

Deploying Your Server Using Git (Optional - Advanced Feature)

Your server supports automatic deployment from a Git repository. When configured, the server will pull the latest version of the repository every time the server starts.

This allows you to manage your server files using version control and update your server by pushing changes to your repository.

When Git deployment is enabled, local file changes inside the Git target folder will be overwritten when the server starts.

How Git Deployment Works

If a repository URL is configured, the server will perform the following steps during startup:

1. Connect to the configured Git repository
2. Pull the selected branch
3. Overwrite the contents of the Git target folder
4. Start the server using the updated files

This ensures the server always runs the latest version of your repository.

Configuring Git Deployment

Git deployment is configured using the startup variables in the server panel.

Git Repository URL

The URL of the repository that contains your server files.

Example:

```
https://github.com/example/my-fivem-server.git
```

Git Branch

The branch that should be deployed.

Default:

Git Target Path

The folder where the repository will be deployed.

Default:

Your `server.cfg` and resources should normally be located inside this folder when using Git deployment.

Git Username and Git Token

These fields are used when accessing private repositories.

Enter your Git username and a personal access token with repository access.

These values are not required for public repositories.

Typical Workflow

Once Git deployment is configured, updating your server becomes very simple:

1. Make changes to your server files locally
2. Commit and push the changes to your repository
3. Restart the server

On startup, the server will pull the latest version of the repository and update the server files automatically.

Example Repository Layout

```
repository-root
├─ server.cfg
├─ resources/
│   └─ resource_one/
│   └─ resource_two/
│   └─ resource_three/
```

Common Issues

If Git deployment fails, check the following:

- The repository URL is correct
- The branch name exists
- Git credentials are correct when using a private repository
- The server has permission to access the repository

If Git deployment fails, the server may not start if required files such as `server.cfg` are missing.

Revision #2

Created 2026-03-11 16:23:46 UTC by Kentbakk

Updated 2026-03-11 17:17:42 UTC by Kentbakk