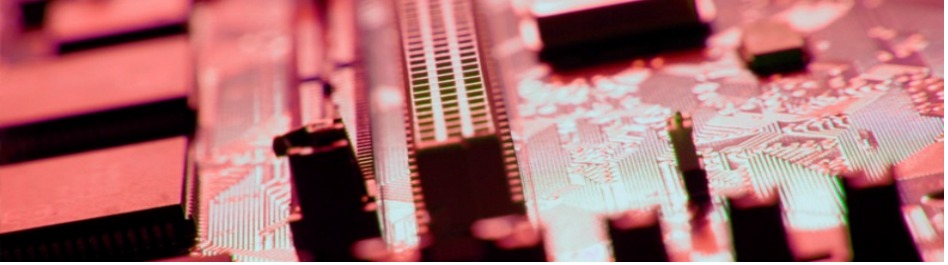

LCI workshop
on high performance clustered computing

Linux Clusters Institute: Cluster Stack Basics

Brett Zimmerman, University of Oklahoma
Senior Systems Analyst, OU Supercomputing Center for Education and Research (OSCER)

A Bunch of Computers

- Users can login to any node
- Filesystems aren't shared between nodes
- Work is run wherever you can find space
- Nodes maintained individually

What's wrong with a bunch of nodes?

- Competition for resources
  - Size and type of problem is limited
- Nodes get out of sync
  - Problems for users
  - Difficulty in management

Cluster Approach

- Shared filesystems
- Job management
- Nodes dedicated to compute
- Consistent environment
- Interconnect

What's right about the cluster approach?

- Easier to use
- Maximize efficiency
- Can do bigger and better problems
- Nodes can be used cooperatively

The Types of Nodes

- Login
  - Users login here
  - Compiling
  - Editing
  - Submitting and Monitoring jobs
- Compute
  - Users *might* login here
  - Run jobs as directed by the scheduler
- Support
  - Users *don't* login here
  - Do all the other stuff

What a cluster needs – the mundane

- Network services – NTP, DNS, DHCP
- Shared Storage -- NFS
- Logging – Consolidated Syslog as a starting point
- Licensing – FlexLM and the like
- Database – User and Administrative Data
- Boot/Provisioning – PXE, build system
- Authentication – LDAP

What a cluster needs -- Specialized

- Interconnect – An ideally low-latency network
- Job manager – Resource manager/ scheduler
- Parallel Storage – Get around the limitations of NFS

Network Services

- NTP – Network Time Protocol, provides clock synchronization across all nodes in the cluster
- DHCP – Dynamic Host Configuration Protocol, allows central configuration of host networking
- DNS – Provides name to address translation for the cluster
- NFS – Basic UNIX network filesystem

Logging

- Syslog
  - The classic system for UNIX logging
  - Application has to opt to emit messages
- Monitoring
  - Active monitoring to catch conditions elective monitoring doesn't catch
    - Resource manager
    - Nagios/cacti/zabbix/ganglia
- IDS
  - Intrusion detection
  - Monitoring targeting misuse/attacks on the cluster

Basic services, continued

- Licensing – FlexNet/FlexLM or equivalent, mediates access to a pool of shared licenses.
- Database – Administrative use for logging/monitoring, dynamic configuration. Requirements of user software.
- Boot/Provisioning – For example PXE/Cobbler, PXE/Image or part of a cluster management suite

Authentication

- Flat files -- passwd, group, shadow entries
- NIS -- network access to central flat files
- LDAP -- Read/Write access to a dynamic tree structure of account and other information
- Host equivalency

Cluster Networking

- Hardware Management – Lights out management
- External – Public interfaces to the cluster
- Internal – General node to node communication
- Storage – Access to network filesystems
- Interconnect – high-speed, low-latency for multi-node jobs

Some of these can share a medium

Interconnect

In the most recent Top 500 list (http://top500.org) there were 224 installations relying on Infiniband, 100 using Gigabit Ethernet, and 88 using 10 Gigabit Ethernet

- Ethernet – Latency of 50-125 µs (GbE), 5-50 µs (10GbE), ~5 µs RoCEE

- Infiniband – Latency of 1.3 µs (QDR) .7 µs (FDR-10/ FDR), .5 µs (EDR)

Parallel Filesystem

Lustre - http://lustre.org/

PanFS - http://www.panasas.com/

GPFS - http://www-03.ibm.com/software/products/en/software

Parallel filesystems take the general approach of separating filesystem metadata from the storage. Lustre and PanFS have dedicated nodes for metadata (MDS or director blades). GPFS distributes metadata throughout the cluster

Cluster Management

- Automates the building of a cluster
- Some way to easily maintain cluster system consistency
- The ability to automate cluster maintenance tasks
- Offer some way to monitor cluster health and performance

Cluster Managemement Software

The resource manager knows the state of the various resources on the cluster and maintains a list of the jobs that are requesting resources

The scheduler, using the information from the resource manager selects jobs from the queue for execution

- Rocks (http://www.rocksclusters.org/wordpress/)
- Bright Cluster Manager (http://www.brightcomputing.com/Bright-Cluster-Manager)
- xCAT (Extreme Cluster/Cloud Administration Toolkit) (http://sourceforge.net/p/xcat/wiki/Main_Page/

Configuration Management

While it is true that booting with a central boot server can make it easier to make sure the OS on each compute node (or, at least, each type of compute node) has an identical setup/install, there are still files which wind up being more dynamic. Some such files are password/group/shadow and hosts files.

- Rsync
- Cfengine
- Chef
- Puppet
- Salt

Software Installation and Management

All linux distros have some sort of package management tool. For Redhat/CentOS/Scientific based clusters, this is rpm and yum. Debian has dpkg and apt

In any case pre-packaged software tends to assume that it is going to be installed in a specific place on the machine and that it will be the only version of that software on the machine.

One a cluster, it may be necessary to look at software installation differently from a standard linux machine

- Install to global filesystem
- Keep boot image as small as possible
- Maintain multiple versions

Software installation and management

There are a couple of tools useful for navigating the difficulties of maintaining user environments when dealing with multiple versions of software or software in non-standard locations.

- SoftEnv (http://http://www.lcrc.anl.gov/info/Software/Softenv)

  Useful for packaging static user environment required by packages

- Modules (http://modules.sourceforge.net/)

  Can be used to make dynamic changes to a users environment.

Resource Manager/Scheduler

- Accepts job submissions, maintains a queue of jobs
- Allocates nodes/resources and starts jobs on compute nodes
- Schedules waiting jobs
- Available options
  - SGE (Sun Grid Engine)
  - LSF / Openlava (Load Sharing Facility)
  - PBS (Portable Batch System)
    - OpenPBS
    - Torque
  - SLURM

Best Practices

Here is a quick overview of the general functions to secure a cluster

- Risk Avoidance
- Deterrence
- Prevention
- Detection
- Recovery

The priority of these will depend on your security approach

Risk Avoidance

- Provide the minimum of services necessary
- Grant the least privileges necessary
- Install the minimum software necessary

The simpler the environment, the fewer the vectors available for attack.

Deterrence

- Limit the discoverability of the cluster
- Publish acceptable use policies

Prevention

- Fix known issues (patching)
- Configure services for minimal functionality
- Restrict user access and authority
- Document actions and changes

Detection

- Monitor the cluster
- Integrate feedback from the users
- Set alerts and automated response

Recovery

- Backups
- Documentation
- Define acceptable loss