www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

A Quick Start Guide to Installing Ceph

This seems to be the age of "Big Data". Every sector out there seems to have a need for it. From biotechs doing genome sequencing to financial providers mining market data, the ability to store massive amounts of structured or unstructured data is the key to their success. Accessing that data quickly is just as important. Traditionally, centralized storage was the go-to solution. You invested in an expensive Storage Area Network, and in return, it provided excellent performance and scalability.

From a small business perspective, a traditional SAN presents several challenges. 1. Cost – A SAN is typically composed of a storage controller, some disk shelves, and a separate fibre channel network. Then you have some not-so obvious costs like SAN software & licensing (management software, replication software), and HBA costs. 2. Administrative overhead – Ethernet switching and routing is ubiquitous. Fiber channel on the other hand, requires experience with FC switches, zoning, multipathing, etc. You'd be best suited to hiring a dedicated storage administrator. 3. Scaling with respect to cost – You invest in the SAN equipment, increase your compute capacity, purchase more SAN equipment, rinse, repeat. As you grow your SAN, how do you plan for upgrades? How do you justify eventual forklifts?

Enter the new era – distributed filesystems. Ok, perhaps this isn't so new. Google developed their own in house proprietary filesystem years ago, called BigFiles. It was designed to run on commodity servers, be resilient (since it runs on commodity servers), and perform well. No HBAs, no separate fiber infrastructure, no costly SAN. The idea is that as one unit of computing is added, you get an additional spindle or two of IO – so performance is linear as you scale.

Several open source distributed filesystems have gained in popularity recently. One of which I'll be discussing today is Ceph. Developed as a drop-in replacement for Hadoop's distributed filesystem, I'll show you how you can quickly deploy it to serve as your primary storage. And even if you're not sequencing any genomes, you likely either have: A VMware cluster, an Exchange installation, or users who simply like to store lots of files. Any of these situations can reap the benefits.

…

Obtain the packages

You can grab the latest source from Ceph.com directly. The latest release is named "Dumpling". However, depending on your distribution, you may run into a dependency nightmare trying to build it. Pre-built packages exist for the most common distributions.
On my Ubuntu 11.10 servers, I installed these packages:
apt-get install ceph ceph-client-tools gceph libceph1 librados2 librbd1 python-ceph
Architecture

Ceph distributes its data amongst storage daemons (OSD). Typically one OSD per disk is required. Ceph maintains a master copy of the cluster map on Monitor servers.
There are several interfaces into the Ceph storage cluster:

www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

1. POSIX client. This allows you the ability to mount a filesystem that you can 'ls', 'cd', etc into. The client can run in userspace via FUSE, or as a kernel module.
2. Block storage client. This allows you to carve out a LUN, which you would then format as the filesystem of your choice, and then mount.
3. Object Gateway. This mimics the Amazon S3 service. Using an API, you can 'GET' and 'PUT' files via HTTP.

For my example, I will be describing the POSIX filesystem. The POSIX filesytem requires yet another daemon to keep track of the additional metadata associated to file permissions, inodes, etc. This is called the MDS.
So at minimum, for our example, we will build two OSDs, one MDS, and one MON.

Setup SSH Access

The monitor (MON) or admin server requires SSH access to all other servers.

- Create a global "ceph" user. If you're using NIS this should be trivial. Otherwise, create a "ceph" user on each machine.
- As ceph on the admin server, generate an SSH key

```
ceph@adminhost1:~$  ssh-keygen -t  dsa
```

- Copy the resulting public key to the other hosts

```
ceph@adminhost1:~$  scp ~/.ssh/id_dsa.pub OSDSERVER1:~/
ceph@adminhost1:~$  scp ~/.ssh/id_dsa.pub OSDSERVER2:~/
ceph@adminhost1:~$  scp ~/.ssh/id_dsa.pub MDSSERVER1:~/
```

- Add the public key to each machine's authorized_keys file

```
ceph@OSDSERVER1:~$  cat id_dsa.pub >> ~/.ssh/authorized_keys
ceph@OSDSERVER2:~$  cat id_dsa.pub >> ~/.ssh/authorized_keys
ceph@MONSERVER1:~$  cat id_dsa.pub >> ~/.ssh/authorized_keys
```

Setup the main configuration file

The main configuration file (ceph.conf) defines all OSDs, Monitors, and MDS servers. Once you populate it, distribute it to all machines. Alternatively, you could keep this on an NFS share.

```
[global]
    auth supported = cephx
    keyring = /mnt/home2/ceph/keyring.admin

[osd]
    osd data = /home/ceph/osd$id
    osd journal = /home/ceph/osd$id/journal
    osd journal size = 512
    keyring = /mnt/home2/ceph/keyring.$name

    filestore xattr use omap = true
    filestore filemap = false

[osd.11]
    host = OSDSERVER1
    cluster addr = 10.141.1.150:6800
    public addr = 10.141.1.150:6801
```

www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

```
[osd.21]
    host = OSDSERVER2
    cluster addr = 10.141.1.22:6800
    public addr = 10.141.1.22:6801

[mon]
    mon data = /home/ceph/mon$id
[mon.1]
    host = MONSERVER1
    mon addr = 10.141.0.181:6789

[mds]
    keyring = /mnt/home2/ceph/keyring.$name
[mds.1]
    host = MDSSERVER1
```

Create the cluster

Finally, create each directory on each server. For instance, the config file specifies that osd.11 will run on OSDSERVER1. Per the global [osd] section, the data should go to into /home/ceph/osd11. On the Monitor server, /home/ceph/mon1 should be created.

You're now ready to create the cluster:

```
$  mkscephfs -a -c  ~/ceph.conf -k ~/keyring.admin
```

The mkcephfs command will create a client.admin key and store it in the keyring.admin file that we specified. When you run admin commands, Ceph will use this key to authenticate with the other daemons.

Start up the daemon on the admin / Monitor server:

```
$  /etc/init.d/ceph -a start
```

What this will trigger is an ssh to all OSDs where their services will be started as well.

Check that your cluster is running, with a healthy status:

```
ceph@monserver1:/home/ceph$ ceph health
2013-10-25 11:04:59.367696 mon <- [health]
2013-10-25 11:04:59.368108 mon0 -> 'HEALTH_OK' (0)
```

Mount the cluster file system

Create a mountpoint:

```
$  mkdir /mnt/ceph
```

Find the correct key for the admin user:

www.spkaa.com
Ph: 888-310-4540

SPK and Associates
900 E Hamilton Ave, Ste.100
Campbell, CA 95008

```
$  ceph auth list
client.admin
        key: AQD8xGpS+JzUGBAAxHDolz1x+iUERqeethUFAw==
        caps: [mds] allow
        caps: [mon] allow *
```

Use this key as a mount parameter:

```
$  mount -t ceph ip.address.of.MON:6789:/ /mnt/ceph -o
name=admin,secret= AQD8xGpS+JzUGBAAxHDolz1x+iUERqeethUFAw==
```

And you're all set! You will want to benchmark your system to identify any network bottlenecks, as well as to identify the best stripe size and stripe count settings for your workload.

Mike Solinap
Sr. Systems Integrator
SPK & Associates