SimpleX Chat: Configurazione e deployment di un relay di trasporto dedicato

SimpleX Chat: Configurazione e deployment di un relay di trasporto dedicato

L'architettura decentralizzata di SimpleX Chat consente la totale sovranità sui metadati e sulle code di consegna messaggi attraverso il disaccoppiamento tra client e server di transito.

Ospitare un proprio **chat relay** personale (SMP Server) garantisce un instradamento affidabile dei flussi di messaggistica e una gestione autonoma delle risorse di rete.

Di seguito vengono analizzate le procedure di configurazione, la gestione dei processi in ambiente Linux e l'integrazione di servizi systemd resilienti.

Primo ostacolo: Haskell odia systemd (e ama i TTY)

La prima idea del sistemista medio è creare un servizio systemd pulito che avvii `simplex-chat --relay`. Avvii il servizio, digiti `systemctl status` ed ecco lo schiaffo: crash immediato con `Prelude.undefined`.

Il motivo? La CLI di SimpleX è scritta in Haskell e si aspetta un terminale interattivo (TTY). Quando systemd lo avvia in background come demone puro, la libreria TUI di Haskell impazzisce e si suicida.

La pezza a colori? **Tmux**. Bisogna costringere systemd ad avviare la CLI dentro una sessione tmux staccata:

```
[Unit]
Description=SimpleX Chat Relay Client (in Tmux)
After=network.target

[Service]
```

```
Type=forking
User=franco
ExecStart=/usr/bin/tmux new-session -d -s simplex /usr/bin/simplex-chat --
relay
ExecStop=/usr/bin/tmux kill-session -t simplex
Restart=always

[Install]
WantedBy=multi-user.target
```

In questo modo creiamo un TTY fittizio. Funziona. Ma quando ti colleghi con `tmux attach`, ti scontri con il secondo problema: la console sgrana, il prompt si sposta a metà dello schermo e sovrascrive l'help. Perché? Perché la TUI non sa come ridimensionarsi se ti colleghi da un terminale con dimensioni diverse. La cura è staccarsi e riattaccarsi forzando il ridimensionamento: `tmux attach -d -t simplex`.

Come usare il tuo relay sui dispositivi mobili

Una volta che il servizio è attivo e funzionante dentro tmux, puoi collegarti alla sessione (`tmux attach -d -t simplex`) per configurare ed estrarre l'indirizzo di invito del tuo relay.

All'avvio, la CLI configurata con `--relay` inizializzerà il profilo e ti chiederà di inserire un nome visualizzato e confermare la creazione dell'indirizzo:

```
No chat relay user profile found, it will be created now.
Please choose chat relay display name.
display name: <tuo_nome>
Current user: <tuo_nome>
Create relay address (Yn): y
Chat relay address is created:
https://smp<N>.simplex.im/r#<stringa_identificativa>
```

(Se perdi il link o lo vuoi rivedere in un secondo momento, puoi utilizzare il comando `/sa` nel prompt della CLI).

Ora, copia questo indirizzo e portalo sul tuo client mobile o desktop quotidiano:

1. Apri SimpleX Chat sul telefono.

2. Vai su **Impostazioni** (Settings) -> **Rete e Server** (Network & Servers) -> **Server di chat** (Chat relays).
3. Clicca su **Aggiungi server** (Add server) e incolla l'indirizzo copiato.

ATTENZIONE: allo stato attuale delle cose (v6.5.5), questo relay personale servirà **esclusivamente** per ospitare ed instradare i canali pubblici (1:molti). Tutte le altre comunicazioni (chat 1:1 e i gruppi standard) continueranno a funzionare esattamente come prima in modalità P2P/mesh classica, come se il relay non ci fosse.

La possibilità di creare veri e propri "relayed groups" per le conversazioni molti:molti su canali cifrati transitanti via relay è in fase di rilascio (prevista entro fine giugno 2026), e sarà quella la vera svolta per abbattere il consumo di batteria dei partecipanti ai gruppi.

Fino ad allora, usalo per i tuoi canali di broadcast e tieni d'occhio i changelog.

[Che il talpone sia con voi]

Fluxer Icon

Canale di discussione e supporto

Canale di discussione e supporto. Per confrontarti su configurazioni o altro entra su fluxer.dreadful.work (https://fluxer.dreadful.work/invite/dreadful).