How I Back Up My VPS with Syncthing

This guide documents how I use Syncthing to back up my VPS's critical BookStack data folder (`/opt/bookstack_backups`) to my personal workstation automatically and securely.

□□ Backup Directory

I use a dedicated folder on my VPS for storing compressed BookStack backups:

```
/opt/bookstack_backups
```

Each backup is generated nightly via a cron job and includes both the database and uploaded content.

□□ Syncthing Folder Setup

On both the VPS and my workstation:

- Syncthing is installed and configured as a **systemd service**
- The `/opt/bookstack_backups` folder is shared
- Sync is **one-way from VPS to workstation** for integrity

To enable and start the system-wide Syncthing service (replace `yourusername`):

```
sudo systemctl enable syncthing@yourusername.service
sudo systemctl start syncthing@yourusername.service
```

To verify the service is running:

```
systemctl status syncthing@yourusername.service
```

> ⚠ Tip: If Syncthing is installed system-wide or runs under root, use the appropriate service name or create a dedicated user account just for it.

□□ UFW Firewall Rule with DDNS Lockdown

To limit Syncthing access to my workstation only, I use a dynamic DDNS-resolved UFW rule.

Example IP update script:

```
/opt/scripts/update-syncthing-ufw.sh
```

This script:

- Resolves the current DDNS IP
- Checks for existing UFW rules
- Updates UFW only when changes are detected

Example UFW rule:

```
sudo ufw allow from 123.45.67.89 to any port 22000 proto tcp
```

Blocked access attempts are logged and managed with Fail2Ban.

□□ Backup Cron Job

The backup script runs daily at **2:00 AM** using cron.

Crontab entry:

```
0 2 * * * /opt/scripts/bookstack-backup.sh
```

Script actions:

- Dumps the MySQL database
- Archives the uploaded files
- Compresses to `.tar.gz` with the date in the filename

□□ Local Redundancy Strategy

Once synced to my local system:

- Backups are **rotated weekly**
- A cleanup script deletes older archives
- Restores can be performed with:

```
tar -xzvf bookstack-backup-YYYY-MM-DD.tar.gz -C /restore/location
```

□□ Supporting Scripts

These scripts make the entire backup and security process seamless:

Script Name	Purpose
`bookstack-backup.sh`	Creates nightly backups
`update-syncthing-ufw.sh`	Updates UFW with resolved DDNS IP
`syncthing-log-summary.sh`	Parses and displays UFW-blocked Syncthing traffic
`bookstack-logrotate.conf`	Handles log file cleanup

□□ Key Takeaways

- Daily automated backup process
- Synced offsite to a secure system
- DDNS + UFW keeps access tightly controlled
- Full restore with a single `tar` command

Revision #2
Created 13 May 2025 19:15:24 by Nate Nash
Updated 4 June 2025 00:31:42 by Nate Nash