rubytreesoftware.com

http://www.rubytreesoftware.com/resources/basic-postgresql-backup-and-restore

Basic PostgreSQL Backup and Restore

Learn how to simply backup and restore a PostgreSQL database using SQL dumps. Learn different techniques and get code and scripts you can use.

All commands should be run as the postgres user.

```
sudo su - postgres
```

Basic backup and restore commands

```
# Backup a single database
pg_dump db_name > db_backup.sql

# Restore a single database
psql db_name < db_backup.sql

# Backup an entire postgres database cluster
pg_dumpall > cluster_backup.sql

# Restore an entire postgres database cluster
psql -f cluster_backup.sql postgres
```

Backup and Restore using compression

```
# Backup a single database
pg_dump db_name | gzip > db_backup.gz

# Restore a single database
gunzip -c db_backup.gz | psql db_name

# Backup a database cluster
pg_dumpall | gzip > cluster_backup.gz

# Restore a database cluster
gunzip -c cluster_backup.gz | psql postgres
```

Backup and be able to restore individual tables

```
# Backup a single database
pg_dump -Fc db_name > db_backup.dmp

# Restore a single database
pg_restore -d db_name db_backup.dmp

# Can use pg_dumpall to backup all global information
# then use pg_dump to backup each database
pg_dumpall > global_only_backup.sql --globals-only
```

Restore the database and stop on errors

```
psql db_name < db_backup.sql --set ON_ERROR_STOP=on
```

After restore, vacuum and analyze tables

```
vacuumdb -a -z
```

Basic Backup script

```
#!/bin/bash
#
# Takes a full backup of the database and stores it in the backup folder
# Run this script as the postgres user
#

DATE=`date +%Y-%m-%d`

echo `date` - Delete old backups
find ~/backup/* -mtime +1 -delete

echo `date` - Do a full postgres cluster dump
pg_dumpall | gzip > ~/backup/db_cluster_dump_$DATE.gz

echo `date` - Sync pg_backups with S3
# /usr/local/bin/aws s3 sync ~/backup s3://bucket_name/backup

echo `date` - Sync postgres configuration files with S3
# /usr/local/bin/aws s3 sync /etc/postgresql/9.3/main s3://bucket_name/backup

echo `date` - Backup complete
```

Schedule the script using cron

```
# Make sure script is executable by postgres
chmod 770 backup_script.sh

crontab -e
```

```
# m h  dom mon dow   command
  0 4  *   *   *     ~/backup_script.sh > ~/backup_script.log 2>&1
```

Please go ahead and leave a comment below if you have any questions about this tutorial.