zorel / **roinroin**

Code · Issues · Pull requests · Actions · Projects · Security and quality

Insights

roinroin Public

Notifications · Fork 0 · Star 0

main · Go to file · Code

zorel Release v0.5.0 ✓		11abe0b · 23 minutes ago
.github/workflows	Release v0.5.0	23 minutes ago
assets	Release v0.5.0	23 minutes ago
src	Release v0.5.0	23 minutes ago
.gitignore	Initial commit: native ...	3 weeks ago
CHANGELOG.md	Release v0.5.0	23 minutes ago
Cargo.lock	Release v0.5.0	23 minutes ago
Cargo.toml	Release v0.5.0	23 minutes ago
LICENSE	Initial commit: native ...	3 weeks ago
Makefile	Initial commit: native ...	3 weeks ago
README.md	Release v0.5.0	23 minutes ago
roinroin.example.toml	Release v0.5.0	23 minutes ago
screenshot.png	Add timeline text sele...	2 weeks ago

About

Native Linux tribune client for gb2c and direct bouchots

Readme

MIT license

Activity

0 stars

0 watching

0 forks

Report repository

Releases 6

v0.5.0 Latest

17 minutes ago

+ 5 releases

Contributors 1

zorel Aurélien DEHAY

Languages

Rust 99.8% · Makefile 0.2%

README · MIT license

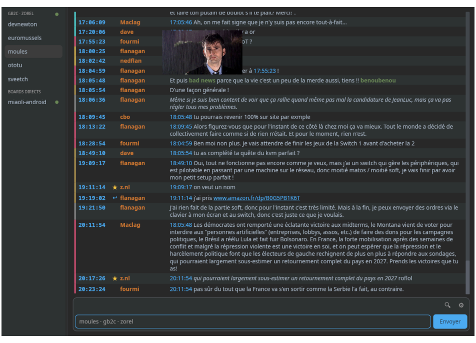
roinroin

Client de tribune **natif pour Linux**, écrit en Rust avec iced. Il parle à la fois :

- à un ou plusieurs serveurs **gb2c** (API HTTP + flux SSE `/gb2c/poll`) ;
- à des **boards bouchot directs**, sans serveur intermédiaire (backend TSV ou XML + POST de formulaire), y compris LinuxFR avec OAuth2.

Les deux types de sources peuvent être utilisés **en même temps**. Une même tribune peut être suivie via gb2c et via son board direct ; elle apparaît alors comme deux entrées distinctes, avec leur propre état de connexion, leurs propres posts et leurs propres compteurs de non-lus.

Le client suit le standard tribune / bouchot et reprend les fonctions des clients existants `gb3tui` , `gb3-android` et `wmcoincoin` , avec une intégration Linux soignée.



17:06:09 Maclag 17:05:46 Ah, on me fait signe que je n'y suis pas encore tout-à-fait...

17:20:06 dave a or

17:55:23 fourmi oT ?

18:00:25 flanagan

18:02:42 nedflan

18:04:59 flanagan er à 17:55:23 !

18:05:48 flanagan Et puis bad news parce que la vie c'est un peu de la merde aussi, tiens !! benoubenou

18:05:54 flanagan D'une façon générale !

18:06:36 flanagan *Même si je suis bien content de voir que ça rallie quand même pas mal la candidature de JeanLuc, mais ça va pas régler tous mes problèmes.*

18:09:45 cbo 18:05:48 tu pourrais revenir 100% sur site par exmple

18:13:22 flanagan 18:09:45 Alors figurez-vous que pour l'instant de ce côté là chez moi ça va mieux. Tout le monde a décidé de collectivement faire comme si de rien n'était. Et pour le moment, rien n'est.

18:28:54 fourmi 18:04:59 Ben moi non plus. Je vais attendre de finir les jeux de la Switch 1 avant d'acheter la 2

18:49:10 dave 18:05:54 tu as complété ta quête du kvm parfait ?

19:09:17 flanagan 18:49:10 Oui, tout ne fonctionne pas encore comme je veux, mais j'ai un switch qui gère les périphériques, qui est pilotable en passant par une machine sur le réseau, donc moitié matos / moitié soft, je vais finir par avoir mon petit setup parfait !

19:11:14 ★ z.nl 19:09:17 on veut un nom

19:19:02 ↩ flanagan 19:11:14 j'ai pris www.amazon.fr/dp/B0G5PB1K6T

19:21:50 flanagan J'ai rien fait de la partie soft, donc pour l'instant c'est très limité. Mais à la fin, je peux envoyer des ordres via le clavier à mon écran et au switch, donc c'est juste ce que je voulais.

20:11:54 Maclag 18:05:48 Les démocrates ont remporté une éclatante victoire aux midterms, le Montana vient de voter pour interdire aux "personnes artificielles" (entreprises, lobbys, assos, etc.) de faire des dons pour les campagnes politiques, le Brésil a réélu Lula et fait fuir Bolsonaro. En France, la forte mobilisation après des semaines de conflit et malgré la répression violente est une victoire en soi, et on peut espérer que la répression et le harcèlement politique font que les électeurs de gauche rechignent de plus en plus à répondre aux sondages, qui pourraient largement sous-estimer un retournement complet du pays en 2027. Prends les victoires que tu as!

20:17:26 ★ z.nl 20:11:54 *qui pourraient largement sous-estimer un retournement complet du pays en 2027* roflol

20:23:24 fourmi 20:11:54 pas sûr du tout que la France va s'en sortir comme la Serbie l'a fait, au contraire.

Aperçu des fonctions

Sources et réseau

- flux **SSE** gb2c en direct, avec reconnexion automatique, backoff, surveillance d'inactivité et filtrage des posts déjà reçus lors des rejeux ;
- liste des tribunes et identités bloquées découvertes via `/gb2c/list` et `/gb2c/config` ;
- boards directs **TSV** et **XML** (`<board><post>…`) tolérants aux réponses malformées, avec limites de taille et nettoyage des caractères de contrôle ;
- polling direct toutes les 30 secondes, en parallèle pour chaque board ;
- envoi vers `/gb2c/post` ou directement vers l'URL de post du board ;
- OAuth2 LinuxFR : authentification directe depuis le client (redirection native, code collé), ou via gb2c si aucun identifiant d'application n'est configuré ; rafraîchissement automatique du jeton par la même voie ;
- plusieurs serveurs gb2c configurables ;
- pastilles d'état à droite de chaque serveur et board : vert connecté, bleu en cours de connexion/rafraîchissement/envoi, rouge en erreur.

Lecture des posts

- résolution complète des **norloges** : `#42` , ISO `2017-06-29T13:45:01.421` , `2017-06-29 13:45:01` , `06/29#13:45:01` , `13:45:01` , `13:45` , suffixes `^2` / `²` et `@bouchot` ;
- graphe de conversation : racine, réponses directes et indirectes, couleur de fil stable, compteur de réponses ; une horloge courte (`13:45`) qui vise l'un de nos propres posts compte comme une réponse ;
- boitakon (**bàk**) : filtrage des identités bloquées ; l'option `hungry` filtre aussi les réponses, sinon seuls les posts de l'identité bloquée sont filtrés ;
- affichage des posts filtrés au choix : `gray` (la ligne reste visible, son texte est remplacé par `[message masqué]`) ou `hide` (la ligne disparaît) ;
- mise en page compacte sur une seule ligne, comme gb3-android : horloge, auteur et message partagent la même taille de police ;
- survoler l'horloge d'un post met en évidence ce post et les norloges qui le référencent ; survoler une norloge dans un message met en évidence le post référencé (dans les deux sens) ; si le post référencé est hors du viewport, un aperçu flottant l'affiche en haut de la tribune, formaté comme un post normal ; l'aperçu reste ouvert le temps d'y amener le curseur ;
- colonne de posts alignée : horloge, statut et identité ont des largeurs fixes (identité tronquée à 16 caractères), le message commence toujours à la même position ; si un login existe c'est lui qui est affiché, et son info/UA

apparaît au survol ; sinon l'info courte est affichée ;

- références inverses : sélectionner un post met en évidence ses réponses ;
- `Maj+clic droit` sur n'importe quel post ouvre tout son arbre de discussion dans un panneau à droite : racine, réponses, réponses aux réponses et toutes les branches ; le panneau suit les nouveaux messages et se scinde en deux colonnes quand un autre arbre est ouvert, avec les mêmes filtres que la tribune ;
- BML : `<b>` , `<i>` , `<u>` , `<s>` , `<tt>` , `<code>` , spoilers cliquables, liens bruts, liens `<a>` , entités HTML, bigornos `nick<` ;
- totoz affichés par leur nom en vert ; au survol, l’image ou l’animation apparaît à sa taille d’origine 5 pixels au-dessus du nom, sans cadre, et reste immobile tant que le pointeur demeure sur le totoz ;
- liens : le titre des pages HTML et un aperçu des images sont récupérés à l’arrivée des posts ; au survol du lien, ils apparaissent 5 pixels au-dessus du texte du lien, une petite croix rouge signalant un contenu inaccessible ou protégé ;
- horloge des posts au format `HH:MM:SS` , colonne fixe ;
- détection des canards `\_o<` / `>o_/` ;
- listes **plop** : les messages des auteurs choisis sont remplacés par `pika` , `plop` , `prout` , `grouik` ou `meuh` ;
- compteurs de non-lus et badges bigorno/réponse dans le panneau des tribunes ;
- sélection du texte des posts à la souris dans le panneau principal, avec surlignage et copie par `Ctrl+C` ; la sélection peut traverser plusieurs posts ; un lien sélectionné (ou cliqué à droite) est copié avec son URL complète plutôt qu'avec son libellé ;
- cloche horaire optionnelle (`hourly_bell`).

Écriture

- éditeur sur une seule ligne avec brouillon persistant, bouton « Envoyer » à droite ; le nom de la tribune et sa source servent d'indication dans le champ vide ;
- raccourcis `Alt+…` pour insérer les balises de formatage : le curseur est placé entre les deux balises, ou le texte sélectionné est encadré par celles-ci ; `Alt+M` insère un moment, en ajoutant la sélection après le mot « Moment » le cas échéant ;
- fermeture automatique des balises : taper `<b>` , `<i>` , `<u>` , `<s>` , `<tt>` , `<code>` ou `<spoiler>` insère aussitôt la balise fermante et laisse le curseur entre les deux ;
- citation d'un post au clic sur son horloge, insérée à la position du curseur, avec bascule automatique du suffixe `@tribune` quand on change de tribune ;
- envoi par bouton, `Entrée` ou `Ctrl+Entrée` , restauration du brouillon si l'envoi échoue ;
- insertion des totoz et emojis dans le brouillon à la position du curseur ; choisir un totoz dans la recherche revient à la tribune, l'insère avec un espace et place le curseur juste après ;
- autocomplétion des totoz directement dans l'éditeur : dès `[:` suivi de deux caractères, les totoz correspondants s'affichent juste au-dessus du champ avec un aperçu miniature (aucune liste s'il n'y a pas de correspondance) ; `↑` / `↓` pour changer de suggestion, `Entrée` ou `Tab` pour l'insérer, `Échap` pour fermer la liste ; le curseur reste dans le champ ;
- le focus et le curseur sont placés dans l'éditeur au démarrage, à la sélection d'une tribune et au retour sur celle-ci, ainsi que dans le champ de recherche à l'ouverture des écrans ToToz et Emoji.

Intégration Linux

- **notifications de bureau** freedesktop.org via D-Bus (`notify-rust`) : bigorno, réponses à mes posts, ou tous les nouveaux posts ;
- notification seulement quand la fenêtre n'est pas au premier plan (option, les bigornos sont toujours notifiés) ;
- demande d'attention de la fenêtre pour les bigornos/réponses ;
- fichier `.desktop` , icône hicolor et `make install` ;
- thème sombre (palette gc2/gb3), clair ou détection système ;
- aucune WebView, aucun JavaScript : toute l'interface est native iced ;
- rendu `wgpu` (Vulkan/OpenGL) quand il est disponible, avec repli logiciel `tiny-skia` sinon ; X11/Wayland requis.

Compilation

Prérequis :

- Rust **1.88 ou plus** (`rustup` recommandé) ;
- un environnement graphique X11 ou Wayland ;
- aucune police emoji système n'est nécessaire : une police couleur compatible (Twemoji Mozilla, COLRv0) est embarquée dans le binaire ;
- éventuellement `libxkbcommon` et les bibliothèques Wayland/X11 de la distribution (winit les utilise) ;
- un démon de notifications pour l'intégration bureau (GNOME, KDE, mako, dunst…).

Sur Debian/Ubuntu, les en-têtes utiles sont :

```
sudo apt install build-essential pkg-config libxl
    libwayland-dev libx11-dev
```

`notify-rust` parle D-Bus directement (zbus), il n'y a pas besoin de la bibliothèque C libdbus.

```
cargo build --release
./target/release/roinroin
```

Options utiles :

```
roinroin --gb2c http://localhost:16666 --nicknam
roinroin --no-notifications
roinroin --debug-http
roinroin --profile
roinroin --help
```

`--profile` (ou `ROINROIN_PROFILE=1`) affiche sur stderr, toutes les cinq secondes, le nombre et le coût des mises à jour, des rendus et des analyses de conversations. Les opérations anormalement lentes sont aussi journalisées, ce qui permet d'identifier un pic CPU sans profiler externe.

Ou :

```
make
```

```
make run
```

Tests :

```
cargo test
```

Téléchargements

Les [releases GitHub](#) fournissent des archives précompilées pour Linux :

- `x86_64-unknown-linux-gnu` (PC 64 bits) ;
- `aarch64-unknown-linux-gnu` (ARM 64 bits, Raspberry Pi 64 bits).

Chaque archive contient le binaire `roinroin`, le fichier `.desktop`, l'icône et la licence.

Installation dans `~/.local` :

```
make install
# binaire   : ~/.local/bin/roinroin
# .desktop  : ~/.local/share/applications/roinro
# icône     : ~/.local/share/icons/hicolor/scala
```

`make install PREFIX=/usr/local` installe ailleurs (avec les droits adaptés).

Configuration

Au premier lancement, roinroin crée :

- `${XDG_CONFIG_HOME:-~/.config}/roinroin/config.toml`
- `${XDG_DATA_HOME:-~/.local/share}/roinroin/state.json`

Un exemple commenté est fourni : [`roinroin.example.toml`](#).

Les champs importants :

Champ	Rôle
`nickname`	pseudo envoyé avec les posts et utilisé pour les bigornos/réponses
`servers`	liste de serveurs gb2c (`name`, `url`, `enabled`)
`selected_boards`	boards directs actifs (prédéfinis ou personnalisés)
`hidden_tribunes`	tribunes gb2c masquées (clé `gb2c:URL|nom`)
`custom_boards`	boards directs supplémentaires
`blocked`	identités masquées (login ou info)
`blocked_mode`	posts filtrés : `gray` (ligne conservée, texte masqué) ou `hide` (ligne absente)
`hungry`	la bàk filtre aussi les réponses aux posts filtrés (`true` par défaut)
`plop_list`	auteurs dont le message est remplacé par un plop
`totoz_server`	serveur totoz (`https://nsfw.totoz.eu` par défaut)
`notifications`	activation, bigorno, réponses, tous les posts, focus
`history_limit`	nombre de posts conservés par tribune (2000 par défaut)
`hourly_bell`	notification à chaque heure pile
`theme`	`dark`, `light` ou `system`
`language`	langue de l'interface (`french` pour l'instant)
`linuxfr_client_id`	identifiant de l'application OAuth LinuxFR (connexion directe sans gb2c)
`linuxfr_client_secret`	secret de l'application OAuth LinuxFR, stocké dans le fichier `0600`

Les boards prédéfinis sont : `dlfp`, `euromussels`, `gabuzomeu`, `moules`, `devnewton`, `ototu`, `reveildutroll`, `sveetch`. Il suffit de les mettre dans `selected_boards` ; ils sont déjà connus du client.

LinuxFR

Deux méthodes d'authentification sont disponibles ; choisir la tribune directe `dlfp` n'impose plus de passer par gb2c.

Directe, sans gb2c. Créer une application OAuth sur linuxfr.org/api/applications avec l'adresse de redirection native `urn:ietf:wg:oauth:2.0:oob`, puis renseigner son identifiant et son secret dans les réglages. Le bouton **Ouvrir l'autorisation** ouvre LinuxFr ; après validation, LinuxFr affiche un code d'autorisation, à coller dans le champ puis à échanger avec **Connecter le code**. Le rafraîchissement se fait directement sur `https://linuxfr.org/api/oauth/token`. LinuxFr (Doorkeeper 4.x) refuse les redirections `http` et une boucle locale `https` imposerait un certificat auto-signé : la redirection native est la voie prévue pour les applications installées. Le secret reste dans le fichier de configuration local, créé avec les droits `0600`.

Via gb2c. Si aucun identifiant d'application n'est renseigné, le bouton ouvre `/gb2c/linuxfr/authorize`. gb2c termine l'OAuth par une redirection vers `/linuxfr.html#<jeton base64>` et ne peut pas transmettre ce fragment à un client natif : coller l'URL de retour (ou le jeton) dans le champ LinuxFR et appliquer. roinroin accepte aussi un jeton d'accès brut ; le rafraîchissement se fait alors via `/gb2c/linuxfr/refresh`.

Raccourcis

Raccourci	Action
`Entrée` / `Ctrl+Entrée` / `Alt+Entrée`	envoyer
`Alt+B` / `I` / `U` / `S`	gras, italique, souligné, barré
`Alt+T` / `C` / `P`	télétexte, code, spoiler
`Alt+A` / `M`	BLAM, Moment
`Ctrl+Tab` / `Ctrl+Shift+Tab`	tribune suivante / précédente
`Ctrl+1` … `Ctrl+9`	tribune par position
`Ctrl+R` / `F5`	tout actualiser
`Ctrl+,`	réglages
`Ctrl+M`	tout marquer comme lu
`Ctrl+C`	copier le texte sélectionné dans la tribune
`Ctrl+T` / `Ctrl+E`	ToToz / Emoji
`Échap`	fermer un panneau ou désélectionner

Architecture

```
src/
  main.rs            démarrage iced, icône de fenê
  app.rs             état, update Elm, abonnements
  ui.rs              arbre de widgets, posts, comp
  hover_text.rs      texte riche avec liens cliqua
  visible.rs         suivi des posts réellement vis
  net.rs             gb2c, boards directs, SSE, to
  tsv.rs             parseurs TSV et XML bornés et
  message.rs         scanner BML, norloges, bigorn
  conversation.rs    résolution des références, gr
  model.rs           clés de source/tribune/post,
  selection.rs       positions et plage de sélecti
  config.rs          configuration TOML, état JSON,
  notify.rs          notifications D-Bus
  theme.rs           palette et styles iced
```

Le client est volontairement sans base de données : les posts vivent en mémoire, l'état persistant se limite aux derniers posts lus, aux tribunes connues, à la tribune active, au brouillon et à la taille de la fenêtre et du panneau des tribunes. Les réponses distantes sont toujours traitées comme non fiables (bornes de taille, UTF-8 tolérant, caractères de contrôle filtrés).

Différences assumées avec les clients existants

- roinroin peut afficher **gb2c et des boards directs simultanément** ; les autres clients choisissent généralement l'un ou l'autre.
- L'interface est en français pour l'instant ; la structure de configuration prévoit déjà une traduction.
- Pas de tray icon ni de mode dockapp : l'intégration passe par les notifications, le `.desktop` et la fenêtre native.
- Tous les posts conservés sont affichés (jusqu'à `history_limit` par tribune), sans fenêtre glissante ni bouton « charger les précédents ».
- Les totoz inline sont des pastilles cliquables (recherche/prévisualisation) plutôt que des images chargées automatiquement dans chaque post, pour éviter de télécharger des dizaines d'images à l'ouverture d'une tribune.

Licence

MIT. Voir `LICENSE`.

La police embarquée `assets/TwemojiMozilla.ttf` garde sa licence propre : code de twemoji-colr sous Apache-2.0, dessins Twemoji sous CC-BY-4.0. Voir `assets/TwemojiMozilla.LICENSE.md`.

© 2026 GitHub, Inc.