

Linux Setup

Chatlix auf Linux. Status: experimentell, manuelle Installation.

- [Status auf Linux](#)
- [Manuelle Installation](#)
- [Systemd-Unit \(optional\)](#)

Status auf Linux

Status auf Linux

Das Connect-Binary fuer Linux heisst `chatlix-connect-linux`. Es ist statisch gelinkt (x64, aktuell 0.4.0) und braucht keine zusaetzlichen Libraries. Du startest es aus einem Terminal oder als User-Systemd-Service.

Status pruefen

Das Binary bringt ein CLI-Tool fuer den Status mit. Ein Aufruf ohne Argumente zeigt einen einzeiligen Status in der Konsole, mit `status` bekommst du die Details:

```
./chatlix-connect-linux status
```

Die Ausgabe enthaelt:

- **Pairing-Status** — `paired` mit Account-Slug, oder `unpaired`. Wenn `unpaired`, lauft erst `pair`.
- **WS-Status** — Verbindung zu `wss://ws.chatlix.app/connect`. Erwartet: `connected`. Wenn `connecting...` mehr als ein paar Sekunden anhaelt, pruefe DNS und ausgehende Firewall.
- **OBS-Verbindung** — falls in der Config aktiviert, sieht man hier `obs: connected` (`host:port`) oder `obs: disabled`. Bei Fehler steht der konkrete Fehlertext (z.B. `auth failed`, `connection refused`).
- **Game-Watcher** — `running` / `idle` / `not_configured`. Game-Watcher ist nur bei aktivem Game-Watcher-Setup interessant.
- **Letzter Heartbeat** — Sekunden seit letztem Ping. Sollte dauerhaft < 30s sein.

Live-Status im Dashboard

Der gleiche Status wird parallel an das Dashboard gemeldet und ist dort unter `/dashboard/connect` sichtbar. Das ist die Quelle der Wahrheit fuer Rewards, die Connect brauchen — wenn das Dashboard `offline` zeigt, helfen lokale `running`-Meldungen nichts.

Logs einsehen

Laufende Logs liegen unter:

```
~/.local/share/chatlix-connect/log.txt
```

Das File rotiert ab ~5 MB. Bei Problemen ist der Anfang einer Sitzung interessant (Pairing-Load, WS-Handshake), bei Reward-Fehlern das Ende.

Im Vordergrund laufen lassen

Fuer kurze Sessions reicht es, das Binary in einem Terminal-Tab laufen zu lassen. Beendet wird mit `Ctrl+C`. Fuer dauerhaften Betrieb solltest du den User-Systemd-Service nutzen (siehe Seite « Systemd-Unit (optional) »).

Schnelltest

Wenn du nicht sicher bist, ob Connect korrekt arbeitet:

1. Connect im Terminal starten (Vordergrund).
2. Im Dashboard unter `/dashboard/connect` pruefen, dass die Anzeige auf `online` springt (< 5s).

3. Im Dashboard unter `/dashboard/actions` eine Test-Aktion auslösen (z.B. eine OBS-Source togglen).
4. Terminal-Log und Status-Ausgabe prüfen — die Aktion sollte als Event durchlaufen.

Wenn das funktioniert, ist die Basis-Verbindung sauber. Reward-spezifische Probleme sind dann eine Ebene darüber.

Manuelle Installation

Manuelle Installation

Fuer Linux gibt es kein Installer-Paket. Du laedst das Binary, machst es ausfuehrbar und pair-st den Client.

Voraussetzungen

- x86_64 Linux (alle gaengigen Distros).
- `curl` oder `wget`.
- Ausgehender Zugriff auf `chatlix.app` (HTTPS 443) und `ws.chatlix.app` (WSS 443).
- Optional: OBS WebSocket auf localhost, wenn du OBS-Actions nutzen willst.

Glibc-Abhaengigkeiten gibt es keine relevanten — das Binary ist statisch gelinkt.

Download und Installation

```
curl -O https://chatlix.app/downloads/chatlix-connect-linux
chmod +x chatlix-connect-linux
./chatlix-connect-linux pair
```

Der `pair`-Command zeigt einen 6-stelligen Code und eine URL. Du oeffnest die URL im Browser (im selben Account eingeloggt) und bestaetigst den Code. Sobald die Pairing-Bestaetigung durch ist, schreibt das Binary die Config nach:

```
~/.config/chatlix-connect/config.json
```

Darin steht ein Pairing-Token. **Diesen Token nicht weitergeben** — er identifiziert deinen Channel gegenüber Chatlix.

Empfohlene Ablage

Wenn du Connect dauerhaft nutzen willst, leg das Binary nach `~/bin`:

```
mkdir -p ~/bin
mv chatlix-connect-linux ~/bin/
```

Danach laeuft es per `~/bin/chatlix-connect-linux` aus jedem Verzeichnis. Stell sicher, dass `~/bin` in deinem `PATH` ist (`echo $PATH`). Andernfalls passt du das in `~/.bashrc` / `~/.zshrc` an.

Erster Start

Nach dem Pairing startest du Connect ohne weitere Argumente:

```
~/bin/chatlix-connect-linux
```

Du solltest in der Konsole sehen:

```
[INFO] config loaded
[INFO] ws connected to wss://ws.chatlix.app/connect
[INFO] paired as <slug>
```

Im Dashboard unter `/dashboard/connect` springt der Status innerhalb weniger Sekunden auf `online`.

Update

Updates laufen ueber denselben Download. Stoppe Connect (`Ctrl+C` oder `systemctl --user stop chatlix-connect`), tausche die Datei und starte neu. Die Config bleibt erhalten, ein neues Pairing ist nicht noetig.

```
curl -O https://chatlix.app/downloads/chatlix-connect-linux
chmod +x chatlix-connect-linux
mv chatlix-connect-linux ~/bin/
```

Game-Watcher auf Linux

Der Game-Watcher fuer Linux wird als Source-Tarball ausgeliefert, nicht als Binary:

```
curl -O https://chatlix.app/downloads/chatlix-game-watcher-source.tar.gz
tar xzf chatlix-game-watcher-source.tar.gz
cd chatlix-game-watcher
npm install
npm run start
```

Voraussetzung: Node.js \geq 20. Beim ersten Start fragt der Watcher dieselben Pairing-Daten ab wie Connect; danach laeuft er parallel und meldet erkannte Game-Sessions an deinen Channel.

Sauber deinstallieren

```
systemctl --user disable --now chatlix-connect 2>/dev/null
rm -f ~/bin/chatlix-connect-linux
rm -rf ~/.config/chatlix-connect ~/.local/share/chatlix-connect
```

Das Pairing kannst du zusaetzlich im Dashboard unter `/dashboard/connect` revoked — danach ist der alte Token serverseitig ungueltig.

Systemd-Unit (optional)

Systemd-Unit (optional)

Wenn Connect bei jedem Login automatisch starten soll, leg eine User-Systemd-Unit an. Das laeuft im User-Scope (kein root noetig) und ueberlebt Logout/Login, wenn `loginctl enable-linger` aktiv ist.

Unit-Datei

Datei: `~/.config/systemd/user/chatlix-connect.service`

```
[Unit]
Description=Chatlix Connect
After=network-online.target

[Service]
Type=simple
ExecStart=%h/bin/chatlix-connect-linux
Restart=on-failure
RestartSec=5s

[Install]
WantedBy=default.target
```

`%h` ist der Home-Pfad des Users. Wenn dein Binary woanders liegt, passe `ExecStart=` an.

Aktivieren

```
systemctl --user daemon-reload
systemctl --user enable --now chatlix-connect
systemctl --user status chatlix-connect
```

Die Status-Ausgabe sollte `active (running)` zeigen und im Log `ws connected` enthalten.

Logs ansehen

```
journalctl --user -u chatlix-connect -f
```

Das zeigt einen Live-Stream. Mit `-n 200` siehst du die letzten 200 Zeilen, mit `--since '1 hour ago'` einen Zeitfilter.

Die ausführlichen Logs liegen zusätzlich in `~/.local/share/chatlix-connect/log.txt` — `journalctl` enthält nur stdout/stderr.

Auto-Start ohne aktive Session

Standardmaessig laufen User-Units nur, solange du eingeloggt bist (TTY oder GUI). Wenn Connect auch ohne aktive Session laufen soll (z.B. auf einer Streaming-Maschine, die headless bootet):

```
sudo loginctl enable-linger $USER
```

Danach startet der User-Service mit dem Boot. Pruefen kannst du das mit:

```
loginctl show-user $USER | grep Linger
```

Erwartet: `Linger=yes`.

Restart und Stop

```
systemctl --user restart chatlix-connect
systemctl --user stop chatlix-connect
systemctl --user disable chatlix-connect
```

Nach einem Update des Binary reicht `restart` — die Config wird neu eingelesen, das Pairing bleibt bestehen.

Typische Fallstricke

- **Failed to start chatlix-connect.service: Unit not found** — User-Bus nicht aktiv. Lauft `systemctl --user daemon-reload` und stellt sicher, dass du als regulärer User eingeloggt bist (nicht via `sudo su -`).
- **Restart-Loop** — Connect crasht direkt nach Start. In den Logs steht meistens `pairing token invalid` (Token serverseitig revoked) oder `cannot reach ws.chatlix.app` (DNS/Firewall). Erst Ursache fixen, dann Service wieder enablen.
- **Mehrere Sessions** — Wenn du Connect parallel in einer Shell startest während die Unit laeuft, klappt nur eine WS-Verbindung gleichzeitig. Beende die Shell-Instanz und nutze die Unit.
- **PATH-Variablen** — Systemd-User-Units erben deinen Login-Shell-PATH nicht. Wenn dein Binary z.B. `~/ .nvm/...` braucht, schreibe die voll qualifizierten Pfade in die Unit.

Cleanup

```
systemctl --user disable --now chatlix-connect
rm ~/.config/systemd/user/chatlix-connect.service
systemctl --user daemon-reload
```

Das Binary und die Pairing-Config bleiben unangetastet — die Unit ist nur der Wrapper.