layout: layouts/article.html title: "Instant notifications for SimpleX Chat mobile apps" date: 2022-04-04 preview: Design of private instant notifications on Android and for push notifications for iOS.

permalink: "/blog/20220404-simplex-chat-instant-notifications.html"

Instant notifications for SimpleX Chat mobile apps

Published: April 04, 2022

SimpleX Chat is the first chat platform that is 100% private by design - it has no access to your connections

Since we released SimpleX Chat mobile apps couple of weeks ago we've had a lot of excitement from our users - nearly 2000 people downloaded the app after the announcement!

Huge thanks to everybody who downloaded and connected to us via the chat - there were many great questions and suggestions, and on some days I spent most of the time chatting to our users :)

Since we released the app, we've added and released:

- support for iPhone 7.
- configurable SimpleX servers.
- message replies, editing and deletion.
- profile images.
- and, most importantly, private instant message notifications on Android devices - more on that below.

Install the apps and make a private connection!

Once you install the app, you can connect to anybody:

1. Create your local chat profile - it is not shared with SimpleX servers. It is local to your devices, and it will be shared with your contacts only when you connect.
2. To make a private connection, you need to create a one-time connection link or a QR code via the "Add contact" button in the app. You can show the QR code to your contact in person or via a video call - this is the

most secure way to create a connection - or you can share the link via any other channel. Only one user can connect via this link.
3. Once another user scans the QR code or opens the app via the link the connection will be created and you can send end-to-end encrypted messages privately, without anybody knowing you are connected.

See demo video that shows how two users connect and send the first messages.

Why we are doing it

We are building SimpleX Chat because we believe that privacy is a fundamental human right, and that protecting our personal network of contacts is even more important than the content of the messages - sharing this network can lead to various adverse consequences, from manipulating us into buying goods we don't need, manipulating election processes, and in some cases, prosecuting innocent people. For example, Mohamedou Ould Salahi was detained in Guantanamo prison for 15 years after a single "wrong" phone call. His story is told in his memoir and in The Mauritanian movie.

Problem - users expect to be instantly notified when messages arrive!

Our first users realized that what we take for granted in messaging apps - instant message notifications - is missing in our first release of SimpleX Chat apps. Quite a few people thought that it was a bug, rather than a missing feature. Sorry to disappoint!

Why can't we just do what messenger X does?

SimpleX Chat is the first and the only messenger we know of that operates without user identities of any kind. There are no phone numbers, emails, usernames, public keys, or any other addresses or identifiers to uniquely identify the users to the network or servers. That is why we say it is 100% private by design, and fundamentally different than other chat platforms.

Instead, SimpleX Chat assigns these identifiers to unidirectional message queues. What looks to SimpleX Chat users like contacts and groups [1], to SimpleX servers looks like an unorganized and unrelated collection of unidirectional message queues. Our servers do not know which queues belong to which users, contacts or groups. Even a single conversation can happen via two different servers (one for sent and another for received messages). This makes our personal network of contacts invisible to the servers.

But it also creates a problem for instant notifications - all push notification services require having a device token.

So, how can we operate without identities and still have instant notifications?

[1] yes, we have groups in our terminal app, and the UI to manage them is coming to mobile apps soon. Some users have already firgured out how to create groups via chat console.

We've cracked it for Android!

After some research into how push notifications work on Android, and open-source alternatives to Google push notifications, we discovered how to avoid sharing device tokens with any servers.

We have implemented message reception as a background service (in Android terminology, a "foreground service" showing a notification icon when the service is running) following the same design as ntfy.sh created by Philipp Heckel, who, in turn, credits the design to the blog post by Roberto Huertas. Big thanks to them!

How does it work? When the app is first started on an Android device, it starts the background service that keeps the TCP connections to the messaging servers open with almost no traffic (only doing periodic checks that connections still exist). It consumes only a few percents of battery per day, depending on how stable your internet connection is, and delivers message notifications as soon as messages arrive.

This service continues running when the app is switched off, and it is restarted when the device is restarted even if you don't open the app - so the message notifications arrive instantly every time. To maximize battery life, it can be turned off by switching off "Private notifications". You will still receive notifications while the app is running or in the background.

So, for Android we can now deliver instant message notifications without compromising users' privacy in any way. The app version 1.5 that includes private instant notifications is now available on Play Store, in our F-Droid repo and via direct APK downloads!

Please let us what needs to be improved - it's only the first version of instant notifications for Android!

Our iOS approach has one trade-off

iOS is much more protective of what apps are allowed to run on the devices, and the solution that worked on Android is not viable on iOS.

We already have background refresh in the iOS app that periodically checks for new messages, and if you use the app every day it delivers notifications within 10 or 20 minutes. It is not instant, but it may be usable for some. If you use the app infrequently, however, this delay can become several hours, or your phone may stop checking for the new messages completely. This is not ideal!

The only solution known to us is using Apple's push notifications service (APN) to deliver push notifications.

We planned for it, so we added to v1 of SMP (the protocol used by our servers) an extension allowing the client to subscribe to notifications from message queues, via separate queue addresses, and using separate cryptographic keys for each queue. This has to be enabled by the client for each queue separately. We haven't used this extension so far, and now we are building a SimpleX notification service based on it.

If the user enables push notifications, then for each contact the app would enable a notification subscription and pass credentials to the notification server together with the device token required to deliver push notifications to user's device.

The notification server will subscribe to these notifications from SMP servers. The notifications do not include any message content, only the signal that a message has arrived to the server. Notification server is only allowed to send 2-3 hidden notifications per hour to the device. The notification is end-to-end encrypted and contains information about which server has a message, so that the client can connect to the server, retrieve and decrypt the message, and show the notification to the users including sender name and the message content. None of this information is shared with any server.

If the user receives more than 2-3 messages per hour, the notification server can send additional visible notifications that would simply say "you have a new message", and the user will have to open the app to receive and see these messages. We are also investigating whether we can use "mutable-content" notifications that allow doing some processing when the notification arrives before showing it to the users.

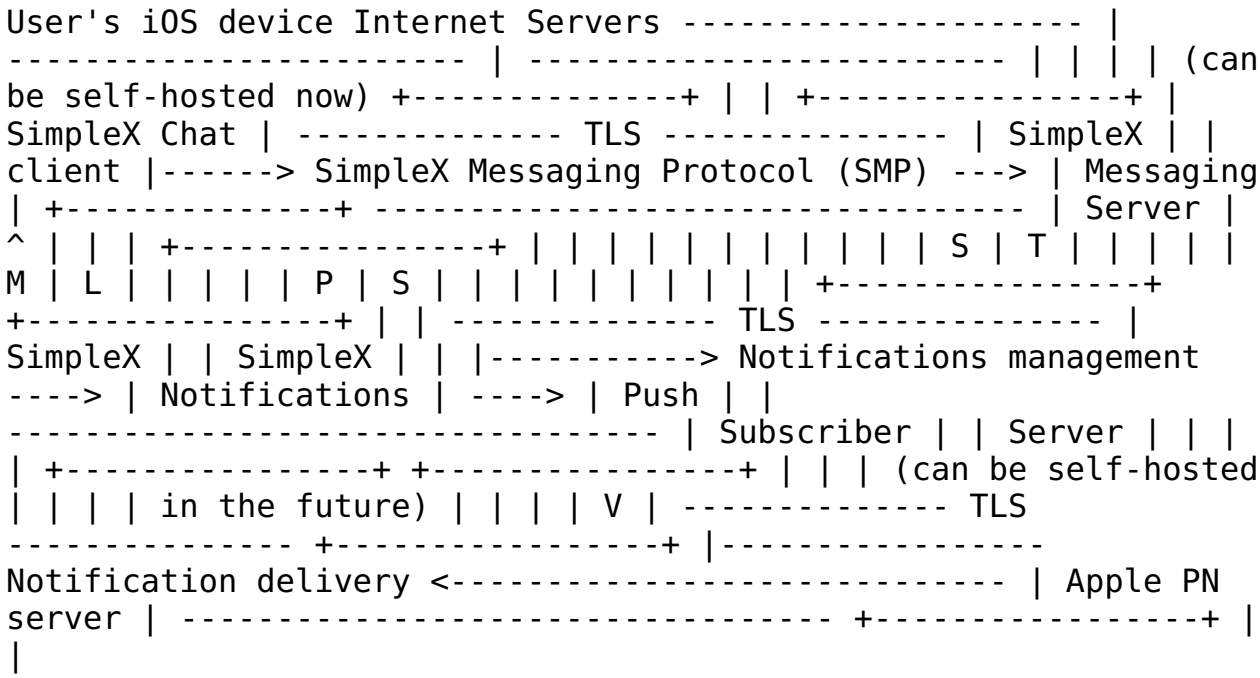
It is a substantial amount of development, we are aiming to release it later this month.

This design is a compromise between privacy and convenience. The notification server will have to have a device token to deliver notifications. Several things we did (or plan to do) to improve this compromise:

1. The notification server will only store device tokens and queue addresses in memory, making it more complex for a potential attacker to access. If server has to be restarted, they would lose all configured notification subscriptions and the clients would have to create them again. We will program the clients to periodically check for the existence of notification subscriptions on the notification server.
2. The notificaiton server will not know the addresses of the messaging queues used to receive or send messages. A different address is used to subscribe to notifications. So while the notification server would have the knowledge of how many queues your device has (and on which servers), it still won't know who is sending you the messages.
3. We are also planning to split the logic of notification subscriptions and delivering notifications to the devices to two different servers. The server that subscribes to the notifications could be self-hosted, allowing

you full control of how you deploy it. Only this server would know which messaging servers you use or how many messaging queues you have. The server that delivers notifications to the devices will be managed by SimpleX Chat as we have to authorize it with Apple's push notification service. This split will not be available in the first release. We plan to add it a bit later.

So, with the notification servers added, our network design will look like this:



Please let us know what you think about this design and about this privacy / usability trade-off:

- For you, is this an acceptable compromise, if you can choose to disable instant notifications?
- Do you have any ideas about how this design could be improved upon?

Thank you!