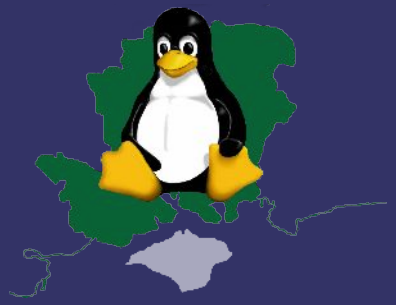

Desktop Virtualisation

Adam John Trickett

www.iredale.net
adam.trickett@iredale.net
PGP Key: 0xAF0DB8C8

In The Beginning...

- Computers were:
  - Very Expensive
  - Very Rare
  - Very slow
  - Ran one thing at a time
  - Ran for only one user

Time Passes...

- By the 1960s computers are:
  - Still expensive
  - Getting more common
  - Getting faster
  - Required to do more than one things at one

Multics

- Started in 1964
- MIT, GE & Bell Labs
- Multi-user, multi-tasking system
- Gave rise to Unix and hence Linux

- *Last Multics system decommissioned 30/10/2000*

Virtual Machines

- Pioneered by IBM in the 1960s
- 1966 IBM CP-40 & CP-67
- 1972 IBM System/370 & VM370

Virtual V Multitasking

- Run lots of programs at the same time
- Lots of users at the same time
- Can run on their own kernel

- Run lots of programs at the same time
- Lots of users at the same time
- All running on the same kernel

Time Passes...

- Computers get faster and cheaper
  - IBM VM systems get smarter
  - IBM VM systems run multitasking software
  - Unix systems get smarter
  - Unix systems run using hardware assisted VM
- IBM release the “PC” using the Intel 8086

PCs

- Primitive processor
- Limited memory
- Primitive single user operating system:
  - No multi-tasking
  - No VM
  - Reboot becomes a way of life
- Cheap

Time Passes...

- Processors get a lot faster
- Memory gets cheap and fast
- Operating systems grow up, Windows NT and Linux
  - Multi-user
  - Pre-emptive multi-tasking
  - BSOD for some users...!

Machine Emulators

- Desktop computers are fast enough to fully emulate the complete hardware of an older slower system, e.g.
  - VersatIle Commodore Emulator
  - DOSbox
  - MESS

QEMU

- Can emulate several processors
- Can emulate several hardware platforms
- Contains framework to run a virtual system on a regular PC
- Sub-component used by other emulators

Qemu Components

- CPU emulators
- Firmware
- VGA graphics emulator
- IDE subsystem & virtual disks
- Network subsystem
- PS/2 mouse and keyboard
- Sound etc

CLI Invocation

- Create a virtual file system:

```
qemu create -f qcow2 vdisk.qcow2 10G
```

- Start a Qemu session:

```
qemu-system-x86_64 -boot c \
-hda vdisk.qcow2 -cdrom cd.iso \
-soundhw all -m 256 -localtime -k en-gb \
-net nic -net user
```

Kqemu / "Qemu Accelerator"

- Kernel module
- Guest usermode code runs directly on host CPU
- Does not require special CPU
- Only x86 or AMD64
- Same scripts as Qemu
- Some old or odd OS wont run under kqemu

KVM QEMU

- Kernel module – in mainline kernel
- KVM an alternative to kqemu
  - Can be used just like Kqemu/Qemu
- Requires CPU x86/AMD64 with virtualisation features

VirtualBox

- Three Licences:
  - Open source
  - Proprietary personal
  - Proprietary commercial
- Uses some parts of Qemu
- Built in Qt based GUI and CLI
- Guest OS drivers
- Can use hardware virtualisation if available

Comparison

- Qemu/KQemu/KVM
  - CLI no native GUI
  - Can use CPU virtualisation
  - Guest graphics performance poor
  - Support for older hardware
- VirtualBox
  - GUI or CLI
  - Can use CPU virtualisation
  - Good guest graphics performance
  - Older OS hardware problems

Conclusion

- Use VirtualBox, unless:
  - Host not x86/AMD64 CPU
  - Guest not x86/AMD64 CPU
  - Old guest e.g. Windows 95
  - Don't need a GUI

Demo

Resources

- http://www.iredale.net/p/by-type/talk/
- http://www.hants.lug.org.uk/
- http://www.slideshare.net/

Thank You

Any Questions?