

# Standard Operating Procedure (SOP) for MongoDB Database Backup to S3



## Introduction

- ⌵ This document outlines the step-by-step procedure for backing up a MongoDB database to Amazon S3. Follow these instructions carefully to ensure a smooth and secure backup process.



## Procedure

### 1. Install and Configure s3cmd

```
sudo apt update  
sudo apt install s3cmd -y
```

## Configure s3cmd:

```
s3cmd --configure
```

During configuration, you'll need to provide the following information:

- **Access Key:** Your AWS access key ID 🔑
- **Secret Key:** Your AWS secret access key 🗝️
- **Default Region:** The AWS region you want to use (e.g., `us-west-2`) 🌐
- **S3 Endpoint:** Leave this blank for default settings
- **Use HTTPS:** Confirm with `y` for yes 🔒
- **Load config from the given file:** Press Enter to accept the default
- **Save settings:** Confirm with `y` for yes 💾

The configuration will be saved to `/root/.s3cfg`.

## Verify s3cmd Installation:

Check if the installation is successful by listing S3 buckets:

```
s3cmd ls
```

## 2. Create the Backup Script

### Create the Backup Script:

```
vim /opt/path/to/script/<filename>.sh
```

### Add the Following Script to the File:

```
#!/bin/bash

# Set the directory name as today's date
DIR=$(date +%d-%m-%y)
DEST=/opt/backup/temp/path/$DIR

# Create the destination directory
mkdir $DEST

# Backup MongoDB Database
mongodump --username <user name> --password <password> --db <db name> --out $DEST

# Create a tarball of the backup
sudo tar -cf $DEST.tar -C $DEST .

# Move to S3
sudo cp $DEST.tar /usr/bin/backup/backup/2016/03/16/
```

```
# Move to S3
sudo s3cmd put $DEST.tar s3://<bucket name>/path/of/file/

# Remove local files
sudo rm -rf /opt/backup/temp/path/*
```

## Make the Script Executable:

```
chmod 755 /opt/path/to/script/<filename>.sh
```

## 3. Schedule the Backup 🕒

Open the crontab file for editing:

```
crontab -e
```

## Add a Cron Job to Schedule the Backup:

```
30 19 * * * /opt/path/to/script/<filename>.sh
```

Open the crontab file for editing:

```
crontab -e
```

### Add a Cron Job to Schedule the Backup:

```
30 19 * * * /opt/path/to/script/<filename>.sh
```

This cron job will run the backup script every day at 7:30 PM.



## Conclusion

By following this SOP, you've successfully set up an automated MongoDB database backup to Amazon S3. This process ensures your data is regularly backed up and stored securely in the cloud. Remember to periodically check your S3 bucket to confirm that backups are being uploaded successfully. Happy backing up! 🚀