

# Observability

- Logrotation for Log & AccessLog

# Logrotation for Log & AccessLog



## Introduction

This post is about enabling log rotation for the Traefik `access.log` and `traefik.log`. Some information is sourced from [this post](#), with additional insights by Aeoneros.

Traefik will close and reopen its log files, assuming they're configured, upon receiving a `USR1` signal. This enables the logs to be rotated and processed by external programs like `logrotate`. However, this does not work on Windows due to the lack of USR signals.

The `access.log` and `traefik.log` require proper setup to ensure efficient log management. Below are the steps to manually create log rotation for these logs.

## Step 1: Configure the Paths for Logs

In your Traefik static configuration file (e.g., `traefik.toml`), define the file paths for the logs. For example:

```
[accessLog]
  filePath = "/mnt/glustermount/data/traefik_data/access.log"

[log]
  filePath = "/mnt/glustermount/data/traefik_data/traefik.log"
```

Refer to the official documentation for [access logs](#) and [general logs](#).

## Step 2: Install Logrotate

Ensure the `logrotate` package is installed on your Debian-based system. Install it if necessary:

```
sudo apt update
sudo apt install logrotate
```

## Step 3: Create a Logrotate File

Create a new configuration file for Traefik log rotation:

```
sudo nano /etc/logrotate.d/traefik
```

## Step 4: Configure Logrotate

Paste the following content into the file:

```
/mnt/glustermount/data/traefik_data/access.log
/mnt/glustermount/data/traefik_data/traefik.log {
    compress
    create 0640 root root
    daily
    delaycompress
    missingok
    notifempty
    rotate 5
    postrotate
        # Send USR1 signal to reopen logs
        docker kill --signal=USR1 $(docker ps --filter "name=traefik_traefik" --format "{{.ID}}")
        # Pause to give Traefik time to process the signal
        sleep 5
        # Verify that Traefik is writing to the correct log files
        if [ -s /mnt/glustermount/data/traefik_data/access.log.1 ] || [ -s
/mnt/glustermount/data/traefik_data/traefik.log.1 ]; then
            # Restart service if still writing to the rotated file
            docker service update --force traefik_traefik
```

```
fi
endscript
}
```

### Explanation:

- The `postrotate` script first sends a `USR1` signal to Traefik to reopen its log files.
- If Traefik continues writing to the rotated files, it restarts the container using `docker service update --force`.
- This ensures that both logs are properly rotated and written to their new files.

## Step 5: Debugging Logrotate

Test your `logrotate` configuration in debug mode:

```
sudo logrotate -d /etc/logrotate.d/traefik
```

## Step 6: Force Logrotate

Force log rotation to execute the script:

```
sudo logrotate -f /etc/logrotate.d/traefik
```

## Step 7: Verify Log Rotation

Check if new files were created and if Traefik is writing to the correct log files:

```
ls -lh /mnt/glustermount/data/traefik_data/*.log*
tail -f /mnt/glustermount/data/traefik_data/access.log
tail -f /mnt/glustermount/data/traefik_data/traefik.log
```

## Troubleshooting

If nothing is working, revert the changes by following these steps:

1. Rename or delete the logrotate file:

```
sudo mv /etc/logrotate.d/traefik /etc/logrotate.d/traefik.bak
```

1. Edit the logrotate status file to remove entries related to the logs:

```
sudo nano /var/lib/logrotate/status
```

1. Look for and remove entries related to `/mnt/glustermount/data/traefik_data/access.log` and `/mnt/glustermount/data/traefik_data/traefik.log`.

Reapply the steps once the issue is resolved.