mínimo, el 2.6.26, preferiblemente con Userspace-Tools 0.12.2 [1].

La restricción de que sólo se ejecuta en procesadores Intel con soporte Vanderpool, o en su equivalente AMD, Pacifica, no es un problema. La mayoría de las CPUs, con la excepción de algunos Atoms, Celerons y algunas series OEM de bajo coste para portátiles, vienen con estas extensiones incorporadas. AMD las incluye en todas las CPUs recientes desde la aparición de los Dual Cores (con la excepción de Sempron).

Como regla general, Linux sólo puede pasar el hardware que el anfitrión no esté utilizando a una instancia virtualizada. Para que esto suceda, primero hay que descargar los módulos, o impedir que se carguen por medio de una lista negra. Como normalmente se hace en Gentoo, es buena idea no ligar los módulos con su propio kernel.

Listado 1: lspci --vv | grep IRQ

```
01 Interrupt: pin D routed to IRQ 10
02 Interrupt: pin D routed to IRQ 12
03 Interrupt: pin D routed to IRQ 12
04 Interrupt: pin ? routed to IRQ 9
05 Interrupt: pin A routed to IRQ 11
06 Interrupt: pin A routed to IRQ 12
07 Interrupt: pin A routed to IRQ 10
08 Interrupt: pin A routed to IRQ 11
09 Interrupt: pin A routed to IRQ 11
```

Desconectar las IRQs Compartidas

Es muy importante deshabilitar la funcionalidad de las interrupciones compartidas... y no sólo para los elementos del propio kernel. Actualmente, KVM sólo puede acceder a los dispositivos PCI con sus propias interrupciones. El acceso falla si múltiples dispositivos comparten una interrupción. Es buena idea analizar frecuentemente las interrupciones en uso con `lspi -vv | grep IRQ` (Listado 1) y luego comparar la salida con el dispositivo al que se desea acceder (Listado 2). En este ejemplo, el acceso no funcionará, ya que la interrupción 11 está compartida con tres dispositivos.

Sobre Debian

Los repositorios de Debian Lenny aún suministran la versión obsoleta KVM 77 si se le pregunta por ella mediante el comando `kvm -version`. (El próximo Debian Squeeze, así como Ubuntu 9.10 y otros derivados de Debian, proporcionan la versión v0.11.1).

Los paquetes para los i386 o AMD64 se encuentran disponibles para descargar desde el servidor de Debian. Para ello, como root, hay que ejecutar el comando `dpkg -i` para instalarlo. Si instaló una versión más antigua anteriormente, no debería experimentar ningún problema de dependencias.

`qemu-kvm --help` debería mostrar un mensaje de éxito (Listado 4). Ahora ya puede comenzar a trabajar con Debian.

El Listado 3 es un ejemplo con una tarjeta PCIe RDSI de cuatro puertos de Cologne Chip Designs. Su interrupción (19) sólo es usada por la tarjeta RDSI. Los administradores de sistemas Ubuntu o Debian pueden introducir los módulos que no necesiten en `/etc/modprobe.d/blacklist.conf`:

```
blacklist mISDN_dsp
blacklist hfcmulti
blacklist mISDN_core
```

El ejemplo muestra los módulos para la tarjeta RDSI HFC4S activa; hay controladores para el nuevo subsistema `mISDNv2`, el cual fue introducido con el kernel 2.6.28 y es altamente recomendable para las tarjetas RDSI HFC. Si está utilizando la herramienta de telefonía Asterisk, también necesita el canal del controlador `lcr_chan` del proyecto Linux Call Router [3].

Encontrar el ID PCI

Antes de pasar el acceso a una tarjeta PCI o PCIe a un sistema operativo virtualizado, hace falta el ID PCI, que se puede obtener por medio del comando `lspci`:

```
[...]
00:11.0 Network controller: AVM GmbH B1 ISDN
```

y luego pasárselo a KVM como un parámetro en la línea de comandos a la hora de ejecutarlo.

RDSI para el Huésped

Supongamos que se desea configurar el sistema huésped para que utilice la herramienta contestador automático de fax Capi-

suite conectada a la tarjeta RDSI Fritz del sistema anfitrión. El primer problema es que los únicos paquetes que se encuentran disponibles son RPMs para SUSE, y su modificación es un trabajo duro. Cuanto más moderno sea el kernel de Debian, más difícil será impedir que el compilador se queje. El ejemplo que sigue muestra un contestador automático Capi suite basado en Debian Etch, en el que ni los módulos ni el compilador causarían ningún problema, ya que el sistema utiliza aún el kernel 2.6.18. KVM ejecuta el sistema huésped en segundo plano y le asigna una dirección IP independiente para acceder por medio de SSH.

Tras instalar un sistema estándar, el huésped aún necesita los paquetes `build-essential`, `rpm` y `capiutils`, junto con las cabecezas. El sistema huésped también necesita los controladores de la tarjeta Fritz. Como el anfitrión normalmente usa estos controladores de forma exclusiva, con el antiguo módulo Hisax ISDN, habrá que utilizar una lista negra. Las siguientes dos líneas en `/etc/modprobe.d/blacklist.conf` es todo lo que se necesita:

```
blacklist
hisax_fcpcipnp
blacklist hisax
```

Listado 2: lspci -vv | less

```
01 [...]
02 00:11.0 Network controller: AVM GmbH B1 ISDN
03 Control: I/O+ Mem+ BusMaster+ SpecCycle-
MemWINVVGASnoop-ParErr- Stepping- SERR- FastB2B-
DisINTx-
04 Status: Cap- 66MHz- UDF- FastB2B- ParErr-
DEVSEL=fast >TAbort- <TAbort- <MAbort- >SERR-
<PERR- INTx-
05 Latency: 32
06 Interrupt: pin A routed to IRQ 11
07 Region 0: I/O ports at d800 [size=64]
08 Region 1: I/O ports at dc00 [size=32]
09 Kernel driver in use: b1pci
10 Kernel modules: b1pci
11 Capabilities: [40] Power Management version 3
12 [...]
```

hardware hasta que KVM se lo indique. El huésped pensará originalmente que Hisax es el módulo correcto. Para impedir que suceda esto, hay que crear un fichero `/etc/modprobe.d/blacklist-capi suite` en Etch con el mismo contenido que el mostrado anteriormente en `blacklist.conf`.

A la hora de ejecutar `kvm`, hay que pasarle el ID del dispositivo en la línea de comandos. El sistema huésped debería ahora cargar el módulo:

```
gast # lsmod | grep fcpci
fcpci 592768 1
kernelcapi 43680 2 capi,fcpci
```

Capiinfo indicaría en este momento que una tarjeta PCI se encuentra disponible en el huésped (Listado 5). Ahora no habrá nada que le impida configurar Capi suite como servidor de fax y como contestador automático.

Supongamos que se desea instalar una tarjeta DVB-T en el sistema huésped además de la tarjeta RDSI. La tarjeta DVB-T se va a utilizar como grabadora de vídeo digital con VDR. Es conveniente utilizar otro disco para almacenar las grandes cantidades de datos que la grabadora creará.

Listado 3: lspci -vv

```
01 08:04.0 ISDN controller: Cologne Chip Designs
GmbH ISDNnetwork Controller [HFC-4S] (rev 01)
02 Subsystem: Cologne Chip Designs GmbH Device b752
03 Control: I/O- Mem+ BusMaster- SpecCycle-
MemWINVVGASnoop-ParErr- Stepping- SERR- FastB2B-
DisINTx-
04 Status: Cap+ 66MHz- UDF- FastB2B- ParErr-
DEVSEL=medium>TAbort- <TAbort- <MAbort- >SERR-
<PERR- INTx-
05 Interrupt: pin A routed to IRQ 19
06 Region 0: I/O ports at 5000 [disabled] [size=8]
07 Region 1: Memory at c6000000 (32-bit,
non-prefetchable)[size=4K]
08 Capabilities: [40] Power Management version 2
09 Flags: PMEClk- DSI+ D1+ D2+
AuxCurrent=0mAPME(D0+,D1+,D2+,D3hot+,D3cold-)
10 Status: D0 NoSoftRst- PME-Enable- DSel=0 DScale=0
PME-
11 Kernel driver in use: hfc_multi
12 Kernel modules: hfcmulti
```

El anfitrión ahora ignorará estos controladores y el sistema huésped no detectará el nuevo

Listado 4: qemu-kvm --help

```
01 QEMU PC emulator version 0.10.50
(qemu-kvm-devel-88). Copyright (c) 2003-2008
Fabrice Bellard
02 usage: qemu [options] [disk_image]
03 [...]
04 -pcidevice
host=bus:dev.func[,dma=none][,name=string]expose
a PCI device to the guest OS.
05 dma=none: don't perform any dma translations
(default is to use an iommu) 'string' is used in
log output.
06 [...]
```

Como en el ejemplo anterior, hay que impedir que el sistema anfitrión acceda a la tarjeta DVB e introducir los controladores *dvb-ttpci*, *stv0299*, *saa7146_vv* y *saa7146* del Listado 6 en */etc/modprobe.d/blacklist.conf*. Para la mayoría de las tarjetas DVB, con los módulos del kernel no será suficiente; necesitan además el firmware en */lib/firmware*. Por último, KVM necesita el ID del dispositivo PCI de la tarjeta DVB. En este ejemplo, *lspci* lista el chip Philips SAA7146 como 00:06.0.

Conexión en Caliente PCI en KVM

En tiempo de ejecución, se puede forzar al sistema huésped para que utilice más dispositivos PCI. La consola QEMU Monitor, poco conocida, que se ejecuta con Ctrl + Alt + 2, ayuda en la tarea de añadir dispositivos. El clásico terminal con el fondo negro y el texto blanco permite a los usuarios el uso de algunos comandos prácticos. Se encuentra disponible información detallada, y si se teclea *help*, se puede encontrar información tanto de la sintaxis como de los comandos soportados. *info pci* lista todos los dispositivos PCI conocidos de la instancia virtual y *pci_add* permite añadir dispositivos, como otra tarjeta Ethernet:

```
(qemu) pci_add auto nic
model=e1000
```

```
OK domain 0 bus 0 slot 9 function
0
(qemu)
```

Con un estilo similar a la sintaxis utilizada en KVM, se puede utilizar *host =* para definir el ID PCI del sistema anfitrión. Para que esto funcione de forma correcta, los módulos del kernel *acpiphp* y *pci_hotplug* deben estar cargados en el sistema huésped. En este caso, *dmesg* mostrará información detallada del nuevo dispositivo PCI, y la lista *lspci* seguirá creciendo. Sin embargo, hay algo más aparte de todo esto: se puede utilizar la consola Qemu Monitor para añadir o quitar controladores, así como dispositivos USB y de almacenamiento, más o menos a voluntad. Cuando haya finalizado, pulsando Ctrl + Alt + 1 se regresará a la ventana familiar de KVM.

Listado 6: lsmod | grep dvb

```
01 dvb_ttpci 104576 18
02 dvb_core 99120 2
   stv0299,dvb_ttpci
03 saa7146_vv 49920 1 dvb_ttpci
04 saa7146 19160 2
   dvb_ttpci,saa7146_vv
05 ttpci_eeprom 2672 1 dvb_ttpci
06 i2c_core 26736
   7nvidia,stv0299,ves1x93,dvb_tt
   pci,videodev,ttpci_eeprom,i2c_
   piix4
```

Listado 7: lspci en el Huésped

```
01 00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma]
(rev 02)
02 00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA
[Natoma/Triton II]
03 00:01.1 IDE interface: Intel Corporation 82371SB PIIX3 IDE
[Natoma/Triton II]
04 00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 03)
05 00:02.0 VGA compatible controller: Cirrus Logic GD 5446
06 00:03.0 Ethernet controller: Realtek Semiconductor Co.,
Ltd.RTL-8139/8139C/8139C+ (rev 20)
07 00:04.0 RAM memory: Unknown device 1af4:1002
08 00:05.0 Network controller: AVM Audiovisuelles MKTG & Computer System
GmbH A1ISDN [Fritz] (rev 02)
09 00:06.0 Multimedia controller: Philips Semiconductors SAA7146 (rev
01)
```

Memoria

Los derivados virtualizados de Linux necesitan memoria. Si no se especifica el parámetro *-m*, KVM tomará por defecto un tamaño de 128MB de RAM, que es demasiado pequeño para Windows XP y probablemente demasiado pequeño para muchas de las distribuciones de Linux. El instalador de Fedora y el de YaST presentarán problemas si no tienen un mínimo de 256MB. Los usuarios con

servidores de virtualización que sufren de sobrecarga de memoria pueden mirar el Kernel SamePage Merging (KSM) [4], el cual analiza continuamente las páginas de RAM ocupadas y sólo mantiene una única instancia de cada página de memoria en el caso de que sean idénticas. Red Hat, los nuevos propietarios de Qumranet y Fedora, continúan con su desarrollo activamente.

Hardware en Masa

Ahora el huésped posee una gran selección de hardware y acceso exclusivo a los dispositivos PCI físicos (Listado 7). Una característica que los intrusos y los especialistas forenses siempre han apreciado consiste en interesarse más y más en otros campos. Los administradores pueden ejecutar soluciones personalizadas en sistemas protegidos; los geeks pueden ejecutar grabadoras de vídeo en segundo plano. Y si todo esto no es suficiente, puede esperar a la tecnología de procesadores Nano de Via y soñar con sistemas empotrados con ahorro de energía que ejecuten Windows y Linux en paralelo. ■

Listado 5: capiinfo en el Huésped

```
01 Number of Controllers : 1
02 Controller 1:
03 Manufacturer: AVM GmbH
04 CAPI Version: 2.0
05 Manufacturer Version:
   3.11-07(49.23)
06 Serial Number: 1000001
07 BChannels: 2
08 Global Options: 0x00000039
09 internal controller supported
10 DTMF supported
11 Supplementary Services
   supported
12 channel allocation
   supported(leased lines)
```

En Segundo Plano

No es necesaria ninguna interfaz de usuario para los sistemas virtuales en equipos en producción. El huésped puede ejecutarse completamente como un proceso en segundo plano y se puede utilizar SSH para las tareas administrativas:

```
# qemu-kvm -m 1024 -net nic,
vlan=0,macaddr=Z
00 :80:ad:11:11:11 -net tap
-pcidevice host=Z
05 :06.0 -nographic
-daemonize etch.img
```

Para poder asignar una dirección IP estática al sistema virtual usando DHCP, hay que pasarle una dirección MAC virtual cuando se ejecute el sistema y reservar dicha dirección para este sistema en el servidor DHCP.

RECURSOS

- [1] Kernel Virtual Machine: [http:// www.linux-kvm.org](http://www.linux-kvm.org)
- [2] Blog de Avi Kivity sobre KVM: [http:// avikivity.blogspot.com](http://avikivity.blogspot.com)
- [3] Proyecto Linux-Call-Router: [http:// www.linux-call-router.de](http://www.linux-call-router.de)
- [4] Kernel SamePage Merging: [http:// fedoraproject.org/wiki/Features/KSM](http://fedoraproject.org/wiki/Features/KSM)