What are we using this Year?

- Mac-mini servers
- Intel core i7 quad core 8 hyperthreads
- 16GB of ram
- 2 x 256GB SATA SSD
- A pretty hefty server
- Less than $2k

Drawbacks
- One psu
- OOB is kind of a pain

Ubuntu / KVM
What is it?

• Virtualization of is the abstraction of the manifestation of a resource from the actual physical instance of that resource.

• What Computing/Network resources can be virtualized?
  • Virtually anything! :)

Anything?

- In the context of this course. We're interested in virtualization along two dimensions:
  - Services
  - Hosts
Resource/Service virtualization

- Examples:
  - Load-balancers
  - DNS Based GLB
  - HTTP(S) Virtual Hosting
  - MX records
  - Virtual Switches
  - Virtual Routers
  - Virtual Firewalls
Resource Virtualization - Continued

- HTTP virtual hosts
  - Multiple websites on one system
- Load Balancing
  - One (or many sites or applications) across many systems
  - Can be done at Layer-3/4/7
Host Virtualization

- Examples
  - VMware
  - Virtual-Box (used in class)
  - KVM
  - XEN
  - FreeBSD and Linux Jails
  - Windows Hyper-V
What problem are we attempting to solve with host virtualization.

- **Problem 1 – Idle capacity.**
  - Most of the machines in your datacenter are idle most of the time.
  - Capacity you're not using:
    - Cost money up front
    - Cost money to operate
    - Reduces your return on capital
  - Packing discreet systems into a smaller number of servers provides savings along virtually every dimension.
Problems - Continued

• Problem 2 – Provisioning
  • Spinning up a new service involves:
    – Acquiring the hardware
    – Building the server
    – Integration with existing services
  • With virtualization we're aiming to short-circuit that
    – Capacity is a resource
    – Machine instances may be cloned or provisioned from common basic images
    – Resources are purchased in bulk and assigned to applications as necessary.
Problems - Continued

- Problem 3 – Hardware abstraction
  - Operating systems, servers, and applications evolve at different rates.
  - Providing a common set of infrastructure resources means, virtualized systems are portable across servers.
  - Hardware failure can more easily be managed.
- Abstraction may come at a performance cost however. (some workloads are more expensive than others)
  - See:
Examples – Desktop Virtualization
Desktop Virtualization

- **Uses**
  - Prototyping services or applications before deployment
  - Utilities that don't run on your operating system
  - Isolation of sandbox environments from your desktop
  - Maintaining multiple versions of an environment for support purposes.
  - Staying familiar with unix while running windows (consider compared to the alternative (dual-booting))

- **Issues**
  - Emulating multiple computers on your laptop/desktop is somewhat resource intensive

- **Vmware player and VirtualBox are free.**
  - [https://my.vmware.com/web/vmware/downloads](https://my.vmware.com/web/vmware/downloads)
Examples – Server Virtualization
Server Virtualization - Continued
Server Virtualization
Virtualized Servers as a Service (Amazon Web Services)

• Much as collocated servers, are available from a hosting provider, virtual servers are also available.

• Model is:
  • You pay for what you use.
  • Flexibility, need fewer servers today then you used, yesterday.
  • Leverage other amazon tools (storage/map-reduce/load-balancing/payments etc)
AWS
AWS Steps

- Select availability zone
- Launch new instance
- Select appropriate ami
- Associate with ssh key
- Launch instance
- Add ip
- SSH into new machine instance.
- t1-micro-instances run $54 a year + bandwidth
Try it for free...

- Free tier for the first Calendar year is (per month):
  - 750 hours of EC2 running Linux/Unix Micro instance usage
  - 750 hours of Elastic Load Balancing plus 15 GB data processing
  - 10 GB of Amazon Elastic Block Storage (EBS) plus 1 million IOs, 1 GB snapshot storage, 10,000 snapshot Get Requests and 1,000 snapshot Put Requests
  - 15 GB of bandwidth in and 15 GB of bandwidth out aggregated across all AWS services
- Which is not to say that, at scale EC2 is particularly cheap, (It isn't)
  - Limited capital at risk is in the context of prototyping or experimentation however.
AWS - Continued

• For provisioning purposes cli interaction is possible:
  • http://aws.amazon.com/developertools/351

• Along with tools to support the provisioning and destruction of virtual machines.
Provisioning and management

- Is the glue that makes virtualization usable
- In commercial virtualization environments the provisioning/management toolkits represent the bulk of the licensing cost (VMware) and the secret sauce (VMotion, disaster recovery, backup, etc)

Examples:

- XEN tools – a collection of perl scripts for spinning VMs
- Rightscale – (orchestration multiple public/private clouds)
  [http://www.rightscale.com](http://www.rightscale.com)
- Puppet (host / configuration management) -
Variation In virtualized environments

- Enterprise and Government virtualized environments may tend towards heterogeneity.
  - e.g. the applications (servers) that are being virtualized have accumulated over time
  - Are different enough that management may be depressingly manual
- ASP/Internet services environments may be more homogenous.
  - Leverage a common set infrastructure primitives
  - Thousands of like-systems providing overlapping functionality across hundreds of servers
  - Traditional network elements (e.g. loadbalancers/firewalls) may be virtualized along with the application.
Can you spot the...

- Web-node?
- Database-node?
- Load-balancer?
- Nameserver?
- DHCP Server?
- Email cluster?
- Devnodes?
Complimentary technologies

- NIC teaming or Link aggregation
- Network attached storage and network centric filesystems
  - NFS
  - Hadoopfs
  - GFS2
- Distributed databases
  - Example mysql cluster
  - Couchbase/Membase
  - OracleRAC