

Manage & Operate Semaphore

- [Semaphore Backup Guide](#)
- [Restore Semaphore](#)

Semaphore Backup Guide

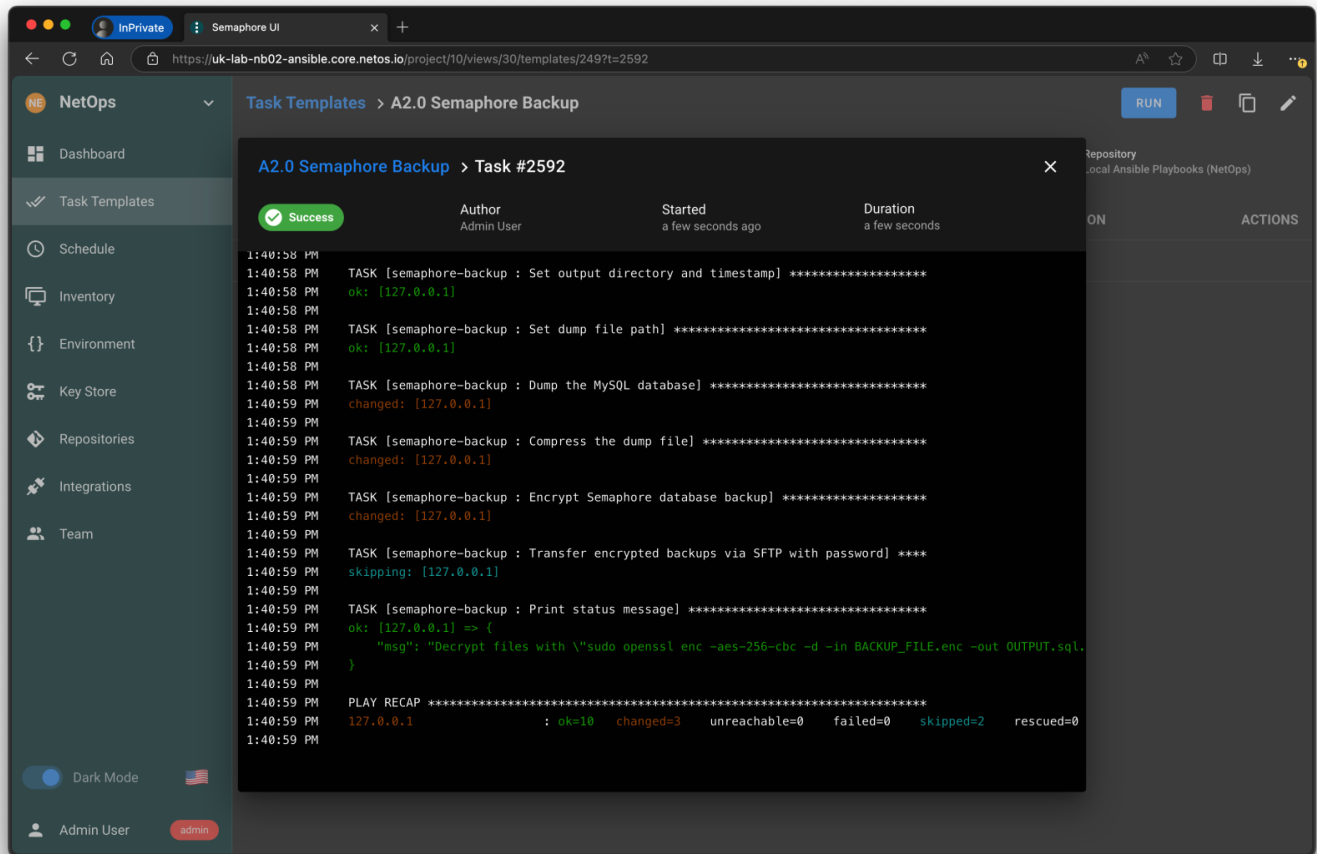
Introduction

The Semaphore backup process provides regular secure backups. In summary the playbook:

1. Manages retention based on your preferences, e.g. deletes backups older than 14 days old
2. Backs up and encrypts the `/netos/` directory, which contains many important things, such as:
 1. Certificates (for NGINX)
 2. Past NetBox plugin ZIP file downloads
 3. Working directories for other applications and tools
 4. Backups (which are excluded from the backup)
3. Backs up the `/etc/nginx/` directory, which contains all the site configurations for different services deployed by Semaphore, such as Semaphore itself, NetBox, Airflow, etc.
4. Backs up and encrypts the `semaphore` MySQL database
5. Optionally SFTP's the backup files to a secure remote SFTP server

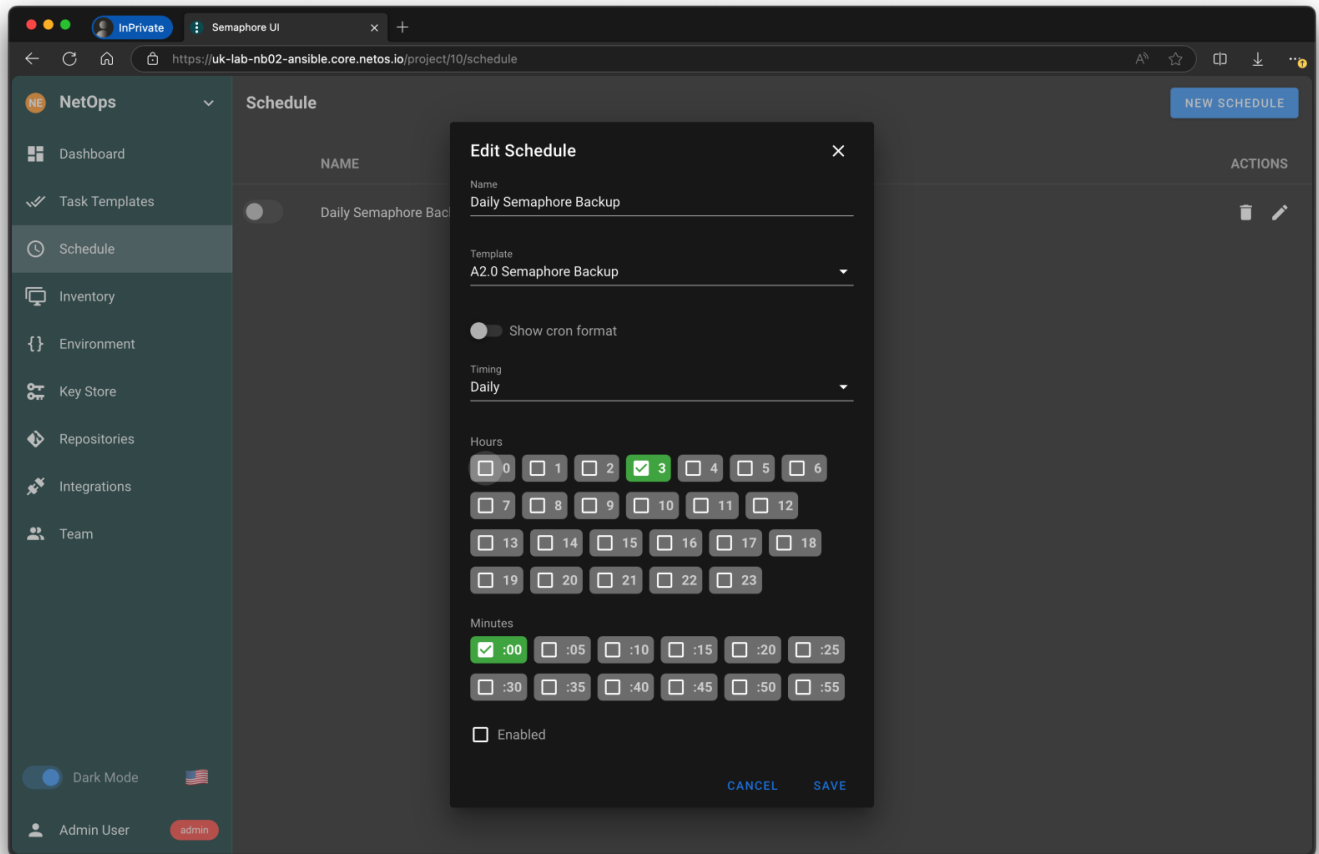
The Semaphore process and MySQL database is NOT stopped during the backup.

Semaphore uses MySQL and not PostgreSQL to ensure isolation from other applications running on the server. For example, NetBox and Airflow both use PostgreSQL, and if/when those databases are restarted, we don't want to impact the management wrapper, i.e. Semaphore.



Cron Scheduling

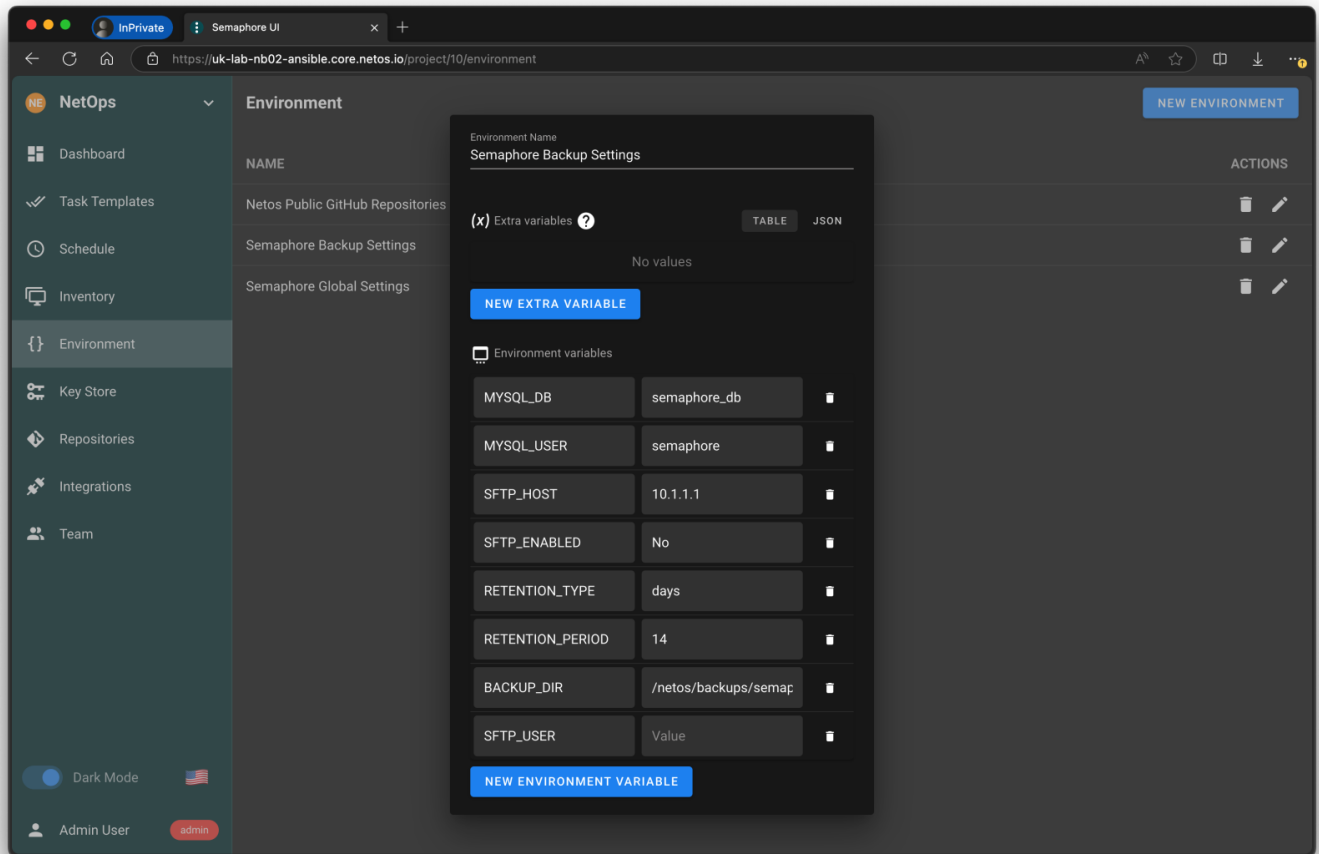
The cron scheduler in Semaphore can be configured to backup the database at regular intervals, for example, at 02:30 every day. You can check the Dashboard page in the menu, or the task history to check the outcomes.



Note that there is a [bug in Ansible Semaphore UI](#) that causes the same task to run many times. The solution is to toggle the "Show cron format" button and use UNIX formatting like [here](#).

Backup File Rotation

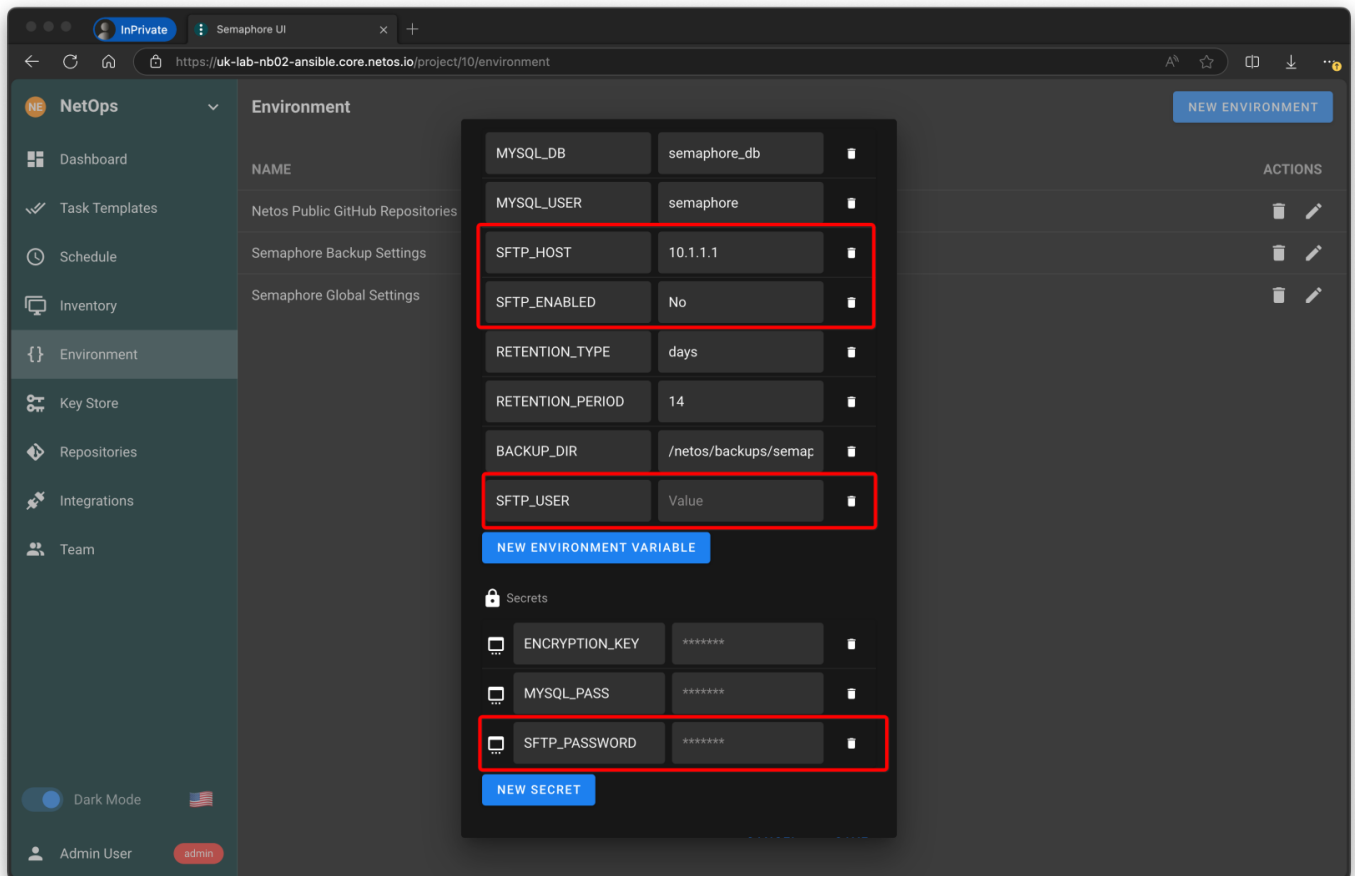
You can set the retention period for backup files stored in `/netos/backups/semaphore` in the Environment / variables.



Ensure you use the exact values of `days` or `weeks`, i.e. no capitals.

Remote SFTP

To enable remote SFTP, change the `No` value in the `SFTP_ENABLED` variable to `Yes`, and set the `SFTP_HOST/USER/PASS` values accordingly.



An example of the encrypted backup folder contents is as follows:

```
netosadm@uk-lab-nb02: /netos/backups/semaphore
netosadm@uk-lab-nb02: /netos/backups/semaphore$
netosadm@uk-lab-nb02: /netos/backups/semaphore$
netosadm@uk-lab-nb02: /netos/backups/semaphore$ ls -alh
total 408K
drwxr-xr-x 2 root root 4.0K Sep 26 12:40 .
drwxr-xr-x 6 root root 4.0K Sep 24 09:42 ..
-rw-r--r-- 1 root root 97K Sep 26 12:39 semaphore_db-20240926123938.sql.gz
-rw-r--r-- 1 root root 97K Sep 26 12:39 semaphore_db-20240926123938.sql.gz.enc
-rw-r--r-- 1 root root 98K Sep 26 12:40 semaphore_db-20240926124058.sql.gz
-rw-r--r-- 1 root root 98K Sep 26 12:40 semaphore_db-20240926124058.sql.gz.enc
netosadm@uk-lab-nb02: /netos/backups/semaphore$
```


Restore Semaphore

Manual Restore

We haven't included a playbook in Semaphore to restore itself, because it's a bit like sawing the branch off that you're hanging from. To restore from your Semaphore backup, you can follow this process.

1. Start with a fresh install following the [Deploy Semaphore Guide](#), except CTRL-C and exit the installation.
2. Search and replace the `semaphore-netos-netops.sql` file in the `/netos/netos-netops/roles/semaphore-install/files` with your SQL backup and restart the install using `deploy.sh`
3. Restore the tar.gz file to `/netos/`
4. Restore the tar.gz file to `/etc/nginx`
5. Restart the server and connect to Semaphore

Decrypting Backup Files

If you want to manually decrypt the `enc` files on your local workstation, use the following commands.

```
openssl enc -aes-256-cbc -d -in BACKUP_FILE.enc -out OUTPUT.sql.gz/.tar.gz
```

You will need the password set in the Semaphore Backup Settings Semaphore environment variable `ENCRYPTION_KEY`.