

Sharing server exchange directory

[Go back to Pandora FMS documentation index](#)

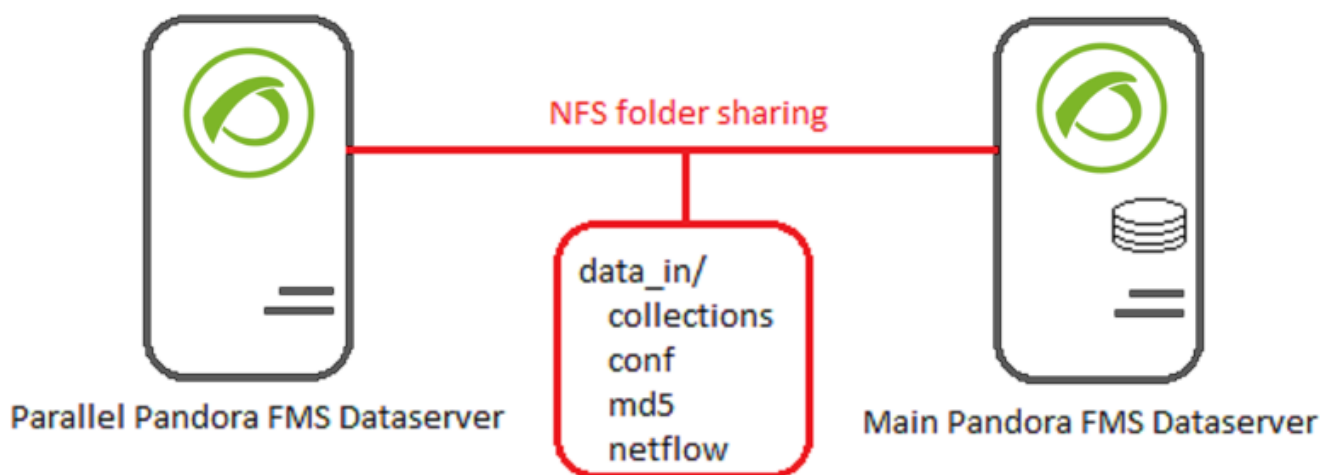
Introduction

Pandora FMS dataserver uses the `/var/spool/pandora/data_in` directory, and all its contents to manage the information that receives and send to the software agents.

That directory also needs to be accessible by the console of Pandora, so the instructions that it sends to the agents can reach them, being config files or collections.

If we have several servers with servers with several consoles, the default configuration, every console will be able to manage the agents of the server where it is located.

Now, let's suppose that we have several Pandora servers working in a common environment.



Each of the agents that each server manages will communicate with their assigned dataserver using the `data_in` folder. On a multiple-dataserver architecture with a single console, unify agent management using NFS or GlusterFS to share this *pool* of common information.



Sharing the `pandora_console/attachment` folder between the different Consoles is also recommended as it makes collection management easier.

Which method should I use?

Although both NFS and GlusterFS are able to share the required files, they are best recommended for different environments:

- If data are stored in an *external server* to that of Pandora FMS, and it will work as its client, *NFS* may be used.
- If data are stored in *Pandora FMS servers* or fault tolerance (at the software level) is required, we recommend **GlusterFS**.



It's mandatory to share *data_in*'s *conf*, *md5*, *collections* and *netflow* folders for HA environments, and we recommend to share the *pandora_console/attachment* folder as well. The *data_in* folder itself **must not be shared**, unless [Tentacle server is configured for concurrent access](#) to XML files.

NFS configuration

First steps

Install the *nfs-utils* package on **all the systems that will share the directory** by NFS:

```
yum install -y nfs-utils
```

Configuration of the NFS server



It's very important for the NFS server to be a **separate server** from those of Pandora FMS. If one of them were configured as NFS server and there were any errors prevented the client from connecting, the shared files would not be accesible, **causing errors** in Pandora FMS. If it is not possible to use a separate server, *GlusterFS* should be used instead.

Edit the file */etc/export* adding the following:

```
/var/spool/pandora/data_in/conf  
[IP_CLIENTE](rw,sync,no_root_squash,no_all_squash)  
/var/spool/pandora/data_in/collections  
[IP_CLIENTE](rw,sync,no_root_squash,no_all_squash)  
/var/spool/pandora/data_in/md5
```

```
[IP_CLIENTE](rw, sync, no_root_squash, no_all_squash)
/var/spool/pandora/data_in/netflow
[IP_CLIENTE](rw, sync, no_root_squash, no_all_squash)
/var/www/html/pandora_console/attachment
[IP_CLIENTE](rw, sync, no_root_squash, no_all_squash)
```

Where *[client_ip]* stands for the IP address of the system with which the resource is going to be shared. For example:

```
/var/spool/pandora/data_in/conf
192.168.70.10(rw, sync, no_root_squash, no_all_squash)
/var/spool/pandora/data_in/collections
192.168.70.10(rw, sync, no_root_squash, no_all_squash)
/var/spool/pandora/data_in/md5
192.168.70.10(rw, sync, no_root_squash, no_all_squash)
/var/spool/pandora/data_in/netflow
192.168.70.10(rw, sync, no_root_squash, no_all_squash)
/var/www/html/pandora_console/attachment
192.168.70.10(rw, sync, no_root_squash, no_all_squash)
```

In case that we have the firewall enabled in our system, open the required ports:

```
# CentOS
firewall-cmd --permanent --zone = public --add-service = nfs
firewall-cmd --reload
```

Once done, we start the services:

```
# CentOS
service rpcbind start
service nfs-server start
service nfs-lock start
service nfs-idmap start
```

Configure NFS to start when the system powers on:

```
chkconfig rpcbind on
chkconfig nfs-server on
chkconfig nfs-lock on
chkconfig nfs-idmap on
```

To refresh any change in the setup of the `/etc/export` restart `nfs-server`

```
service nfs-server restart
```

Configuration of the NFS clients

First, back up the directory:

```
mv /var/spool/pandora/data_in /var/spool/pandora/data_in_locale
```

Note If that system does not have apache installed (is not necessary to install it), add to `/etc/passwd` and `/etc/group` the user apache to avoid permission conflicts:

```
echo "apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin" >> /etc/passwd
echo "apache:x:48:" >> /etc/group
```

Check the folder permissions:

```
chown pandora:apache /var/spool/pandora/data_in
chmod 770 /var/spool/pandora/data_in
```

Check that we can mount successfully the remote folder:

```
mount -t nfs [IP_SERVER_IP]:/var/spool/pandora/data_in/conf
/var/spool/pandora/data_in/conf
mount -t nfs [IP_SERVER_IP]:/var/spool/pandora/data_in/md5
/var/spool/pandora/data_in/md5
mount -t nfs [IP_SERVER_IP]:/var/spool/pandora/data_in/collections
/var/spool/pandora/data_in/collections
mount -t nfs [IP_SERVER_IP]:/var/spool/pandora/data_in/netflow
/var/spool/pandora/data_in/netflow
```

Where `[NFS_SERVER_IP]` stands for the IP address of the server that provides the NFS service. For example:

```
mount -t nfs 192.168.70.10:/var/spool/pandora/data_in/conf
/var/spool/pandora/data_in/conf
mount -t nfs 192.168.70.10:/var/spool/pandora/data_in/md5
/var/spool/pandora/data_in/md5
mount -t nfs 192.168.70.10:/var/spool/pandora/data_in/collections
/var/spool/pandora/data_in/collections
mount -t nfs 192.168.70.10:/var/spool/pandora/data_in/netflow
/var/spool/pandora/data_in/netflow
```

If the previous command fails, check:

- Firewall status.
- If you are running as root.
- If the directory where you want to build it exists.

If everything is right up to here, configure the system to be built automatically if there is a reboot, editing the file `/etc/fstab`:

```
# Add the following lines to the configuration file /etc/fstab
```

```
[NFS_SERVER_IP]:/var/spool/pandora/data_in/conf
/var/spool/pandora/data_in/conf nfs defaults 0 0
[NFS_SERVER_IP]:/var/spool/pandora/data_in/md5
```

```
/var/spool/pandora/data_in/md5  nfs defaults 0 0
[NFS_SERVER_IP]:/var/spool/pandora/data_in/collections
/var/spool/pandora/data_in/collections  nfs defaults 0 0
[NFS_SERVER_IP]:/var/spool/pandora/data_in/netflow
/var/spool/pandora/data_in/netflow  nfs defaults 0 0
[NFS_SERVER_IP]:/var/www/html/pandora_console/attachment
/var/www/html/pandora_console/attachment  nfs defaults 0 0
```

Where [NFS_SERVER_IP] stands for the IP direction of the server that provides the NFS service.

GlusterFS configuration

GlusterFS allows to share Pandora FMS key directories between the servers and thus keep the data available if any of them becomes unreachable. Thanks to this system you will always have an active resource, and the data will be accessible even if not all servers are working.

Requirements

- **Selinux** must be disabled or configured with the proper rules.
- **Firewall** must be disabled or configured with the proper rules.
 - **Port 24009/tcp** must be open.
- The **/etc/hosts** file must be configured with all names and IP addresses in all servers.
- **Additional disks with no partitioning** must be created in all servers.

Package installation

To install GlusterFS search for the available versions:

```
yum search centos-release-gluster
```

Install the latest LTS stable version:

```
yum install centos-release-gluster37
yum install glusterfs glusterfs-cli glusterfs-libs glusterfs-fuse
```

Creating XFS partitions (bricks)



We will use *gluster1.example.com* and *gluster2.example.com* as sample servers for this guide.

Create a new physical volume using the `/dev/xvdb` disk:

```
pvccreate /dev/xvdb
```

```
Physical volume “/dev/xvdb” successfully created
```

Create a volume group in /dev/xvdb:

```
vgcreate vg_gluster /dev/xvdb  
Volume group “vg_gluster” successfully created
```

Create a volume brick1 for XFS bricks in both nodes of the cluster, setting the space to be assigned to them with the `-L` parameter:

```
lvcreate -L 5G -n brick1 vg_gluster  
Logical volume “brick1” created.
```

Alternatively you can set the space to be assigned as a disk percentage:

```
lvcreate -l 100%FREE -n brick1 vg_gluster
```

Configure the filesystem as XFS:

```
mkfs.xfs /dev/vg_gluster/brick1
```

Create the mount point and mount the XFS brick:

```
mkdir -p /glusterfs/brick1  
mount /dev/vg_gluster/brick1 /glusterfs/brick1
```

Open the `/etc/fstab` file where to add the following line:

```
/dev/vg_gluster/brick1 /glusterfs/brick1 xfs defaults 0 0
```

Enable and start `glusterfsd.service` in both nodes:

```
systemctl enable glusterd.service --now
```

From the first GlusterFS node, connect to the second and create the Trusted Pool (Storage Cluster):

```
gluster peer probe gluster2.example.com  
peer probe: success.
```

Verify the cluster peer:

```
gluster peer status  
Number of Peers: 1  
Hostname: gluster2.example.com  
Uuid: e528dc23-689c-4306-89cd-1d21a2153057  
State: Peer in Cluster (Connected)
```

Creating the HA volume

Then use the XFS partition `/glusterfs/brick1` in both nodes to create a **HA replicated volume**.

Create a subfolder in `/glusterfs/brick1` mount point. It is needed for GlusterFS to work.

```
mkdir /glusterfs/brick1/brick
```

Create a GlusterFS replicated volume:



Run this command in **just one of the nodes** (in the example, `gluster1.example.com`).

```
gluster volume create glustervol1 replica 2 transport tcp
gluster1.example.com:/glusterfs/brick1/brick \
gluster2.example.com:/glusterfs/brick1/brick
  volume create: glustervol1: success: please start the volume to access
data
```

```
gluster volume start glustervol1
  volume start: glustervol1: success
```

Verify the GlusterFS volumes:

```
gluster volume info all
  Volume Name: glustervol1
  Type: Replicate
  Volume ID: 6953a675-f966-4ae5-b458-e210ba8ae463
  Status: Started
  Number of Bricks: 1 x 2 = 2
  Transport-type: tcp
  Bricks:
  Brick1: gluster1.example.com:/glusterfs/brick1/brick
  Brick2: gluster2.example.com:/glusterfs/brick1/brick
  Options Reconfigured:
  performance.readdir-ahead: on
```

Mounting the volumes in clients

Install the client packages for GlusterFS:

```
yum install glusterfs glusterfs-fuse attr -y
```

Create a folder for Pandora FMS files:

```
mkdir /pandora_files/
```



The path `/pandora_files/` is only used as an example, and any other folder can be used.

Mount the GlusterFS volumes on the client:

```
mount -t glusterfs gluster1.example.com:/glustervol1 /pandora_files/
```

Add the following line to `/etc/fstab`:

```
gluster1.example.com:/glustervol1 /pandora_files glusterfs defaults,_netdev  
0 0
```

Once the partition has been mounted in `/pandora_files/`, proceed to create all the required directories in this folder:

```
cd /pandora_files/  
mkdir collections md5 conf netflow attachment
```

Copy all these directories from to the original folder in `/var/spool/pandora/data_in`:

```
cp -rp /var/spool/pandora/data_in/conf /pandora_files/  
cp -rp /var/spool/pandora/data_in/md5 /pandora_files/  
cp -rp /var/spool/pandora/data_in/collections /pandora_files/  
cp -rp /var/spool/pandora/data_in/netflow /pandora_files/  
cp -rp /var/www/html/pandora_console/attachment /pandora_files/
```

Delete the old folders:

```
rm -rf /var/spool/pandora/data_in/conf  
rm -rf /var/spool/pandora/data_in/md5  
rm -rf /var/spool/pandora/data_in/collections  
rm -rf /var/spool/pandora/data_in/netflow  
rm -rf /var/www/html/pandora_console/attachment
```

And create the symlinks in both servers to the cluster:

```
ln -s /pandora_files/conf /var/spool/pandora/data_in/  
ln -s /pandora_files/md5 /var/spool/pandora/data_in/  
ln -s /pandora_files/collections /var/spool/pandora/data_in/  
ln -s /pandora_files/netflow /var/spool/pandora/data_in/  
ln -s /pandora_files/attachment /var/www/html/pandora_console/
```



Now both servers will be sharing the Pandora FMS key



directories, so the process is complete. In case you need more shared volume, follow the steps shown [in this section of this guide](#).

Increasing volume

It is possible to enlarge a GlusterFS volume with no downtime by increasing the number of bricks in a volume.

In order to do so, a new disk must be created, following the same steps as before:

```
lvcreate -L 5G -n brick2 vg_gluster
Logical volume "brick2" created.
```

Configure it as XFS:

```
mkfs.xfs /dev/vg_gluster/brick2
```

Create a new mount point and mount the new brick:

```
mkdir -p /bricks/brick2
mount /dev/vg_gluster/brick2 /bricks/brick2
```

Extend in /etc/fstab:

```
/dev/vg_gluster/brick2 /bricks/brick2 xfs defaults 0 0
```

Create the folder for the new brick:

```
mkdir /glusterfs/brick2/brick
```

Expand the volume:

```
gluster volume add-brick glustervol1
gluster1.example.com:/glusterfs/brick2/brick \
gluster2.example.com:/glusterfs/brick2/brick
```

Verify the volume:

```
gluster volume info glustervol1
Volume Name: glustervol1
Type: Distributed-Replicate
Volume ID: 6953a675-f966-4ae5-b458-e210ba8ae463
Status: Started
Number of Bricks: 2 x 2 = 4
Transport-type: tcp
Bricks:
```

```
Brick1: gluster1.example.com:/glusterfs/brick1/brick  
Brick2: gluster2.example.com:/glusterfs/brick1/brick  
Brick3: gluster1.example.com:/glusterfs/brick2/brick  
Brick4: gluster2.example.com:/glusterfs/brick2/brick
```

Check disk usage before the rebalancing:

```
df -h | grep brick
```

Rebalance:

```
gluster volume rebalance glustervol1 start
```

Check the rebalance:

```
gluster volume rebalance glustervol1 status
```

Check disk usage again:

```
df -h | grep brick
```

Check the files in the bricks:

```
ls -l /glusterfs/brick*/brick/
```

Configuring Tentacle Server for NFS concurrent access

If you want to store the agents' XML files in the same shared disk (instead of having each server handle their own locally), Tentacle servers of both computers must be configured so the XML files get distributed into separate folders. This will avoid problems when the Dataservers process the files in both Pandora FMS servers.

To that end, create two folders within the directory `/var/spool/pandora/data_in/`:


```
mkdir /var/spool/pandora/data_in/xml_srv1  
mkdir /var/spool/pandora/data_in/xml_srv2
```

Fix the permissions of both directories:

```
chmod pandora:apache /var/spool/pandora/data_in/xml_srv1  
chmod pandora:apache /var/spool/pandora/data_in/xml_srv2
```



In case you followed the GlusterFS guide, replace `/var/spool/pandora/data_in/` by `/pandora_files/` in the previous steps, and create the required symlinks:



```
ln -s /pandora_files/xml_srv1
/var/spool/pandora/data_in/
ln -s /pandora_files/xml_srv2
/var/spool/pandora/data_in/
```

Edit the TENTACLE_EXT_OPTS value in the file /etc/init.d/tentacle_serverd to set the XML file delivery folder:

```
TENTACLE_EXT_OPTS="-i.*\conf:conf;.*\md5:md5;.*\zip:collections"
```

In server number 1, it becomes:

```
TENTACLE_EXT_OPTS="-
i.*\conf:conf;.*\md5:md5;.*\zip:collections;.*\data:xml_srv1"
```

In sever number 2, it becomes:

```
TENTACLE_EXT_OPTS="-
i.*\conf:conf;.*\md5:md5;.*\zip:collections;.*\data:xml_srv2"
```

Finally, edit the configuration file of both Pandora FMS servers as follows:

```
# Pandora FMS server number 1
# incomingdir: It defines directory where incoming data packets are stored
# You could set directory relative to base path or absolute, starting with
/
incomingdir /var/spool/pandora/data_in/xml_srv1
```

```
# Pandora FMS server number 2
# incomingdir: It defines directory where incoming data packets are stored
# You could set directory relative to base path or absolute, starting with
/
incomingdir /var/spool/pandora/data_in/xml_srv2
```

After applying all the indicated changes, restart both the pandora_server service as well as the tentacle_serverd service in **both** servers.

[Go back to Pandora FMS documentation index](#)

From:
<https://pandorafms.com/manual/> - Pandora FMS Documentation

Permanent link:
https://pandorafms.com/manual/en/documentation/07_technical_annexes/10_share_var_spool_directory_between_several_pandora_servers

Last update: 2021/09/16 09:17

