# **GitLab Backup to S3**

The steps to take a GitLab backup and upload it to Amazon S3:

**Step 1: Take GitLab Backup**

SSH into your GitLab server.

Use the belowcommand to create a backup of your GitLab data:

sudo gitlab-rake gitlab:backup:create

It will create a backup archive in the **/var/opt/gitlab/backups** directory.

**Step 2: Install and Configure AWS CLI**

Install the AWS Command Line Interface (AWS CLI) if you haven't already:

sudo apt install awscli

Configure the AWS CLI with your AWS access key, secret key, and default region:

aws configure

**Step 3: Upload Backup to S3**

Use the **aws s3 cp** command to upload the backup to your S3 bucket. Replace **backup.tar** with the actual name of your backup file, and replace **your-bucket-name** with the name of your S3 bucket:

aws s3 cp /var/opt/gitlab/backups/backup.tar s3://your-bucket-name/

**Step 4: Verify Backup Upload**

To verify that the backup has been uploaded to S3, you can list the contents of your S3 bucket:

aws s3 ls s3://your-bucket-name/

**Step 5: Automate the Process (Optional)**

If you want to automate the backup and upload process, you can create a shell script that includes the backup creation and AWS S3 upload commands. You can then schedule this script to run at regular intervals using a cron job.

Here's a simple example of what the script might look like:

#!/bin/bash

# Take GitLab backup

sudo gitlab-rake gitlab:backup:create

# Upload backup to S3

aws s3 cp /var/opt/gitlab/backups/backup.tar s3://your-bucket-name/

Save this script to a file (e.g., **backup\_script.sh**), make it executable (**chmod +x backup\_script.sh**), and then add a cron job to run it periodically:

crontab -e

Add a line like this to run the script every day at midnight:

0 0 \* \* \* /path/to/backup\_and\_upload.sh

replace **/path/to/backup\_script.sh** with the actual path to your script.