# Rclone Backup to Box for Cluster

Markus G. S. Weiss

2025/04/23

# Contents

# 1   Introduction

This tutorial explains how to configure `rclone` on your cluster to back up `/mfs/io/groups/sterling/mf`
to a Box directory named `cluster-backup`, with subfolders for `daily`, `archive`, and
`logs`, and how to schedule it via cron. Users in the sterling group only need to run
the commands in sections 2, 3, 4, and 6. The scripts are maintained centrally under
`/mfs/io/groups/sterling/setup`.

# 2   Prerequisites

- rclone (v1.38 or later) installed on both the cluster and your desktop (with a
  browser)

- Confirm rclone versions match:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1
    .69.1-linux-amd64/rclone version
rclone version
```

- A Box Enterprise SSO account

- Shell access to the cluster with cron available

- **Tip:** Before running any live syncs, test with `-dry-run`:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1
    .69.1-linux-amd64/rclone sync \
  /mfs/io/groups/sterling/mfshome/$USER box:cluster-backup
      /daily \
  --dry-run --fast-list --checksum
```

# 3   Configure the Box Remote with Offline Authorization

Run the rclone config command on the cluster:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone config
```

Press **Enter** to accept each default (shown in <>):

```
No remotes found, make a new one? n
name> box
Storage> box
client_id> <leave blank>
client_secret> <leave blank>
box_config_file> <leave blank>
access_token> <leave blank>
box_sub_type> 2
Edit advanced config? n
Use web browser to authenticate? n
```

rclone will then print a command:

```
rclone authorize "box" "xxxxxxxxxxxxxxxx"
```

1. Copy that exact command to your local machine and run it; complete the OAuth flow in your browser.

2. rclone prints a long token string; back on the cluster, paste it at:

```
config_token> xxxxxxxxxxxxxxxx
```

3. When asked, confirm: y

4. Verify:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1
    .69.1-linux-amd64/rclone lsd box:
```

# 4    Create the Box Folder Hierarchy

On the cluster, run:

```
# Parent folder
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone mkdir box:cluster-backup
# Subfolders
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone mkdir box:cluster-backup/daily
```

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone mkdir box:cluster-backup/archive
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone mkdir box:cluster-backup/logs
```

Verify:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1.69.1-
    linux-amd64/rclone lsd box:cluster-backup
```

# 5   Prepare the Local Environment

Create a local logs directory:

```
mkdir -p ~/logs
```

# 6   Reference Scripts

Scripts are in **/mfs/io/groups/sterling/setup**.

## 6.1   backup.sh

```bash
#!/usr/bin/env bash
set -euo pipefail

data_dir="/mfs/io/groups/sterling/mfshome/$USER"
remote_root="box:cluster-backup"
rclone_bin="/mfs/io/groups/sterling/software-tools/rclone/
    rclone-v1.69.1-linux-amd64/rclone"
date_str=$(date +%F)

# 1) Daily incremental
"$rclone_bin" sync "$data_dir" "${remote_root}/daily" --fast-
    list --checksum --log-file "$HOME/logs/backup-$date_str.log
    " --log-level INFO

# 2) Weekly snapshot (Sundays)
if [[ "$(date +%u)" == "7" ]]; then
```

```
  "$rclone_bin" sync "$data_dir" "${remote_root}/archive/
      $date_str" --fast-list --checksum --log-file "$HOME/logs/
      snapshot-$date_str.log" --log-level INFO
fi

# 3) Upload logs
"$rclone_bin" sync "$HOME/logs" "${remote_root}/logs" --fast-
    list --log-level INFO
```

## 6.2 cronscript

```
SHELL=/bin/bash
PATH=/usr/local/bin:/usr/bin:/bin
MAILTO=$USER@utdallas.edu
TZ=Europe/Berlin

# Run backup.sh daily at 02:00
0 2 * * * /mfs/io/groups/sterling/setup/backup.sh

# Rotate old snapshots (keep 4 weeks)
0 3 1 * * /mfs/io/groups/sterling/software-tools/rclone/rclone
    -v1.69.1-linux-amd64/rclone delete --min-age 28d box:
    cluster-backup/archive
```

# 7 Install the Cron Job

Install via:

```
crontab /mfs/io/groups/sterling/setup/cronscript
crontab -l
```

# 8 Monitoring & Maintenance

- View logs: tail $-$f ~/logs/backup$-$\$(date $+$%F).log

- Clean local logs older than 30 days:

```
find ~/logs -type f -mtime +30 -delete
```

- Test restore:

```
/mfs/io/groups/sterling/software-tools/rclone/rclone-v1
    .69.1-linux-amd64/rclone copy box:cluster-backup/daily/
    path/to/file /tmp && diff /tmp/file /mfs/io/groups/
    sterling/mfshome/$USER/path/to/file
```

- Alerts: Cron emails stderr/stdout. For advanced alerting, grep logs for ERROR or integrate with Slack.

# 9 Additional Notes

- Security: Keep `/.config/rclone/rclone.conf` private. Use a `crypt` wrapper for encryption.

- API rate limits (side note): Adjust `-transfers`, `-checkers`, or add `-tpslimit` if you encounter errors.

- Network/firewall (side note): Ensure outbound HTTPS (443). If behind a proxy, set `HTTPS_PROXY` or use `-proxy`.

- Monthly snapshots: Extend logic with if [ "$(date +%d)" == "01" ].

- Upstream docs: `https://rclone.org/box/`

# 10 Conclusion

In this tutorial, you have:

- Configured offline SSO authorization on a headless cluster

- Created an organized Box folder hierarchy under `cluster-backup`

- Prepared local logging and referenced centralized scripts

- Automated daily syncs and weekly snapshots via `backup.sh` and cron

- Established monitoring, restore procedures, and cleanup routines

- Included best-practice notes on dry-runs, version checks, security, API limits, and network requirements

**Great work!** Your cluster home directory is now automatically and safely backed up to Box every night, with versioning, logs, and tools for easy maintenance and recovery.