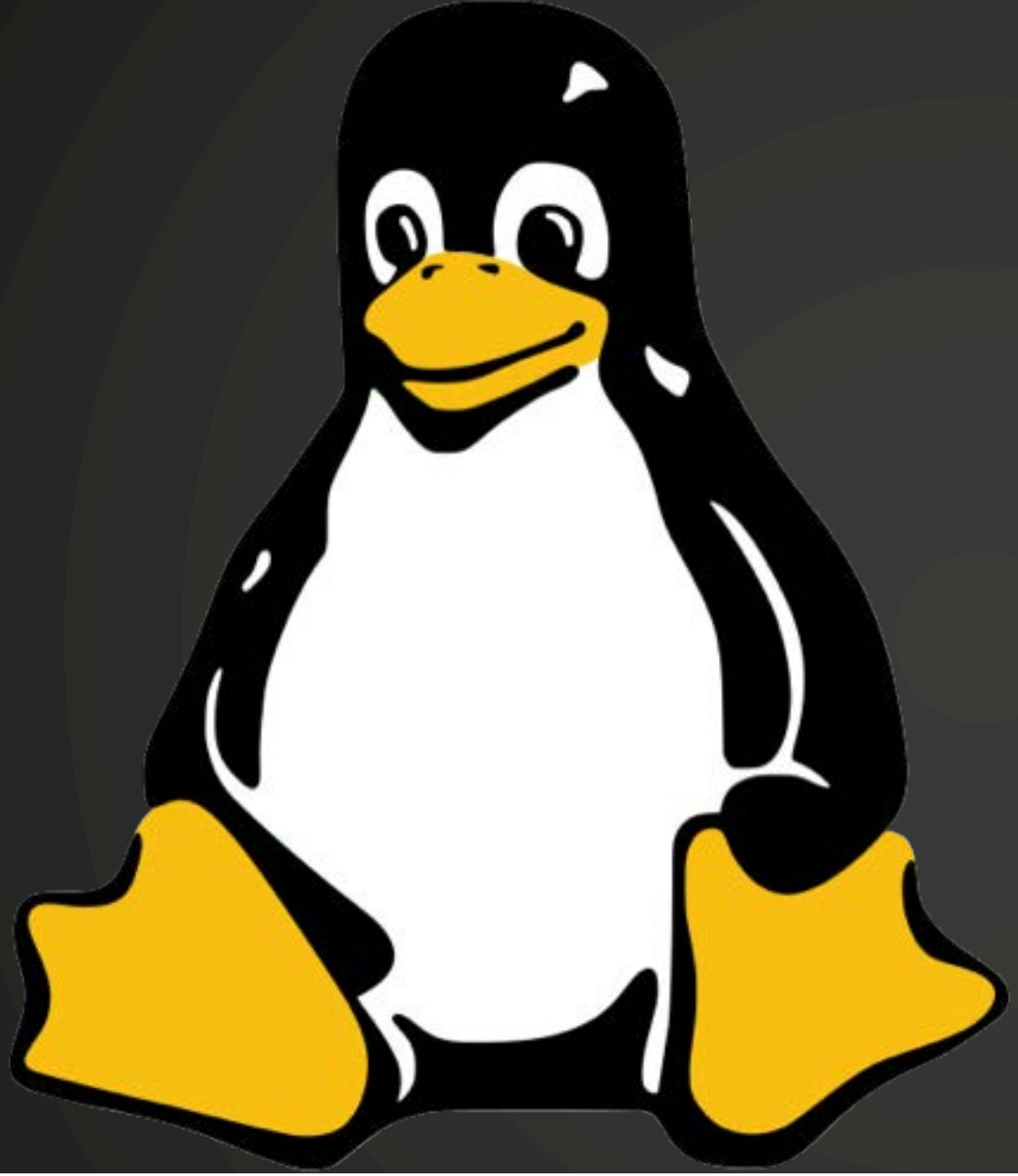
Los Alamos

NATIONAL LABORATORY

EST. 1943



Los Alamos
NATIONAL LABORATORY
EST. 1943

Delivering science and technology
to protect our nation
and promote world stability

CentOS Linux

Packages, configuration & installation

Presented by CSCNSI

Outline

1. What is CentOS?
2. Installing CentOS

What is CentOS?

- CentOS is a popular Linux distribution (https://centos.org)
- CentOS is a free, open source rebuild of (most of) the Red Hat Linux distribution (which is not free)
  - CentOS is mostly “binary compatible” with Red Hat (i.e. if it runs on Red Hat, it will probably run on CentOS without modification)
- CentOS use RPM software packages and Yum repositories of RPMs, just like Red Hat
- CentOS (as of v7) uses systemd to initialize, manage services, etc.
- There are tens of thousands of pre-built software packages for CentOS
- *We’re going to use CentOS 7 for our clusters*

The CentOS installer

- CentOS provides a graphical installer tool
- Important steps once booted into the installer:
  - Set basic configuration (e.g. timezone)
  - Setup the installation destination
    - Partition a disk (possibly with LVM, software RAID, etc)
    - Setup filesystem mount points (must have ”biosboot”, “/” and “/boot” at minimum)
  - Configure network settings
  - Select the packages you want to install
    - Often it’s easiest to just select “minimal” and install more later
  - Click “Begin Installation”… and wait.

RPMs package format

- Originally developed by/for Red Hat, now used by many distributions
- Consists of several parts:
  - Header—information on what dependencies the package needs, who wrote it, etc.
  - Pre/post scripts—scripts that need to be run before and after the package has installed (for, e.g. additional setup)
  - An archive of the files in the package (typically a compressed cpio)
  - Crypto signature to verify integrity and trust
- Packages are installed with the “rpm” command, and information about what is installed is kept in a local database

```
Name: eject
Version: 2.1.5
Release: 1%{?dist}
Summary: A program that ejects removable media using software control
License: GPLv2+
URL: http://www.pobox.com/~tranter
Source0: http://www.ibiblio.org/pub/Linux/utils/disk-management/%{name}-%{version}.tar.gz
BuildRequires: gettext
BuildRequires: libtool

%description
The eject program allows the user to eject removable media (typically...

%prep
%setup -q -n

%build
%configure
make %{?_smp_mflags}

%check
make check

%install
rm -rf $RPM_BUILD_ROOT
make install DESTDIR=$RPM_BUILD_ROOT

install -m 755 -d $RPM_BUILD_ROOT/%{_sbindir}
ln -s ../bin/eject $RPM_BUILD_ROOT/%{_sbindir}

%find_lang %{name}

%files -f %{name}.lang
%doc README TODO COPYING ChangeLog
%{_bindir}/*
%{_sbindir}/*
%{_mandir}/man1/*

%changelog
* Wed Oct 20 2011 John Doe <jdoe@example.com> 0.8.18.1-0.1
- Initial RPM release
```

Example SPEC file. Spec files are used to describe and build RPMs.

Yum repositories

- RPMs provide a nice way to:
  - Bundle packages
  - Install/remove packages…
- They don't create an easy way to *get* packages
- Yum is a tool to:
  - Query *repositories* of packages (possibly remote)
  - Install packages or groups of packages
  - Resolve and auto-install needed dependencies for packages
  - Manage updating of software over time
- A system can have many repositories defined that it can query/install from
- It's fairly simple to make your own local mirrors of repositories

```
13 [base]
14 name=CentOS-$releasever - Base
15 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=os&infra=$infra
16 #baseurl=http://mirror.centos.org/centos/$releasever/os/$basearch/
17 gpgcheck=1
18 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
19
20 #released updates
21 [updates]
22 name=CentOS-$releasever - Updates
23 mirrorlist=http://mirrorlist.centos.org/?release=$releasever&arch=$basearch&repo=updates&infra=$infra
24 #baseurl=http://mirror.centos.org/centos/$releasever/updates/$basearch/
25 gpgcheck=1
26 gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
```

Example repository specification.

```
[root@localhost ~]# yum list available kube*
Loaded plugins: fastestmirror
Loading mirror speeds from cached hostfile
 * base: centos.mirror.lstn.net
 * extras: mirror.hostduplex.com
 * updates: linux.mirrors.es.net
Available Packages
kubernetes.x86_64                    1.5.2-0.7.git269f928.el7    extras
kubernetes-client.x86_64             1.5.2-0.7.git269f928.el7    extras
kubernetes-master.x86_64             1.5.2-0.7.git269f928.el7    extras
kubernetes-node.x86_64               1.5.2-0.7.git269f928.el7    extras
```

Listing available packages with yum.

Why choose CentOS?

- Widely supported
- Huge amount of pre-built software available
- Binary compatible with many commercial software packages (e.g. Matlab)
- Common; likely that new people will already know it
- Commonly used for HPC, so many HPC tools are pre-built for it

Installing CentOS

Preparing the system for install

1. The Master node should already have the following setup
   - Blue (Cluster LAN) ethernet cable in em1 port
   - Orange (Uplink) ethernet cable in em2 port
   - Infiniband cable installed
   - Power cable installed
   - Two drives installed
2. Additionally, make sure KVM cable is plugged in to USB/Video and Port 1 on the KVM switch
3. Pull out the Console and power it on. Make sure Port 1 is selected
4. Put the provided USB installer key in the front USB port

Installation overview

1. Configure RAID array
2. Insert USB installer
3. Start CentOS installer (select from boot manager with <F11>)
   - Set timezone
   - Format/configure destination filesystems
   - Add “admin” user (we’ll make individual users later)
   - Setup network configuration
   - *Begin Installation*
4. Login with the “admin” user to verify installation

RAID Configuration

What is RAID?

- RAID = “Redundant Array of Inexpensive Disks”
- RAID is a technology that can make multiple disks work together and appear to be a single disk (called a Virtual Disk, or VD)
- Advantages of RAID include:
  - Increase fault tolerance (e.g. allow one or more disks to fail without total VD failure)
  - Increase performance by spreading disk operations across many disks

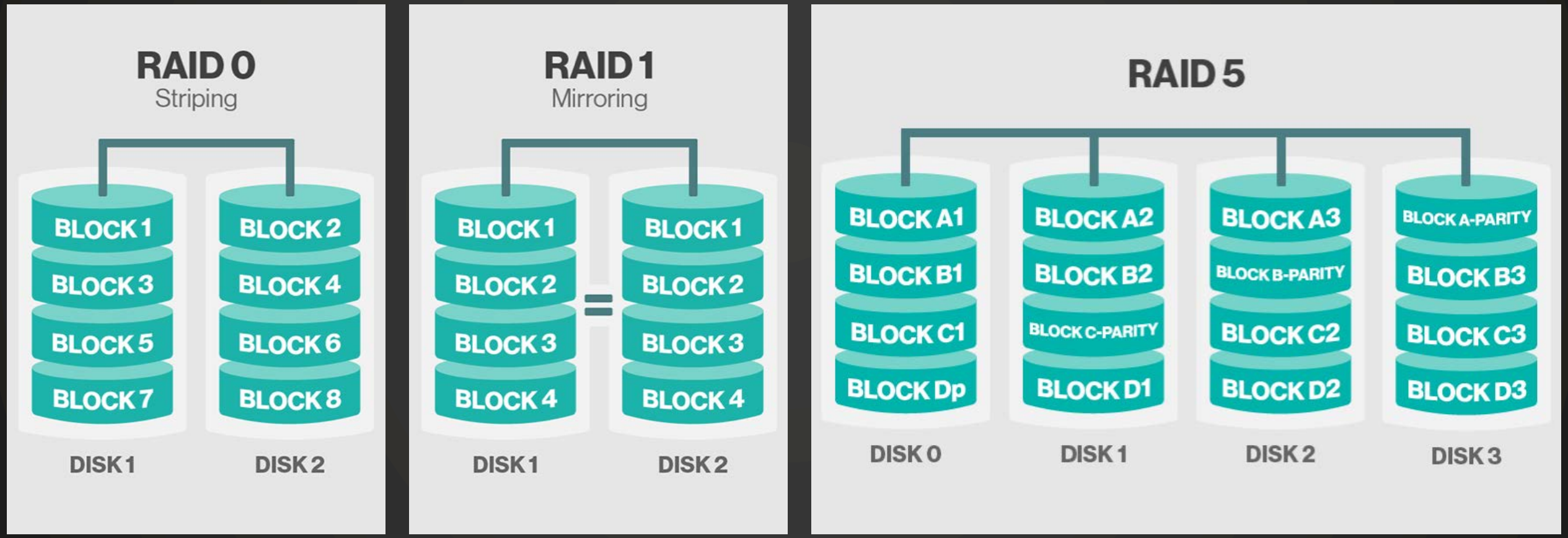
Some RAID terminology

- Striping:
  - Split data across many disks (can give performance gain)
- Mirroring:
  - Make one disk (or VD) a full clone of another disk (or VD)
- Parity:
  - Checksum for row of data
  - Used to check for/correct corrupt data
  - Can reconstruct missing data from a failed disk using the checksum + remaining data
- Overhead:
  - Amount of usable space lost by RAID configuration.

Common RAID levels

- Different RAID configurations are called “levels”. Here are a few common ones:
- RAID0—striping across 2 or more disks; no parity.
  - Pro: striping can increase speed; no usable capacity loss.
  - Con: cannot be recovered if a disk fails; more likely to fail than a single disk.
- RAID1–mirroring across two disks.
  - Pro: complete backup of a disk with rapid recovery.
  - Con: overhead is ½ total space.
- RAID5—striping + single parity across a pull of disks.
  - Pro: striping can increase speed; can recover from a single drive failure.
  - Con: overhead is 1/N total space (where N is drive count). Requires 3 or more drives.
- Other configs include: RAID6 (two parity drives). RAID10, RAID50, RAID60.

RAID Levels (diagram)



http://searchstorage.techtarget.com/definition/RAID

Hardware vs Software RAID

- RAID works by inserting an interface between the disks and the block layer
- There are two ways to do this:
  - Hardware RAID—the disks are physically connected to a special controller. RAID algorithms are implemented in hardware/firmware on the control. The controller usually has a configuration utility and can present VD's as block devices to the operating system.
  - Software RAID—there is no physical controller. Linux (and others) support performing RAID calculations in software on existing block devices, then present the result as a new block device (usually /dev/md*)

Our RAID

- Your server has a Dell PERC H330 Hardware RAID controller we will used to configure the two disks into a RAID1 (mirrored) pair.
- We choose RAID1 because it provides simple, reliable redundancy. It is common to use RAID1 for disks that the OS will be installed on.
- Steps to get to RAID configuration:
  - Power on the system
  - It will take 2-3 minutes to complete the POST process
  - Before the system boots, enter the RAID controller config (Ctrl-R, when prompted)
  - This will drop you into the RAID configuration tool
- Details are in a separate guide.

Install Configuration Options

System Disk Partitions

- We often split the disk into multiple partitions
- Reasons:
  - DoS attack or accidental failure by filling system drive
  - Partition encryption (i.e. /home)
  - Special purpose partitions (swap, /boot, biosboot)
  - Easier backup/recovery
- The defaults will work, but adjust if your team wants

System Disk Partitions

/	root filesystem (default)
biosboot	special partition for GPT
swap	memory swap (default)
/usr; /usr/local	user executables; kernel source; docs
/var	spool; caches; system logs
/tmp	temporary
/boot	kernel; bootloader
/home	user home directories
/opt	locally compiled executables; configs; docs

System Disk Partitions

/	> 8GB
biosboot	always 1MB
swap	> 1.5 x (RAM) if hibernating, less otherwise
/usr	> 10GB
/var	> 2GB
/tmp	> 250MB
/boot	> 250MB -1GB
/home	> (user count) x (allotment)
/opt	> 5GB on a real system

System Disk Partitions Recommendations

biosboot	1MB
swap	16GB
boot	500MB
/	fill remaining space ('grow')

Other installation configuration

- Network settings will be provided to you for the uplink port
- Add a user named “admin”, make sure this user is selected as an administrator
- Set the timezone to Denver
- Use the “Minimal” install set (it’s easier to install more software later)
- Use the XFS filesystem type
- You can use your team name for your hostname
- *Installation details will be provided in a separate guide*

Questions?