

Dal cloud al self-hosting

Fabio Alessandro “Fale” Locati

October 26, 2024

EMEA Principal Specialist Solutions Architect, Red Hat

Why?

My approach

Tips & Tricks

Wrapping up

About me

- IT user since 1996
- Working in IT since 2004
- EMEA Principal Specialist Solution Architect @ Red Hat

Why?

My Google relationship

- Gmail:
 - Limited beta launch: April 1, 2004
 - Public beta launch: February 14, 2007
 - Official launch: July 7, 2009
 - User since September 19, 2004
- Google-first approach

My trigger

- May 13, 2013: Urs Hölzle (SVP Technical Infrastructure and Google Fellow) announced that Google Reader would have been discontinued on July 1, 2013
- May 13, 2013: Urs Hölzle (SVP Technical Infrastructure and Google Fellow) announced that the CalDAV API would have been discontinued on September 16, 2013 for the general public
- May 15, 2013: Nikhyl Singhal (Director of Product Management) communicated that Google Talk was going to be replaced by Google+ Hangout and that XMPP would have been dropped

Enshittification (alternately, crapification and platform decay) is a pattern in which online products and services decline in quality. Initially, vendors create high-quality offerings to attract users, then they degrade those offerings to better serve business customers, and finally degrade their services to users and business customers to maximize profits for shareholders. (source Wikipedia)

Reasons to prefer the self-hosting

- **Goals alignment:** Why “*they*” provide the software?
- **Software quality:** Decreasing quality.
- **Control:** Who controls what?
- **Offline first:** What if the connection is not available?
- **End of Life policies:** Who is in charge?
- **Security:** What’s the attack surface? What’s the collateral damage risk?
- **Privacy:** Who is looking at your data? What are “*they*” using your data for?

A spectrum of trust

Tin foil hat ← Trust → Google first

A horizontal line with arrows at both ends, representing a spectrum. The word 'Trust' is centered on the line. To the left of the line is the text 'Tin foil hat' and to the right is 'Google first'.

My approach

Priorities

1. Reject lock-ins
2. Leave BigCorp(s)
3. Prefer decentralized/federated (self-hostable) options
4. Prefer self-hosting
5. Prefer (technically) sensible implementations

In economics, vendor **lock-in**, also known as proprietary lock-in or customer lock-in, makes a customer dependent on a vendor for products, unable to use another vendor without substantial switching costs. (source Wikipedia)

Lock-ins

- Technical (un-available alternatives)
- Economical (sunk cost)
- Data (un-exportable data)
- Social (walled gardens)
- Habit (un-reusable skills)

Europe helps on some of those with the GDPR, DSA, and DMA.

- Email is your Identity and (usually) the biggest collector of information
- It takes *a lot* to sunset email addresses
- My path:
 - 1996 - My first email (Tiscali)
 - 2004 - Creation of my Gmail
 - 2013 - Start to host my mailbox on Kolab
 - 2018 - Move from Kolab to Fastmail
 - 2020 - Auto-replyer on Gmail
 - March 5, 2024 - Deletion of my Gmail
 - (now) - Evaluate pure self-hosting
- **Remember**, email is your identity!
 - Owning your own domain is key
 - Try to pick the “correct” name at the first shot

Messaging

- Messaging is social by definition
- It is hard to sunset messaging channels
- My path:
 - 1999 - Start using MSN
 - 2005 - Start using Google Talk
 - 2005 - Sunset MSN
 - 2013 - Start to host my XMPP server, create of a Google Account
 - 2015 - Start using Telegram & Signal
 - 2020 - Start using Matrix, make all chats going through Matrix
 - 2020 - Sunset XMPP
 - 2023 - Sunset Signal
 - (todo) Fully sunset Google Hangout
- Messaging is hard:
 - Very few protocols (if any) warrant security, federability, and traction
 - Messaging is social by definition

Socials

- Many social are not very useful if you start thinking about it
- The usage of social is probably more important than having accounts
- Some social tend to track you even outside their website
- My (current) preference: GitHub, LinkedIn, Fediverse (GoToSocial)

- There is no great solution today
- Possible strategies:
 - Reduce the cloud services
 - Move to a different store (e.g.: F-Droid)
 - Move to a more privacy focus build (e.g.: GrapheneOS)
- My long term strategy: Linux Phone
 - Software viability (stability, missing features)
 - Hardware viability (availability, battery life)
 - Banking/OTP apps
 - Wallets (eDocuments?)
- My strategy: F-Droid + help making Linux Phone a thing

Other examples

- OS (Linux)
- Browser (Firefox)
- RSS Feeds (FreshRSS)
- Media (Jellyfin + Delfin + Finamp)
- Recepies (Tandoor)
- Passwords, OTPs, *SPID* (gopass + Android Pass Store)
- VPN (Nebula)
- Maps (Organic Maps + OSM + *Transitous*)
- AI chatbots (llama + granite + ollama + open webui)
- Notes (NextCloud)
- Media backup (NextCloud)
- CalDAV/CardDAV/FileDAV (NextCloud)

Tips & Tricks

First things first

- Evaluate which information you have in which service
- Prioritize urgent services (if they are important)
- Prioritize services that are hubs of information (e.g.: email)

Explore

- It is very hard (i.e.: impossible) to immediately find the perfect solution
- You will need to learn a **lot** of stuff
- YMMV (Your Mileage Might Vary)

Solutions

- Stop-gap solutions might be a good idea
- Quick solutions might stick way more than you were expecting
- Play the long game!

Wrapping up

Wrapping up

- It is possible to exit from cloud services
- You choose how much you want to exit the cloud services
- It is a long and complex process, but you'll learn a lot along the way

Questions?

Email: mail@fale.io
Fediverse: [@fale@fale.io](https://fale.io/@fale)



Links

- `https://killedbygoogle.com`
- `https://switching.software`