Backup the data

Database Backup

1. This MySQL server hosts the databases for nova, glance, cinder, and keystone. With all of these databases in one place, so it's easy to create openstack services database backup.

```
mysqldump --opt --all-databases > openstack.sql
```

File System Backup

1. Compute

The `/etc/nova` directory on both the cloud controller and compute nodes should be regularly backed up.

`/var/log/nova` contains all the logs. So, this directory on both controller and compute nodes should be regularly backed up.

`/var/lib/nova` is another important directory to back up. The exception to this is the `/var/lib/nova/instances` subdirectory on compute nodes. This subdirectory contains the KVM images of running instances. If need to maintain backup copies of all instances then only back up this directory.

2. Image Catalog and Delivery

`/etc/glance` and `/var/log/glance` follow the same rules as their nova counterparts.

`/var/lib/glance` should also be backed up. Take special notice of `/var/lib/glance/images`. If using a file-based back end of glance, `/var/lib/glance/images` is where the images are stored and care should be taken.

3. Identity

`/etc/keystone` and `/var/log/keystone` follow the same rules as other components.

`/var/lib/keystone`, although it should not contain any data being used, can also be backed up just in case.

4. Block Storage

`/etc/cinder` and `/var/log/cinder` follow the same rules as other components.

`/var/lib/cinder` should also be backed up.

5. Networking

`/etc/neutron` and `/var/log/neutron` follow the same rules as other components.

`/var/lib/neutron` should also be backed up.

Automate the Backup task

can easily automate this process by creating a cron job that runs the following script once per day:

```
#!/bin/bash
backup_dir="/srv/node/sdc/backup-openstack/"
log_dir="/srv/node/sdc/backup-openstack/log"
lib_dir="/srv/node/sdc/backup-openstack/lib"
conf_dir="/srv/node/sdc/backup-openstack/conf"
filename="${backup_dir}/mysql-`hostname`-`eval date +%Y%m%d`.sql.gz"
# Dump the entire MySQL database
/usr/bin/mysqldump --opt --all-databases | gzip > $filename

# copy the all the directory to create the backup
cp -r /etc/chrony $conf_dir
cp -r /etc/memcached.conf $conf_dir
cp -r /etc/mysql $conf_dir
cp -r /etc/default/etcd $conf_dir
cp -r /etc/nova $conf_dir
cp -r /etc/keystone $conf_dir
cp -r /etc/glance $conf_dir
cp -r /etc/placement $conf_dir
cp -r /etc/neutron $conf_dir
cp -r /etc/cinder $conf_dir
cp -r /etc/swift $conf_dir
cp -r /etc/openstack-dashboard $conf_dir

#backup the lib directory files
cp -r /var/lib/chrony $lib_dir
cp -r /var/lib/nova $lib_dir
cp -r /var/lib/keystone $lib_dir
cp -r /var/lib/glance $lib_dir
cp -r /var/lib/placement $lib_dir
cp -r /var/lib/neutron $lib_dir
```

cp -r /var/lib/cinder $lib_dir

cp -r /var/lib/swift $lib_dir

#backup the lib directory files

cp -r /var/log/chrony $log_dir

cp -r /var/log/mysql $log_dir

cp -r /var/log/rabbitmq $log_dir

cp -r /var/log/nova $log_dir

cp -r /var/log/keystone $log_dir

cp -r /var/log/glance $log_dir

cp -r /var/log/placement $log_dir

cp -r /var/log/neutron $log_dir

cp -r /var/log/openvswitch $log_dir

cp -r /var/log/cinder $log_dir

cp -r /var/log/swift $log_dir

Delete backups older than 7 days

find $backup_dir -ctime +3 -type f -delete

Recovering Backups

1. To begin, first ensure that the service you are recovering is not running.

For example:

sudo service nova-api stop

sudo service nova-scheduler stop

sudo service nova-conductor stop

sudo service nova-novncproxy stop

sudo service nova-compute stop

2. Now, can import a previously backed-up database:

```
create nova;

create nova_api;

create nova_cell0;
```

`mysql -u root -p nova < path/to/backupfile.sql`

use nova;

show tables;

3. can also restore backed-up nova directories

cp -a /path/to/backup/nova /etc/

4. Once the files are restored, start everything back up:

sudo service mysql restart

sudo service nova-api restart

sudo service nova-scheduler restart

sudo service nova-conductor restart

sudo service nova-novncproxy restart

sudo service nova-compute restart