Requirements

- Two or more filesystems with more than 2TB of disk space each (ideally on different nodes or shared volumes/pools/filesystems). They can be: xfs, ext4, cephfs, or mountable volumes/pools (e.g., ceph, dcache)
- Ability to run xrootd/eos daemons on the nodes holding the filesystems/volumes (see installation instructions below)
- Preferably 10GE NICs
- A stable, DNS-registered hostname. Ideally, the DNS hostname should match the Linux hostname and resolve to the IP address set on the machine interface. Network Address Translation is supported as well
- Open port for xrootd (TCP - 1095) for messaging and real data traffic
- Possibility to log in to the servers remotely (tuning, debug, etc.)

Installation via Docker container

The installation via Docker container consists of a Docker image where the EOS file server daemon lives.

1. Install Docker and Docker compose

Refer to the official installation instructions for your OS: https://docs.docker.com/install/🖉 and https://docs.docker.com/compose/install/🖉

The deployment has been tested and proved working with Docker 17.03 and Docker Compose 1.19.

2. Prepare the filesystem

Assuming to have a block device available at `/dev/md1`, format it as an xfs filesystem and mount it on the host at `/mnt/eosdisk`.

```
mkfs.xfs -L "eos_fst" /dev/md1
mkdir -p /mnt/eosdisk
mount /dev/md1 /mnt/eosdisk
```

It is recommended to edit `/etc/fstab` consistently to make the mount persistent in case of host reboot.

3. Create a configuration file for the container

Copy and paste what follows in a plain-text file called `eos-fst.yaml`

```
# Docker Compose configuration file for EOS FST container
version: "2"

services:
  eos-fst:
    image: gitlab-registry.cern.ch/cernbox/boxedhub/eos-datalakes:alpha
    container_name: eos-fst
    restart: unless-stopped
    cap_add:
      - SYS_PTRACE
    security_opt:
      - apparmor=unconfined
    network_mode: "host"
    volumes:
      - %%%storage-path-on-the-host%%%:/mnt/data01
    environment:
      EOS_INSTANCE_NAME: "eoseulake"
      EOS_GEOTAG: "%%%your-geographical-location%%%"
```

```
EOS_MGM_MASTER1: "eulake.cern.ch"
EOS_MGM_MASTER2: "eulake.cern.ch"
EOS_MGM_ALIAS: "eulake.cern.ch"
EOS_MQ_ALIAS: "eulake.cern.ch"
FST_MOUNTPOINT: "/mnt/data01"
FST_FQDN: "%%%your-machine-hostname%%%"
```

4. Customize the configuration file `eos-fst.yaml` with the specificities of your site

Edit `eos-fst.yaml` and replace

- *%%%storage-path-on-the-host%%%* with the path on the host where the filesystem for EOS is mounted (e.g., */mnt/eosdisk*)
- *%%%your-geographical-location%%%* with an identifier of the location (e.g., *cern*). To support multi-level geoscheduling, specify the location with finer granularity (e.g., *cern::gva::rack1*)
- *%%%your-machine-hostname%%%* with the Fully Qualified Domain Name of the host where the xrootd/eos daemons run (e.g., *eos-fst.cern.ch*)

5. Start the container

Run (with sudo privileges)

```
docker-compose -f eos-fst.yaml up -d
```

The Docker daemon will pull the image from the repository and start the container. The EOS file server daemon will automatically connect to the management node and register the file system.

Known Limitations

- The Docker container supports a single volume where the EOS file storage daemon can read/write (*/mnt/eosdisk* in the example). This will be extended in future.
- The EOS logs are not stored persistently. Restarting the container will cause to old logs to be wiped.

Installation with native eos client (rpm-based)

work-in-progress

-- EnricoBocchi - 2018-05-23

This topic: WLCGDatalakes > FSTDeployment
Topic revision: r2 - 2018-05-24 - EnricoBocchi