Database Backup and Recovery Guide

1. MySQL Database Backup and Recovery

1.1 Backup MySQL Database

You can use the `mysqldump` command to create a backup of your database.

Backup Command:

mysqldump -u root -p mydatabase > mydatabase_backup.sql

1.2 Restore MySQL Database

To restore from a backup file, use the `mysql` command.

Restore Command:

mysql -u root -p mydatabase < mydatabase_backup.sql

2. PostgreSQL Database Backup and Recovery

2.1 Backup PostgreSQL Database

PostgreSQL uses `pg_dump` to create a backup.

Backup Command:

pg_dump -U postgres -F c -b -v -f mydatabase_backup.dump mydatabase

2.2 Restore PostgreSQL Database

Use `pg_restore` to restore the database.

Restore Command:

pg_restore -U postgres -d mydatabase -v mydatabase_backup.dump

3. Automating Database Backups

3.1 Automating MySQL Backup (Linux)

1. Open the crontab:

   crontab -e

2. Add the following line to run the backup every day at midnight:

   0 0 * * * mysqldump -u root -p'password' mydatabase > /backups/mydatabase_$(date +\%F).sql

3.2 Automating PostgreSQL Backup (Linux)

1. Open the crontab:

   crontab -e

2. Add the following line to run the backup every day at midnight:

   0 0 * * * pg_dump -U postgres -F c -b -v -f /backups/mydatabase_$(date +\%F).dump mydatabase

4. Ensuring Data Integrity

1. **Verify Backup Size**

   ls -lh mydatabase_backup.sql

2. **Check Backup Content**

   head -n 20 mydatabase_backup.sql

3. **Test Restoration in a Separate Environment**

   Restore the backup in a test environment before using it in production.

5. Deliverables

Backup & Recovery Scripts

Step-by-Step Documentation

Automation Using Cron Jobs

This guide ensures secure and reliable backup and recovery for MySQL and PostgreSQL databases.