

SELF-HOSTING

# Backup your Hosted Data

View in the help center:  
<https://bitwarden.com/help/backup-on-premise/>

## Backup your Hosted Data

When self-hosting Bitwarden, you are responsible for implementing your own backup procedures in order to keep data safe.

### About hosted data

Bitwarden's Docker containers use volume mapping to persist all important data on the host machine, meaning stopping your containers will not delete any data. Docker containers, on the other hand, are to be considered ephemeral and do not persist data or state.

All Bitwarden data is stored on the host machine in the `./bwdata` directory, relative to the location in which you installed Bitwarden.

### Backup hosted data

It's recommended that you backup and keep safe the entire `./bwdata` directory. In the event of data loss, you will need all or parts of the data contained in this directory to restore your instance.

Particularly important pieces of `./bwdata` to backup regularly include:

- `./bwdata/env` - Instance's environment variables, including database and certificate passwords.
- `./bwdata/core/attachments` - Instance's vault item attachments.
- `./bwdata/mssql/data` - Instance's database data.
- `./bwdata/core/aspnet-dataprotection` - Framework-level data protection, including authentication tokens and some database columns.

Bitwarden will automatically take nightly backups of the `mssql` database container, when running.

### Nightly database backups

Bitwarden will automatically take nightly backups of the `mssql` container database. These backups are kept in the `./bwdata/mssql/backups` directory for 30 days.

In the event of data loss, you can use `./bwdata/mssql/backups` to restore a nightly backup.

### Restore a nightly Backup

In the event of data loss, complete the following steps to restore a nightly backup.

1. Retrieve your database password from the `globalSettings__sqlServer__connectionString=...Password=` value found in `global.override.env`.
2. Identify the Container ID of the `mssql` container using the `docker ps` command.
3. Run the following command to open a bash session for your `mssql` docker container:

*Bash*

```
docker exec -it bitwarden-mssql /bin/bash
```

Your command prompt should now match the identified Container ID of the `bitwarden-mssql` container.

4. In the container, locate the backup file you wish to restore.

**Note**

The backup directory in the container is volume-mapped from the host directory. `./bwdata/mssql/backups` on the host machine maps to `etc/bitwarden/mssql/backups` in the container.

For example, a file `/etc/bitwarden/mssql/backups/vault_FULL_20201208_003243.BAK` is a backup taken on December 08, 2020 at 12:32am.

5. Start the `sqlcmd` utility with the following command:

*Bash*

```
/opt/mssql-tools/bin/sqlcmd -S localhost -U <sa> -P <sa-password>
```

where `<sa>` and `<sa-password>` match the `User=` and `Password=` values found in `global.override.env`.

6. Once in the `sqlcmd` utility, you have two options for backup:

**1. Offline restore (Preferred)**

Run the following SQL commands:

*Bash*

```
1> use master
2> GO
1> alter database vault set offline with rollback immediate
2> GO
1> restore database vault from disk='/etc/bitwarden/mssql/backups/vault_FULL_{Backup File Name}.BAK' with replace
2> GO
1> alter database vault set online
2> GO
1> exit
```

Restart your Bitwarden instance to finish restoring.

**2. Online restore**

Execute the following SQL commands:

*Bash*

```
1> RESTORE DATABASE vault FROM DISK = '/etc/bitwarden/mssql/backups/vault_FULL_20200302_23590
1.BAK' WITH REPLACE
2> GO
```

Restart your Bitwarden instance to finish restoring.