

Storage settings



- [Changing the storage for all subsystems](#)
 - [Using local](#)
 - [Using minio](#)
- [Storage settings for a single subsystem](#)
- [Storage settings](#)
 - [Using local](#)
 - [Using minio](#)
- [S3 servers compatibility](#)
- [Undocumented features](#)
- [Legacy settings](#)

The storage for each subsystem is defined in `app.ini`. It can either be on disk (`local` which is the default) or using a S3 compatible server (`minio`). In both cases each subsystem stores all files (or objects in the S3 parlance) in a dedicated directory as shown in the table below:

subsystem	directory	app.ini sections
Attachments	attachments/	[attachment]
LFS	lfs/	[lfs]
Avatars	avatars/	[avatar]
Repository avatars	repo-avatars/	[repo-avatar]
Repository archives	repo-archive/	[repo-archive]
Packages	packages/	[packages]
Actions logs	actions_log/	[storage.actions_log]
Actions Artifacts	actions_artifacts/	[actions.artifacts]

For instance:

- if the `STORAGE_TYPE` is `local` and `APP_DATA_PATH` is `/appdata`, the default directory to store attachments is `/appdata/attachments`

- if the `STORAGE_TYPE` is `minio`, the default directory to store attachments within the `MINIO_BUCKET` bucket will be `attachments/`.

Changing the storage for all subsystems

The `[storage]` section can be used to modify the storage of all subsystems. The default is to use `local` storage under `APP_DATA_PATH` and is equivalent to writing the following in `app.ini`:

```
[server]
APP_DATA_PATH = /forgejo/data

[storage]
STORAGE_TYPE = local
PATH = /forgejo/data
```

Using local

For `local` storage, the `[storage]` section can only be used to change the path under which all subsystems directories will be created, using the `PATH` settings with an absolute pathname.

For instance:

```
[storage]
STORAGE_TYPE = local
PATH = /mystorage
```

will change the default for storing attachments to `/mystorage/attachments`, LFS to `/mystorage/lfs` etc.

Using minio

The `[storage]` section can be used to change the default storage type used by all subsystems to `minio`.

For instance:

```
[storage]
STORAGE_TYPE = minio

MINIO_ENDPOINT = 127.0.0.1:9000
MINIO_ACCESS_KEY_ID = [redacted]
MINIO_SECRET_ACCESS_KEY = [redacted]
MINIO_BUCKET = forgejo
MINIO_LOCATION = us-east-1
```

will change the default for storing attachments to `attachments/` in the `forgejo` bucket, LFS to `lfs/` in the `forgejo` bucket etc.

“ [!note] Note `MINIO_BASE_PATH` must not be set in the `[storage]` section.

The configuration option `MINIO_USE_SSL` defaults to `false` to maintain compatibility with locally hosted MinIO instances. If an external S3 provider is intended to be used, this option should be set to `true`.

For instance, assuming a MinIO instance at `https://minio.example.com`:

```
[storage]
STORAGE_TYPE = minio

MINIO_USE_SSL = true
MINIO_ENDPOINT = minio.example.com
MINIO_ACCESS_KEY_ID = [redacted]
MINIO_SECRET_ACCESS_KEY = [redacted]
MINIO_BUCKET = bucket
MINIO_LOCATION = us-east-1
```

Storage settings for a single subsystem

It is possible to configure some subsystems to use S3 storage and others to use local storage by adding settings to their respective sections. For instance:

```
[attachment]
PATH = /otherstorage/attachments

[lfs]
STORAGE_TYPE = minio
MINIO_BASE_PATH = lfs/

MINIO_ENDPOINT = 127.0.0.1:9000
MINIO_ACCESS_KEY_ID = [redacted]
MINIO_SECRET_ACCESS_KEY = [redacted]
MINIO_BUCKET = forgejo
MINIO_LOCATION = us-east-1
```

will store attachments in the local directory `/otherstorage/attachments` while `lfs` files will be stored in the S3 server within the `lfs/` directory of the `forgejo` bucket.

Storage settings

The value of `STORAGE_TYPE` can be `local` (the default) for file system directories or `minio` for S3 servers. Each storage type has its own settings, as explained below.

Using local

There is just one setting when the `STORAGE_TYPE` is set to `local`: `PATH`. It must be an absolute path and is interpreted as follows.

In the `[storage]` section, `PATH` is the path under which the directories of each subsystem will be created instead of `APP_DATA_PATH`. For instance, if `APP_DATA_PATH` equals `/appdata`

```
[storage]
STORAGE_TYPE = local
PATH = /mystorage
```

will create attachments in `/mystorage/attachments` instead of `/appdata/attachments`, LFS files in `/mystorage/lfs` instead of `/appdata/lfs`, etc.

In the section dedicated to a subsystem (see the table in the introduction), `PATH` is the base path under which all files will be stored. For instance:

```
[storage]
STORAGE_TYPE = local
PATH = /mystorage

[attachment]
STORAGE_TYPE = local
PATH = /otherstorage/attachments
```

will store attachments in `/otherstorage/attachments` while `lfs` files will be stored in `/mystorage/lfs`.

Using minio

When the `STORAGE_TYPE` is set to `minio`, the settings are used to connect to a S3 compatible server:

- `SERVE_DIRECT`: **false**: Allows the storage driver to redirect to authenticated URLs to serve files directly. Only supported via signed URLs.
- `MINIO_ENDPOINT`: **localhost:9000**: S3 endpoint to connect.
- `MINIO_ACCESS_KEY_ID`: S3 accessKeyID to connect.
- `MINIO_SECRET_ACCESS_KEY`: S3 secretAccessKey to connect.
- `MINIO_BUCKET`: **forgejo**: S3 bucket to store the data.
- `MINIO_BUCKET_LOOKUP`: **auto**: S3 bucket lookup type.
 - `auto` Auto detected
 - `dns` Virtual Host style
 - `path` Path Style
- `MINIO_LOCATION`: **us-east-1**: S3 location to create bucket.
- `MINIO_USE_SSL`: **false**: S3 enabled ssl.
- `MINIO_INSECURE_SKIP_VERIFY`: **false**: S3 skip SSL verification.
- `MINIO_CHECKSUM_ALGORITHM`: Minio checksum algorithm: **default** (for MinIO, garage or AWS S3) or **md5** (for Cloudflare or Backblaze).

When used in the `[storage]` section they apply to all subsystems. When used in the section specific to a subsystem (see the table in the introduction), they are only used for objects that belong to this subsystem. Here is a example:

```
[storage]
STORAGE_TYPE = minio

SERVE_DIRECT = false
MINIO_ENDPOINT = garage:9000
MINIO_ACCESS_KEY_ID = [redacted]
MINIO_SECRET_ACCESS_KEY = [redacted]
```

```

MINIO_BUCKET = forgejo
MINIO_BUCKET_LOOKUP = auto
MINIO_LOCATION = us-east-1
MINIO_USE_SSL = false
MINIO_INSECURE_SKIP_VERIFY = false
MINIO_CHECKSUM_ALGORITHM = md5

[-lfs]
STORAGE_TYPE = minio
MINIO_BASE_PATH = nonstandardlfs/

SERVE_DIRECT = false
MINIO_ENDPOINT = minio:9000
MINIO_ACCESS_KEY_ID = [redacted]
MINIO_SECRET_ACCESS_KEY = [redacted]
MINIO_BUCKET = forgejo
MINIO_BUCKET_LOOKUP = auto
MINIO_LOCATION = us-east-1
MINIO_USE_SSL = false
MINIO_INSECURE_SKIP_VERIFY = false

```

- `MINIO_BASE_PATH`: **only valid in the specific subsystem section (see the table in the introduction)** overrides the default directory in which objects are stored in the `MINIO_BUCKET` bucket.

For all subsystems that use the `minio` storage type found in the `[storage]` section, the directory in which the objects are stored is determined using the table in the introduction. For instance LFS files will be stored in the `lfs/` directory within the `forgejo` bucket.

When the `minio` storage is set in a section specific to a subsystem, the `MINIO_BASE_PATH` setting can be used to override the default directory. In the example above, `MINIO_BASE_PATH = nonstandardlfs/` means LFS objects will be stored in the `nonstandardlfs/` directory within the `forgejo` bucket instead of the `lfs/` directory

S3 servers compatibility

Although the S3 storage type is named `minio` it does not rely on any [MinIO](#) specific features. The S3 storage type is [tested](#) to be compatible with:

- [MinIO](#) 2021.3.17 and 2023-08-23
- [garage](#) v0.8.2

Undocumented features

It is **strongly** recommended to avoid using undocumented features - such as `[storage.attachments]` as an alternative to `[attachment]` for instance (the plural is not a typo, it is a unification problem) - because their behavior is not thoroughly tested and may lead to unexpected results.

Legacy settings

Some settings are deprecated but still supported in the interest of backward compatibility. They should be replaced as follows:

- `[server].LFS_CONTENT_PATH` is replaced with `[lfs].PATH`
- `[picture].AVATAR_UPLOAD_PATH` is replaced with `[avatar].PATH`
- `[picture].REPOSITORY_AVATAR_UPLOAD_PATH` is replaced with `[repo-avatar].PATH`

Legacy settings have a lower priority and will be overridden by their replacement if both are present. For instance:

```
[picture]
AVATAR_UPLOAD_PATH = /legacy_path

[avatar]
PATH = /avatar_path
```

will store avatar files in `/avatar_path`.

☐☐ #1
☐ cocobook-sync ☐☐ 2026-10-28 16:41:03 UTC
☐ cocobook-sync ☐☐ 2026-10-28 16:41:03 UTC