

AUR-Pakete veröffentlichen

Möchte man für das [AUR](#) Pakete veröffentlichen, benötigt man zunächst folgende Dinge:

1. [AUR](#)-Account
2. SSH-Schlüsselpaar

Das anlegen eines Accounts ist selbsterklärend. Daher gehe ich hier nicht weiter darauf ein.

SSH-Schlüsselpaar erzeugen

Dieses wird von ChatGPT wie folgt beschrieben:

```
ssh-keygen -t ed25519 -C "deinname@deinrechner"
```

- `-t ed25519` → moderner, sicherer Schlüsseltyp
- `-C "Kommentar"` → nur ein Hinweis (meist `Benutzername@Rechnername`), optional

Dann kommt:

```
Enter file in which to save the key (/home/thomas/.ssh/id_ed25519):
```

- Standard ist `~/.ssh/id_ed25519`.
- Falls du schon einen Key dort hast, kannst du ENTER drücken (überschreibt nicht automatisch), oder einen neuen Namen vergeben wie `~/.ssh/id_ed25519_aur`.

Dann wirst du nach einer **Passphrase** gefragt – das ist optional, aber empfohlen. Wenn du keine willst, einfach ENTER drücken.

Danach liegen zwei Dateien in `~/.ssh/`:

- **privater Schlüssel:** `id_ed25519`
(gut aufbewahren, darf nie hochgeladen werden)
- **öffentlicher Schlüssel:** `id_ed25519.pub`
(den musst du ins AUR eintragen)

Inhalt der Datei anzeigen:

```
cat ~/.ssh/id_ed25519.pub
```

- Den kompletten Inhalt (eine Zeile, beginnt mit `ssh-ed25519 ...`) kopieren.

- Auf aur.archlinux.org einloggen → **My Account** → **SSH Public Keys** → einfügen → speichern.

Jetzt prüfen, ob die Verbindung klappt:

```
ssh -T aur@aur.archlinux.org
```

Wenn alles passt, kommt so etwas wie:

```
Welcome to the AUR, <AUR-Benutzername>! Interactive shell is disabled.
```

Paket anpassen und veröffentlichen

Zunächst auf der Ordnerstruktur im persönlichen Ordner ein geeignetes Verzeichnis anlagen und darein wechseln und ein Terminal öffnen.

Dann folgendes ausführen um den aktuelle Stand des Paketes in den Ordner zu laden. Hier am Beispiel des Pakets "cockpit-navigator":

```
git clone ssh://aur@aur.archlinux.org/cockpit-navigator.git
```

Es entsteht der Ordner "cockpit-navigator". Darein wechseln:

```
cd cockpit-navigator
```

Darin befindet sich u.a. die Datei PKGBUILD. Diese mit einem Editor nach Bedarf anpassen. U.a. muss sich bei jeder Änderung mindestens die Paketversion ändern.

Hat man alle Änderungen durchgeführt und die Datei gespeichert, muss man die Checksumme(n) anpassen. Dazu im Terminal folgenden Befehl ausführen:

```
updpkgsums
```

Dabei wird auch der Sourcecode runtergeladen, damit die Checksumme gebildet werden kann.

Dann sollte die Datei .SRCINFO neu erstellt werden:

```
makepkg --printsrcinfo > .SRCINFO
```

Nun sollte die Erstellung und Installation des Paketes getestet werden:

```
makepkg -si
```

Läuft alles fehlerfrei durch, müssen jetzt noch die Änderungen dokumentiert und hochgeladen werden:

```
git add PKGBUILD .SRCINFO
git commit -m "einen passenden Kommentar hinterlegen"

git push origin master
```

Jetzt wird man noch mal nach dem Passwort zu seinem SSH-Schlüssel gefragt. Nach der Eingabe erfolgt das Hochladen der Änderung und die Veröffentlichung.

Revision #1

Created 2025-09-21 08:53:22 UTC by Thomas Krueger

Updated 2025-09-21 09:28:29 UTC by Thomas Krueger