Instant Messaging

Extremely Brief IM History

- Bulletin boards with “direct message” features
- Internet relay chat (IRC)
- AOL instant messenger (AIM)
- Short Message Service (SMS)
- Jabber/XMPP (eXtensible Messaging and Presence Protocol)
- Today: whatsapp, signal, telegram, matrix etc.
- Tomorrow? Maybe the return of interop? (MIMI/MLS)

IM Key Features

- Less formal, more ephemeral, messaging (compared to email)
  - But: wonder how a former UK prime minister considers that now?
- Buddy lists/Rosters
- 1:1 or group chats, some allow WebRTC calling
- Presence, user-is-typing, knowing when/if messages seen
- Multiple devices per user
- Ability to edit posted messages (within limits)
- End-to-end encrypted (e2ee) or not (and if so, how well)
- Gateways to other IM systems (some allow, some don’t)
- Silo’d vs. interoperable/federated (linked to business models)
- Emojis, yay:-)

IM systems I’ve used a bit

- I don’t use whatsapp etc so only have some limited of those
- But I have used a bunch of IM systems over the years...

Internet Relay Chat (IRC)

- One of the oldest, from late ‘80’s, peaking about 2003, declining since, but still used
- Users known just by “nick” – probably not authenticated
- Rooms known as “channels”
- Client-Server and Server-Server protocols specified in RFC1459 but implementations evolved away from that (nonetheless, it’s federated)
- Performance/security issues with netsplits
- Try it out: https://webchat.freenode.net/
- Wikipedia: https://en.wikipedia.org/wiki/Internet_Relay_Chat

eXtensible Message and Presnce Protocol (XMPP)

- Also known as: jabber
- Sort-of successor to IRC, popular from early 2000’s, less so now
- Was used internally by many proprietary services, e.g. Google Talk, Facebook chat, even if they didn’t say so much, less so now
- Users authenticate to chosen service, have names like email addresses, nick’s used in rooms which also have email-like names, 1:1 messaging natively supported
- Client-Server and Server-Server protocols more secure than IRC, but “less secure” than more modern protocols
- Still some issues with performance in federations
- Specifications are a mixture of IETF RFCs (e.g. RFC6120) and extension specifications (XEP’s) maintained by the XMPP standards foundation (XSF)
- Try it out: https://xmpp.org/ has HOWTOs, links to client s/w and services where you can register an account
- Wikipedia: https://en.wikipedia.org/wiki/XMPP

Signal

- Messaging app with a focus on end-to-end encryption (e2ee); early versions (~2010) based on XMPP’s Off-the-record (OTR) protocol, but changed significantly later
- Users start with a phone number (still in the process of loosening that requirement), 1:1 messaging and groups supported
- Can do WebRTC calls
- Signal protocol is deliberately not federated and doesn’t support (much) crypto agility
- Very large groups may have performance issues
- Security based on double-ratchet, later also used for whatsapp and others
  - https://signal.org/docs/specifications/doubleratchet/
- Key difference: IM commonly involves multiple asynchronous exchanges unlike email or TLS, and former group members might renege or be compromised, so we’re now interested in post-compromise security
- Wikipedia: https://en.wikipedia.org/wiki/Signal_(software)
- Details: https://signal.org/docs/

Matrix

- Another “modern” messaging scheme, (circa 2014) with federation but “less” e2ee than signal
- Usernames like @joebloggs:example.org, 1:1 messaging also via “rooms,” rooms are named like #roomname:example.org
  - example.org in the above is the “homeserver” for the room/user
- Anyone can install a “homeserver” to which local clients authenticate; homeservers handle federated traffic (with the usual load problems)
- Usual set of clients (desktop, mobile) that are (mostly) shims on a browser engine, so can also be used directly from a browser
- Wikipedia: https://en.wikipedia.org/wiki/Matrix_(protocol)
- Try it out: https://matrix.org/ has HOWTOs and links for clients and homeservers (I only ever tried synapse, not sure if other homeserver implementations are stable)

Others...

Ones I’ve used a little bit:

- Mattermost
- WebRTC chat
- Zulip
- SMS

- Mastodon

Ones I’ve not used:

- Whatsapp
- Instagram
- Facebook messenger
- iMessage

- Telegram

- Twitter/X

MLS/Mimi

- EU Digital Markets Act is now in force
- CEC/EU identify “gatekeepers” who affect many EU citizens and impose more rules on those
- Seems likely that’ll impose a need for users of one gatekeeper to be able to message users of another
  - Could be broader interop is mandated, we’ll see
- That means there’s a need for some interoperable format for such messages
  - Note – does not mean all messages will be in that format
- IETF is working (in the mimi working group) on an MLS-based messaging format/architecture that could be the basis for that interop
  - https://datatracker.ietf.org/wg/mimi/about/
- Mimi will be based on RFC 9420 (message layer format) which is a TLS-like definition of how things like IM apps can get e2ee but perhaps also scale better than some current systems
- There is a *lot* of politics in that one!

Questions

- What do you (dis)like about systems you use?
- What’s the main business asset of these IM operators? (Ignoring other systems the same company may operate)
- What barriers to entry exist?
- How do you think this ecosystem will evolve?