

# SSH Alias anlegen

Unter Windows kannst du für SSH-Server-Aliase die Datei `~/.ssh/config` verwenden – genauso wie unter Linux oder macOS. Damit kannst du z. B. einem GitLab-Server auf einem speziellen Port (wie 2022) einen Alias geben.

## So richtest du den Alias ein:

1. Öffne den Windows-Dateiexplorer und gehe zu deinem Benutzerverzeichnis:

```
C:\Users\<<deinBenutzername>\.ssh
```

Falls der Ordner `.ssh` noch nicht existiert, kannst du ihn anlegen.

2. Öffne (oder erstelle) darin die Datei `config` (ohne Dateierweiterung) mit einem Texteditor wie Notepad oder VSCode.
3. Füge folgenden Block hinzu (Passe die Werte an deine Umgebung an):

```
Host gitlab
  HostName gitlab.example.com
  Port 2022
  User git
  IdentityFile ~/.ssh/id_rsa
```

- **Host:** ist der Alias, den du später im `ssh`- oder `git`-Befehl verwenden kannst (z. B. `gitlab`).
  - **HostName:** ist der tatsächliche Hostname oder die IP-Adresse deines GitLab-Servers.
  - **Port:** der Port, auf dem der SSH-Server läuft (hier 2022).
  - **User:** der Benutzername für den Login (bei GitLab i.d.R. `git`).
  - **IdentityFile:** optional, gibt den privaten SSH-Schlüssel an, falls du nicht den Standardpfad verwendest.
4. Speichere die Datei.

## Beispiel-Verwendung:

Statt so:

```
ssh -p 2022 git@gitlab.example.com
```

einfach so:

```
ssh gitlab
```

Oder beim Klonen eines Repositories:

```
git clone git@gitlab:username/repo.git
```

## Tipp

Wenn dein SSH-Schlüssel einen anderen Namen hat (z. B. `id_gitlab_rsa`), gib ihn explizit bei `IdentityFile` an:

```
IdentityFile ~/.ssh/id_gitlab_rsa
```

## Hinweis zu Pfadangaben unter Windows

`~` funktioniert normalerweise auch unter Windows, aber wenn du Probleme hast, ersetze `~` durch den vollständigen Pfad, z. B.:

```
IdentityFile C:/Users/DeinBenutzername/.ssh/id_rsa
```

Möchtest du, dass ich ein konkretes Beispiel mit deinen Daten schreibe?

---

Revision #1

Created 9 June 2025 07:14:17 by Stefan Mechler

Updated 9 June 2025 07:14:35 by Stefan Mechler