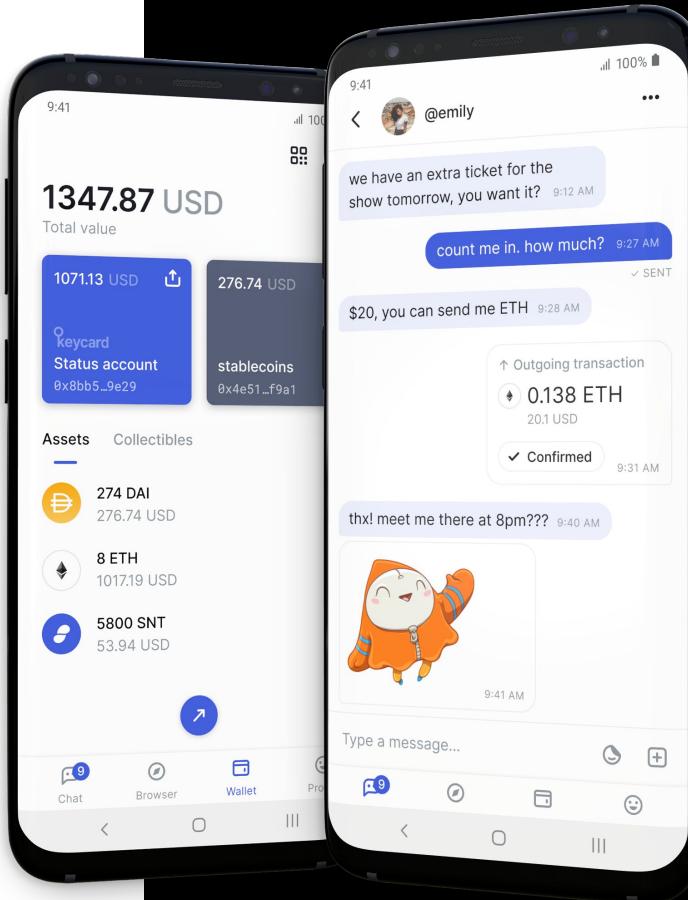




status

V1.0 Reviewers Guide



| | | | |
|--|----|------------------------------------|----|
| 1. Core Features | 11 | 3. Technical Specifications | 37 |
| Anonymous Account Creation | 13 | Serverless tech stack | 39 |
| Private messaging | 16 | Specifications Repo | 40 |
| Integrated Crypto Wallet | 23 | | |
| Private Web3 Browser | 27 | | |
| | | | |
| 2. Status Network Token (SNT) Utility | 30 | 4. Forward Thinking | 41 |
| Ethereum Name Service Registration | 33 | Chat improvements | 43 |
| Decentralized Sticker Market | 34 | Whisper Upgrade | 44 |
| Dap.ps | 35 | SNT Utility | 45 |
| Status Voting Dapp | 36 | Performance Optimization | 46 |

Status is a private, secure communication tool to uphold human rights and empower sovereign communities

Status App ushers in a new era of mobile messaging, transacting and browsing, by breaking the mold of the standard client-server model.

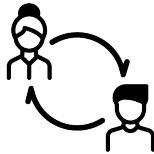
The long term goal of Status has always been to provide communication tools that remove unnecessary, rent-seeking third parties, and oppressive actors to place sovereignty back into the hands of the individual. Over the past 20 years, the internet as we know it has not only been controlled by a few, but these corporations business models create the incentive to extract as much data and information out of end users for massive financial profit.

Legacy social networks consist of 3 parties: the owner, the advertiser, and the user. Despite each playing a critical function, these parties operate with vastly different goals in mind and current models fail to provide a means by which their incentives can coexist and be aligned.



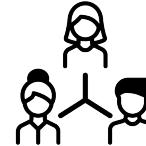
The Owners

attract and retain users to their platform. Their goal has historically been to create a walled garden around a user base and extract value from it. Facebook has published non-consensual social experiments that have demonstrated that by manipulating the information the user consumed while engaged with the network, they could effectively manipulate what that user thinks, feels, and believes (Kramer et al. 2014)¹



The Advertiser

or Data Broker, enables the Owner to extract value from the network, ultimately sustaining the platform. This is done by purchasing the user's data, purchasing qualified leads to their products or services, or by purchasing targeted advertising inventory based on user profiling performed by the Network Owner.



The User's

perspective in the network is quite different. Often users do not approach these networks with the explicit intention of buying products, but rather to connect with friends and loved ones or to reach communities who share their own special interests. Often they are powerless to control the information they consume, or how the network is developed. They have no choice but rely on the Owners and Advertisers to behave ethically, or stop using the network entirely

Owners are able to maintain tight control over the entire system because the platforms used rely on the current available internet infrastructure – the client-server model. They own the servers used to process massive amounts of data and pass communications of all forms from client to client. This can be messages, financial transaction, and content.

This means that there are central points, or servers that serve as intermediaries and hubs in all our communication, be it browsing the web, messaging, streaming videos or sending emails. Under this model, users are dependent on these central points of failure.

This means that it is enough for an adversary to shut down this central point (or points) - usually identified by IP addresses and domain names - or to cut off access to them - and they are able to shut down our communication. Like Turkey's Wikipedia ban, or ban on

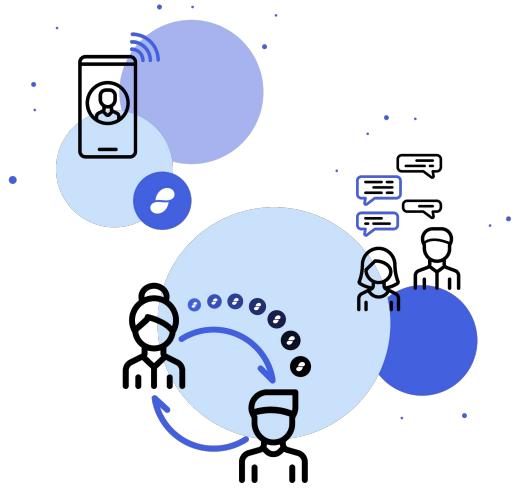


accessing social networks like Facebook, Twitter, and messaging apps, in countries with authoritarian regimes.

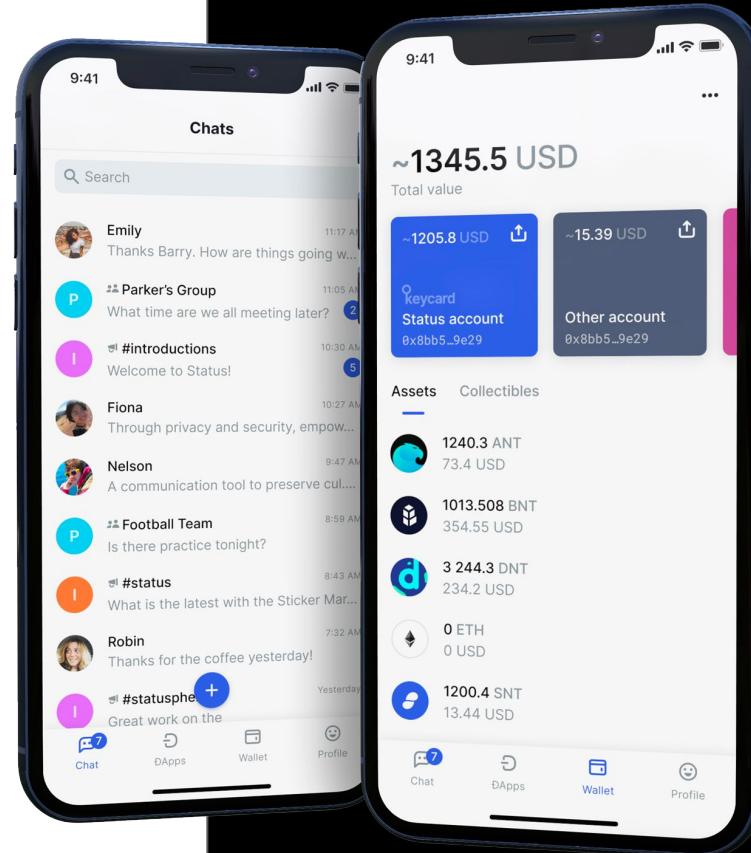
The current model of the internet actually empowers censorship endeavours like this.

Not to talk about the fact that governments can issue subpoenas and seize server logs of websites and ISP-s and different analytics data, and get unrestricted access to information on our individual lives.

Enter Status



Status breaks down the legacy social networks with communication that is **permissionless, trustless, with fair access** and is **cryptographically verifiable**.



Status uses state of the art technology to provide private, secure, communication via a peer-to-peer model without the required surveilling intermediaries. The Status App combines a messenger, crypto wallet, and web3 browser that places the end users in total control of their communications. The trusted third parties no longer have control of our communication and can therefore no longer monetize off of our data.

The messenger is built with the peer-to-peer messaging protocol Whisper. The wallet and browser are built on the Ethereum public blockchain

Available for iOS and Android



*Desktop for MacOS, Windows, and Linux currently in Alpha

The image displays three screenshots of the Status app. The top screenshot shows a messaging interface with a dark background. It includes a message from "Status" at 9:41 AM saying "Hi. Glad you're using Status as well!", a reply from "You" at 9:42 AM saying "Sure thing! How are you today? Want to grab a coffee?", another message from "Status" at 9:42 AM saying "Of course, let's meet over at Fairfax in 20 mins?", a reply from "You" at 9:42 AM saying "Sounds good!", and a message from "Status" at 9:42 AM saying "Cool, see you there.". The bottom right screenshot shows a wallet interface with a blue header displaying "1,382.20 USD". Below the header are three circular icons with arrows. The main area is titled "Assets" and lists three items: "1,109 SNT" with a value of "\$83.96", "1.0273 ETH" with a value of "\$134.32", and "352 OMG" with a value of "\$392.00". The bottom left screenshot shows a browser interface with a dark background, displaying a single page with a light-colored header and body.

Privacy and Security by default

| | Status | WeChat | Whatsapp | Facebook Messenger | Signal | Telegram |
|----------------------------|--------|--------|----------|--------------------|--------|----------|
| Peer-to-peer | YES | NO | NO | NO | NO | NO |
| E2E encryption by default | YES | NO | YES | NO | YES | NO |
| Open Source | YES | NO | NO | NO | YES | NO |
| Anonymous Account Creation | YES | NO | NO | NO | NO | NO |
| Perfect Forward secrecy | YES | NO | YES | YES | YES | YES |
| Payments in Chat | YES | YES | NO | YES | NO | NO |



Glossary

Status introduces a completely new way of interacting with the web. Here is a breakdown of terms that may be unfamiliar or different from what you're used to:

- **Account** - Your Status account, accessed by the **seed phrase** that you create or import during onboarding. A Status **account** can hold more than one Ethereum address, in addition to the one created during onboarding. We refer to these as additional **accounts** within the wallet.
- **Seed Phrase** - A set of friendly-to-read words, randomly selected from the BIP39 standard list and used to recover or access your Ethereum account on other wallets and devices. Also referred to as a "mnemonic phrase," "recovery phrase" or "wallet backup" across the crypto ecosystem. Most crypto apps use this same standard to generate accounts.
- **Chat Key** - Messages on the Status chat protocol are sent and received using encryption keys. The public **chat key** is a string of characters you share with others so they can send you messages in Status.

Glossary

Status introduces a completely new way of interacting with the web. Here is a breakdown of terms that may be unfamiliar or different from what you're used to:

- **Chat Name** - Three random words, derived algorithmically from your **chat key** and used as your default alias in chat. Chat names are completely unique; no other user can have the same three words.
- **ENS Name** - Custom alias for your **chat key** that you can register using the Ethereum Name Service. ENS names are decentralized usernames.
- **Wallet Key** - A 64 character hex address based on the Ethereum standard and beginning with **0x**. Public-facing, your wallet key is shared with others when you want to receive funds. Also referred to as an "Ethereum address" or "wallet address."
- **Peer** - A device connected to the Status chat network. Each user can represent one or more peers, depending on their number of devices.
- **Mailserver** - A node in the Status network that routes and stores messages, for up to 30 days.



Core Features

Core Features

Anonymous Account Creation

- No phone, no email, no bank account
- Key Creation & management
- Password and what it is for / where it is stored

Integrated Crypto Wallet

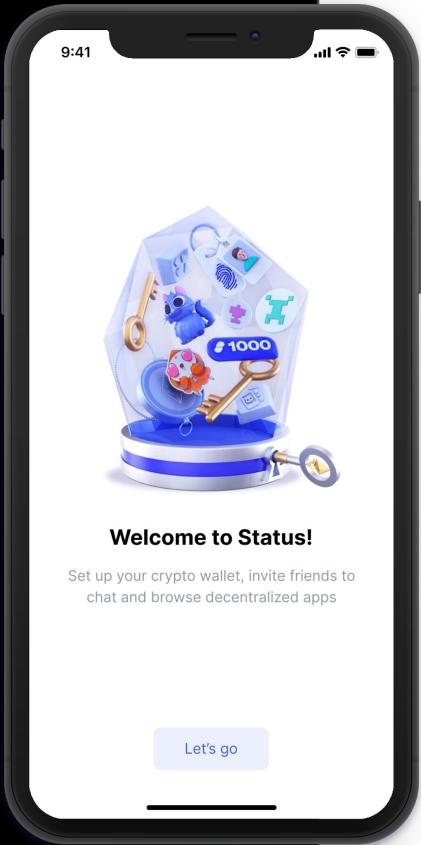
- ERC-20 and ERC-721 tokens
- Integrated with chat and browser
- Multi-account for Decoupling of keys
- Signing phrase

Peer-to-Peer Messaging

- private chats / group chats / public chats
- Send and receive crypto payments in chat
- Whisper & dark routing
- e2e encryption
- perfect forward secrecy

Private Web3 Browser

- Explore and interact with ethereum Dapps
- Dap.ps
- EIP's



ANONYMOUS ACCOUNT CREATION

No email. No phone number. No bank account

Creating an account is entirely anonymous on Status. No secondary forms of verification are required - no email address, no phone number, no bank account.

Your identity is secure and private by design. You get a locally generated cryptographic keypair—private and public, for both wallet and chat—to control your account. This keypair is encrypted and securely stored locally on your device (or on a Keypad – currently only android users). No one can access it. Not even Status. During account creation, you are prompted to select a chat key and name – the three word name and image are a readable version of this key pair. They are always unique so nobody can pretend to be you.

This is how others will see you in Status until you add them as a contact, or provide them with your ENS name (assuming you have one).

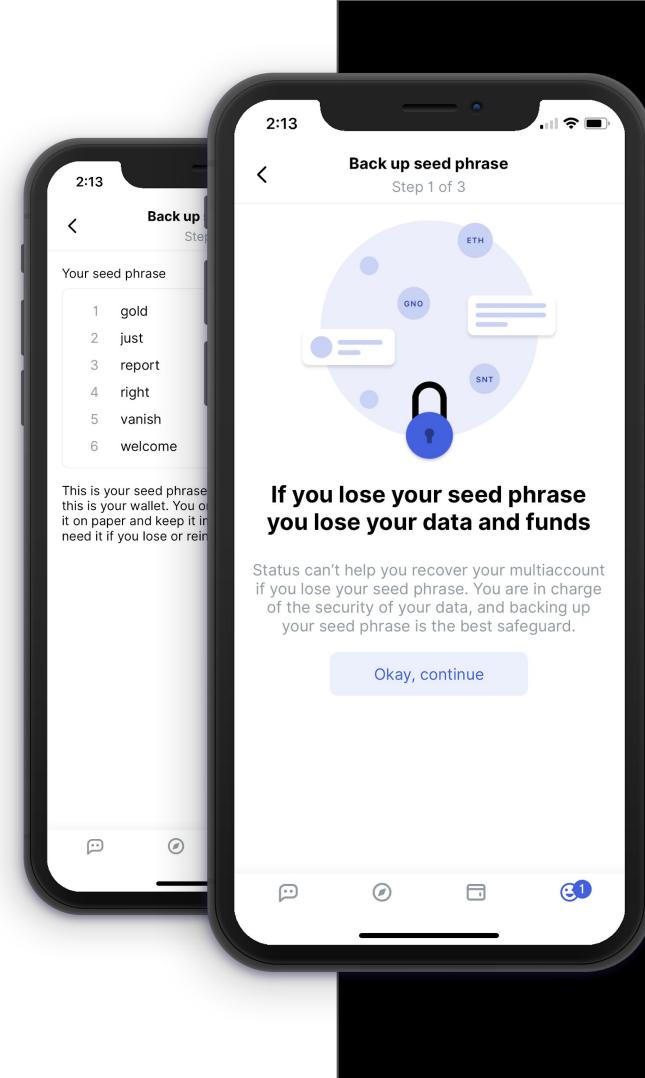
ANONYMOUS ACCOUNT CREATION

The Seed Phrase

It is very important that users backup their seed phrase. The seed phrase is your only means of decrypting and accessing your private keys. Status will never access a user's private chat and wallet keys - in fact, we physically cannot. They are only accessible with your seed phrase. The seed phrase is the randomly generated string of 12 words that you create during onboarding, either in Status or another wallet. Your seed phrase is only ever kept locally in your device's secure storage enclave, and **NEVER** in a third party server vulnerable to attack or a data breach.

We recommend you write down your seed phrase on a piece of paper and store it somewhere safe.

If you lose or change your phone, or uninstall and reinstall Status, it is **the only way** to recover your account again.



ANONYMOUS ACCOUNT CREATION

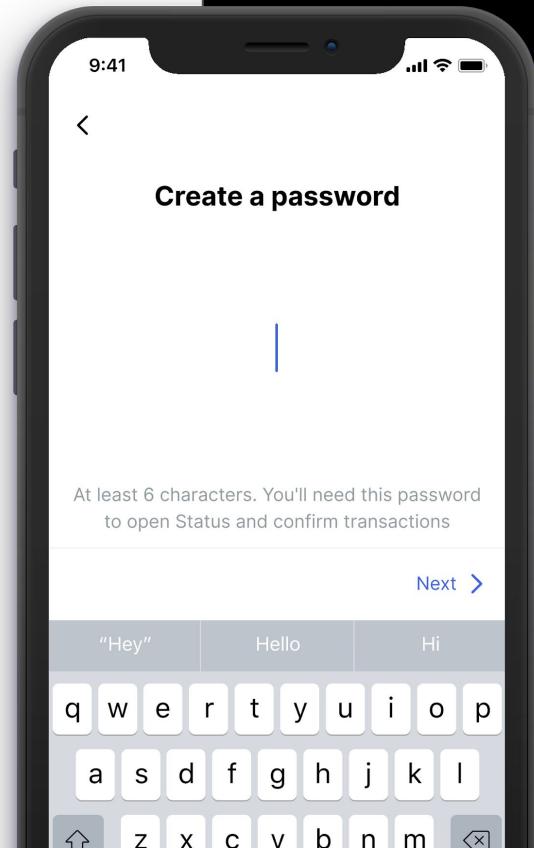
Creating A Password

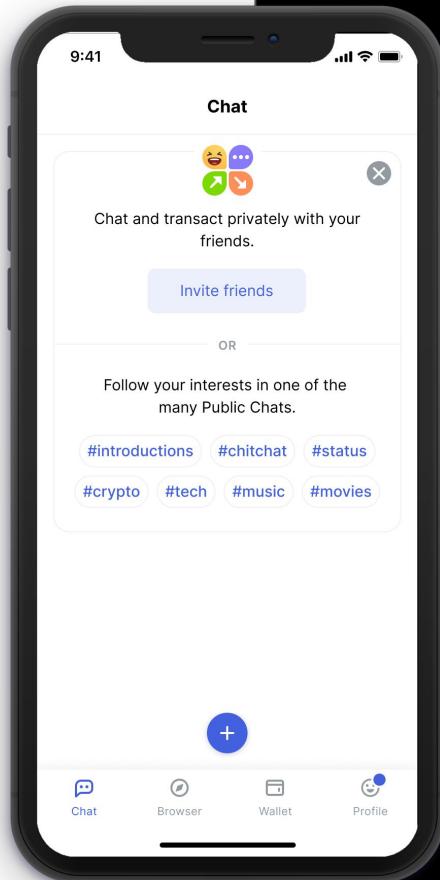
Many people wonder what the password is for if they already have a seed phrase.

While your seed phrase is randomly generated and used to provide access to your private keys, your password is created by you, and superficially locks the app once your keys are on your device. In day-to-day use, you will use the password to unlock your Status account.

It acts much like the passwords you use in current web2 products, but as with your keypair, it's stored locally on your device. We can not change or provide access to your password. However, it's not as important as your seed phrase. If you lose your password, you can always recover your account again using your seed phrase and create a new password of your choice.

Passwords are required for login, as well as to verify transactions. You will be prompted to enter your password when signing or transferring anything on the blockchain.





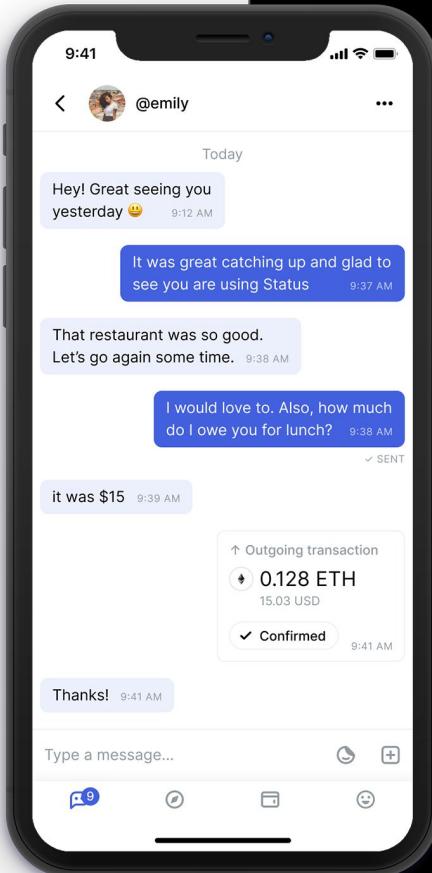
PRIVATE MESSAGING

Core Features

Like other popular messengers available today, Status offers private 1:1 chats, group chats, and public channels

Private Chats - completely private and censorship resistant chats between peers. Without centralized servers and the standard client/server model, no one can access messages, nor any metadata (e.g. who's sending, where) - not even Status.

Public channels - entirely public without restrictions to the number of participants. Public channels are based on whisper topics - more on that in a bit.



PRIVATE MESSAGING

Integrated with the Wallet

The Status messenger is seamlessly integrated with the wallet to enable users to send borderless crypto payments to other peers.

Using chat commands, users can access their wallet from a private 1:1 message, select one of their cryptocurrencies, and sign a transaction to send some to their friend.

Users can also purchase Stickers from the decentralized sticker market using SNT. Stickers are then available for use in all types of chat.

As noted before, chat keys and wallets keys are not connected. If I know your chat key, I do not know anything about your wallet, or your funds, unless you share that address with me as well.

PRIVATE MESSAGING

Whisper & Dark Routing

Status currently uses Whisper – the messaging protocol of the Ethereum stack. It is a new protocol that provides p2p messaging with zero reliance on central servers, data centers and service providers. Whisper aims to achieve complete darkness. Even with encrypted communications, well-funded attackers are able to compromise your privacy by tracking metadata. Darkness means that in the standard mode of operation, messages cannot be tracked or inspected and do not leak any metadata. Every message is broadcast to every peer in the network, making patterns between them impossible to understand.



whisper

PRIVATE MESSAGING

Whisper & Dark Routing

Whisper leverages the following features to ensure darkness and the prevention of metadata leakage:

- **Peer-to-peer:** leverages Ethereum's DΞVp2p wire protocol to gossip messages around the network.
- **Multicast broadcasting:** When sending a message to a specific person, a user actually broadcasts that message to the entire network. Because it is encrypted with the intended recipient's public key, only the designated recipient can view the message.
- **Proof-of-work:** Currently, Status sets a lower PoW value for Whisper messages than other clients default to, as higher PoW uses too much battery and compute power for most mobile phones. This is an active area of our research.
- **Time-to-live:** A message will bounce from node to node in the network even after it reaches its intended recipient until its time to live has expired - this provides plausible deniability that the message was intended for a specific recipient.
- **Envelopes:** As the name suggests, contain all the parts of a message. They can be compared to physical letters - all the nodes, regardless if they are the recipient or not, need to be able to handle the envelope, but only the recipient node knows how to "unseal" the actual message text. They are a packet format that encapsulates message text, along with metadata like time-to-live, and the message itself. Envelopes themselves are not encrypted, as they are part of the protocol and need to be readable by the nodes.
- **Topics:** Topics are short strings - hashes to be precise - which are set by the sender (or at the application layer) and help categorize messages. In more technical language: topics are cryptographically secure, probabilistic, partial-classifications of the message.

PRIVATE MESSAGING

Replacing Whisper with Waku

Whisper is not without its limitations and challenges. Status is the only instance of Whisper being used in a real production environment, so we've made some ad hoc modifications to support our use case.

What are these fundamental issues? In short:

1. Whisper is not scalable, most pressingly when it comes to bandwidth usage—there are not enough active Whisper nodes to support great volume of use.
2. It has low spam resistance: proof-of-work is a poor mechanism for Whisper's heterogeneous nodes.
3. Lack of incentivized infrastructure leads to reliance on Status' own nodes as centralized choke points.

4. Lack of formal, clearcut specification makes it hard to analyze and implement.
5. Whisper runs over devp2p, which limits where it can run and how.

In the coming months, Status will supplement the current implementation of Whisper with a fork of a protocol called Waku.

Waku is being developed by the Vac team at Status, which conducts R&D on a modular, peer-to-peer messaging stack for private, secure, censorship resistant communication.

Learn more about Vac and Waku [here](#)

PRIVATE MESSAGING

End-to-End Encryption by Default

Even with encrypted communications, well-funded attackers are able to compromise your privacy. Darkness means that in the standard mode of operation, messages cannot be tracked or inspected and do not leak any metadata.

The Whisper protocol also sends messages end-to-end encrypted by default—no need to toggle this as a setting.



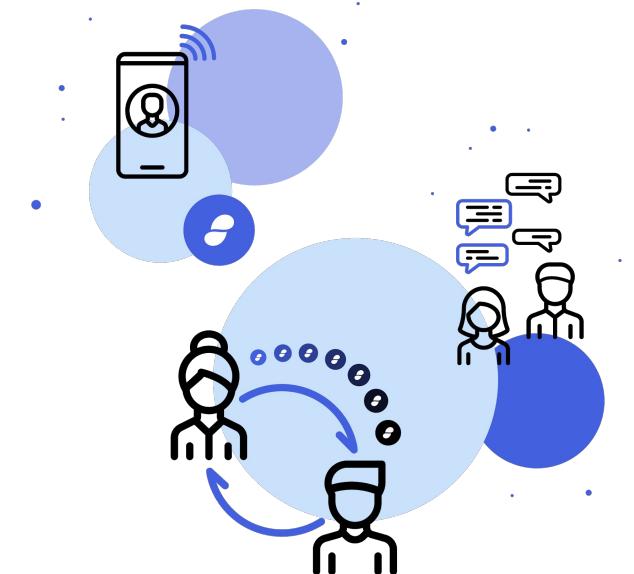
PRIVATE MESSAGING

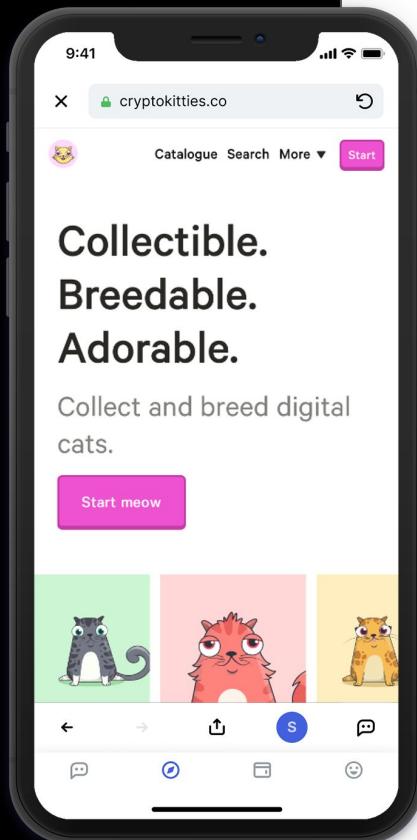
Perfect Forward Secrecy

Status adds another layer of privacy to your private 1:1 messages with perfect forward secrecy. If a user's encryption keys are compromised, not only are future messages at risk, but previous messages are as well. With perfect forward secrecy, past messages are protected.

Perfect forward secrecy is a feature of specific 'key-agreement' protocols which ensure that your keys for a given session will not be compromised, even if the private keys of the participants are. With perfect forward secrecy, past messages cannot be decrypted even by a third-party who manages to get a hold of a private key.

Perfect forward secrecy means that the encryption system automatically and frequently changes the keys it uses to encrypt and decrypt information, such that if the latest key is compromised, it exposes only a small portion of the user's sensitive data.





INTEGRATED CRYPTO WALLET

ERC20 & ERC721 Tokens

Status implements an entirely non-custodial ethereum wallet, meaning that users are responsible for their tokens. Unlike a bank, Status does not hold, store, or maintain any tokens. Only the holder of a wallet's private key can access their tokens—hence the need for seed phrase backup!

Status supports all ERC-20 tokens. If a desired token is not listed, users are able to add custom tokens to their wallet using the token's contract address.

Status also supports ERC721 collectibles, or non-fungible tokens (NFTs). Users can send and receive NFTs such as Cryptokitties directly in their Status wallet. The list of supported NFTs is currently limited. To request additional NFTs, ping the team in [#status](#).

INTEGRATED CRYPTO WALLET

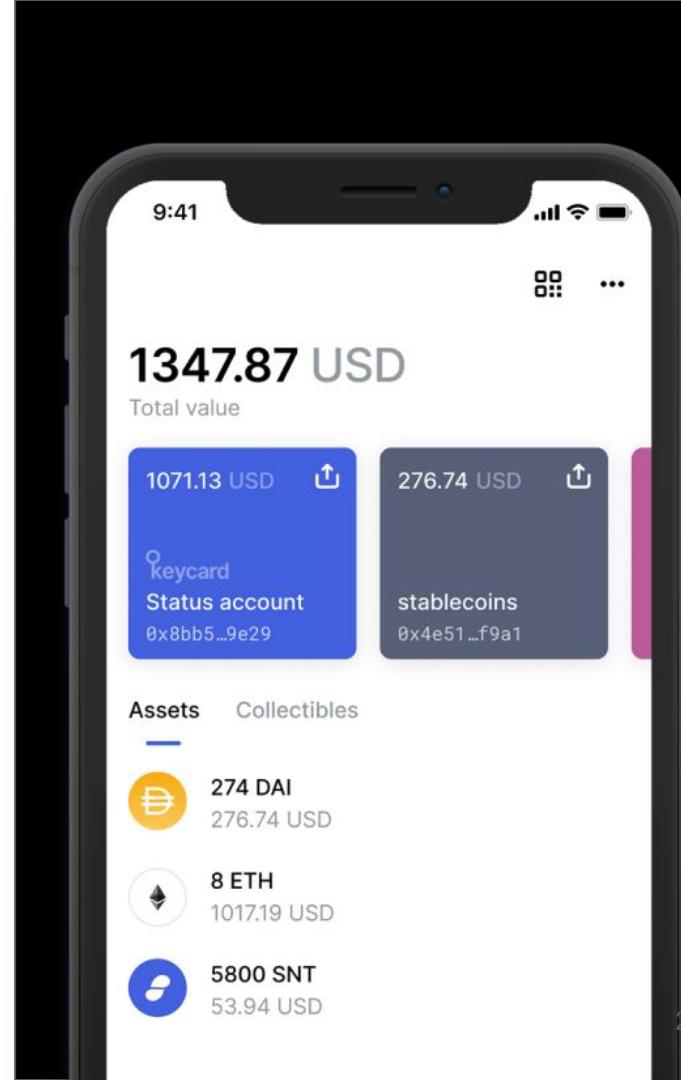
Multi-Accounts

The new Status multi-account structure allows users to manage multiple wallet keys in one place. This means that, just like a bank account, users can maintain a primary account for transacting and interacting with DApps, as well as other, separate accounts for saving, trading, etc. These accounts can all be accessed from a single Status wallet.

When making a transaction in Status, users will have the option to select from their various accounts.

Users can also easily send funds between their accounts.

This feature provides further privacy for users, as they can more easily spread their funds across multiple Ethereum addresses.



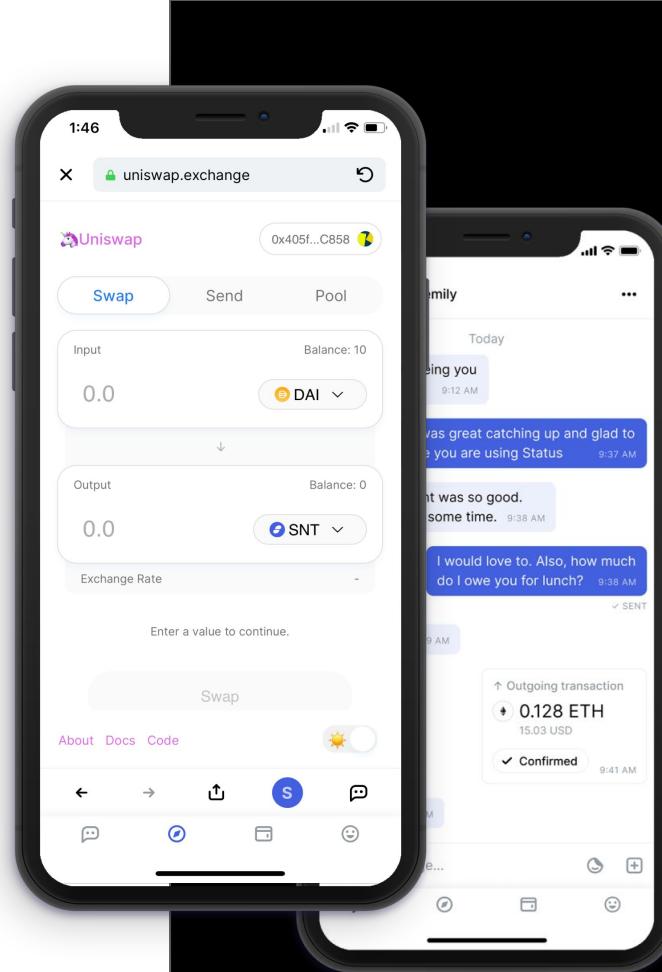
INTEGRATED CRYPTO WALLET

Connected to Messenger & Browser

The Status wallet is seamlessly integrated with the messenger and browser - enabling users to easily send tokens (ERC20 & ERC721) directly within a private chat.

Users can also purchase stickers within the decentralized sticker market for use in public, private, and group chats. All purchases of stickers made are available within the chat interface.

The wallet is also integrated with the DApp browser for simple in DApp purchases and exchanges.



INTEGRATED CRYPTO WALLET

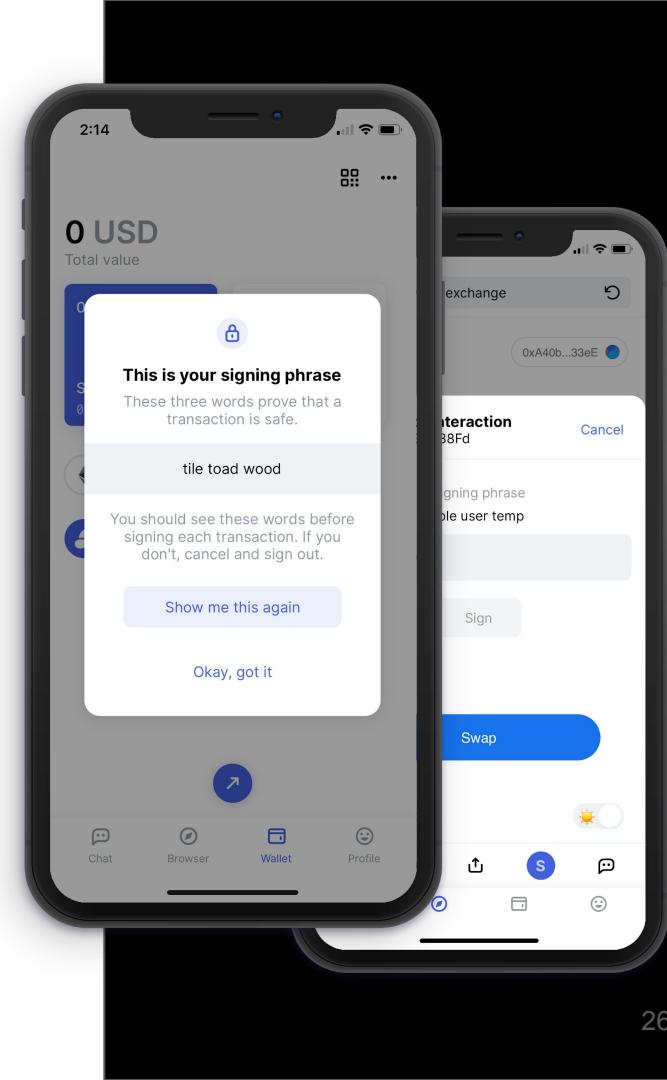
Signing Phrase

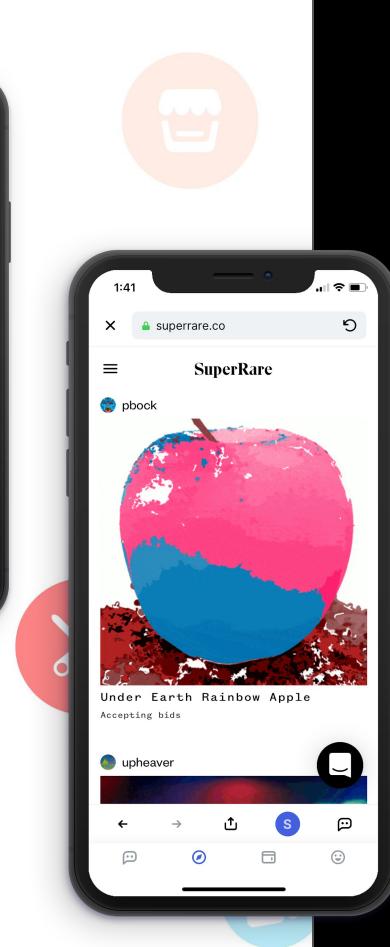
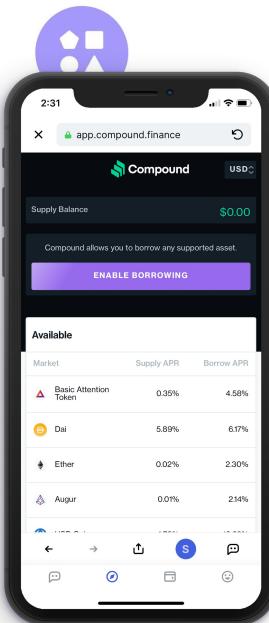
The signing phrase is an added layer of protection for your wallet and funds. It is an anti-phishing mechanism, designed to protect you from malicious actors intercepting a transaction. You'll see your unique three word signing phrase when you set up your wallet for the first time. Commit this phrase to memory and check for it each time you sign a transaction.

The signing phrase confirms that a transaction is being generated by Status and that the receiving party is as specified.

If you see the wrong signing phrase, cancel the transaction and log out of Status. Verify that you have a build provided by an official source: the Play Store, App Store, or <https://status.im>.

The signing phrase is auto-generated and stored only on the user's device.





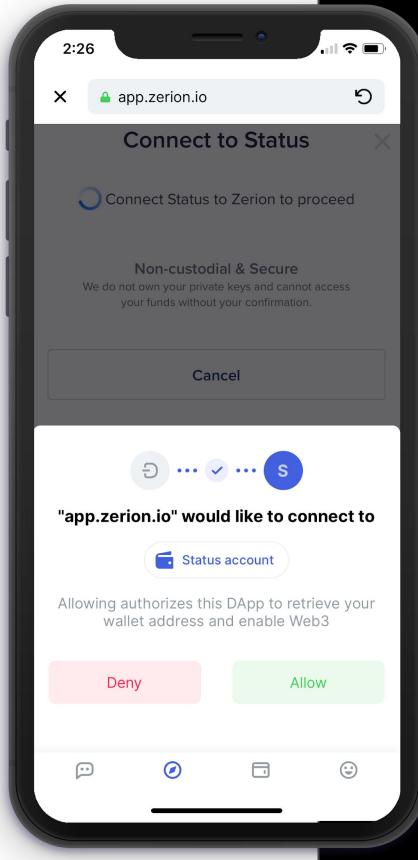
PRIVATE WEB3 BROWSER

Access a Growing Ecosystem of Dapps

Status includes a web3-enabled browser, so users can access and interact with their favorite decentralized applications, such as Kyberswap, MakerDAO Oasis, Cryptokitties, Airswap, etc. Any of the growing ecosystem of exchanges, marketplaces, collectibles, games, social networks, and more can be used in the Status browser. So can traditional websites!

The browser is privacy-preserving. None of your activity is tracked, though sites you have visited are stored in memory and listed for your convenience.

The browser is integrated with the crypto wallet for in-app purchases and exchanges.

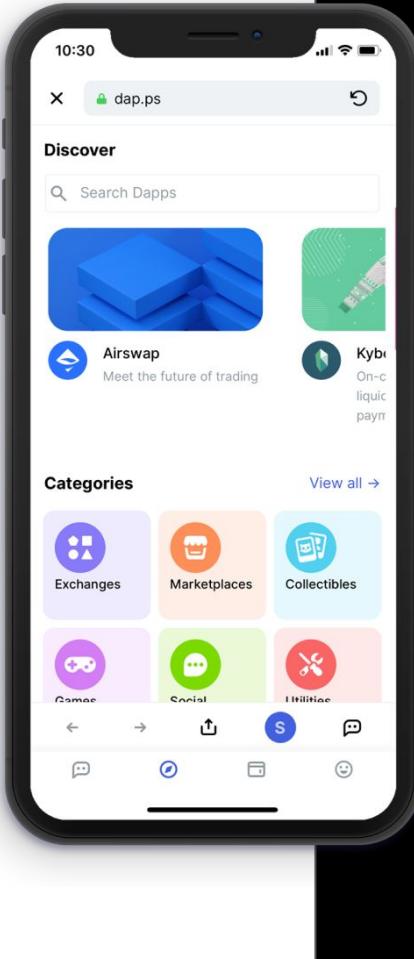


PRIVATE WEB3 BROWSER EIP1102 & Privacy Features

The Status browser uses Ethereum Improvement Proposal 1102 to prevent DApps or other websites from skimming information about your Ethereum account. EIP 1102 is a privacy standard requiring DApps to get your permission before they can see your personal data or interact with your account.

Before you grant permission, a DApp can know only that you are running a web3-enabled browser. When the DApp wants to initiate transactions to your account, it will send a permission request in the Status browser.

This flow closely resembles other mobile app permissions, e.g. for camera usage. Once authorized, the DApp will remember your setting. You can edit or revoke permissions for a given DApp from your profile.



PRIVATE WEB3 BROWSER

Dap.ps - The Token Curated Dapp Directory

Status maintains <https://dap.ps> - an independent, token-curated DApp directory. DAp.ps is a decentralized list of Ethereum DApps, helping users to explore web3.

To be listed, DApp developers can submit their DApps stake SNT to boost their ranking. To balance their influence, the more SNT a developer stakes, the less SNT they can get back and the easier it is for a community member to downvote that DApp.

For their part, users can use SNT to upvote or downvote DApps, resulting in a fair, community driven DApp directory.

Status Network Token (SNT) Utility

Status Network Token (SNT) Utility

Ethereum Name Service Registration

Decentralized Sticker Market

Dap.ps

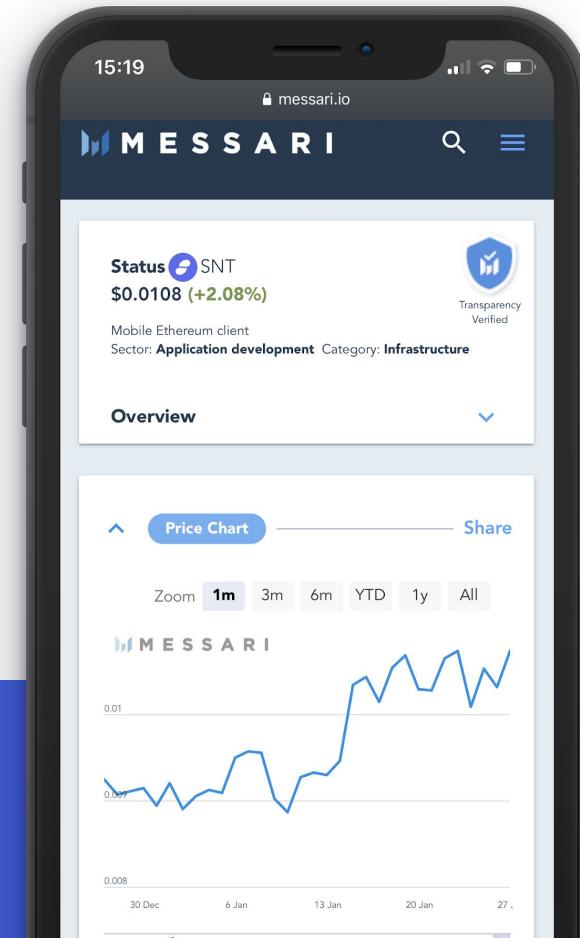
Status Voting Dapp

SNT UTILITY

The Status Network Token

SNT powers and incentivizes participation in the Status Network. It is an ERC 20 token that is used to access and power decentralized services in The Status Network and application.

To learn more about the Status Network Token and its distribution, check out the listing on the [Messari Disclosures Registry](#)



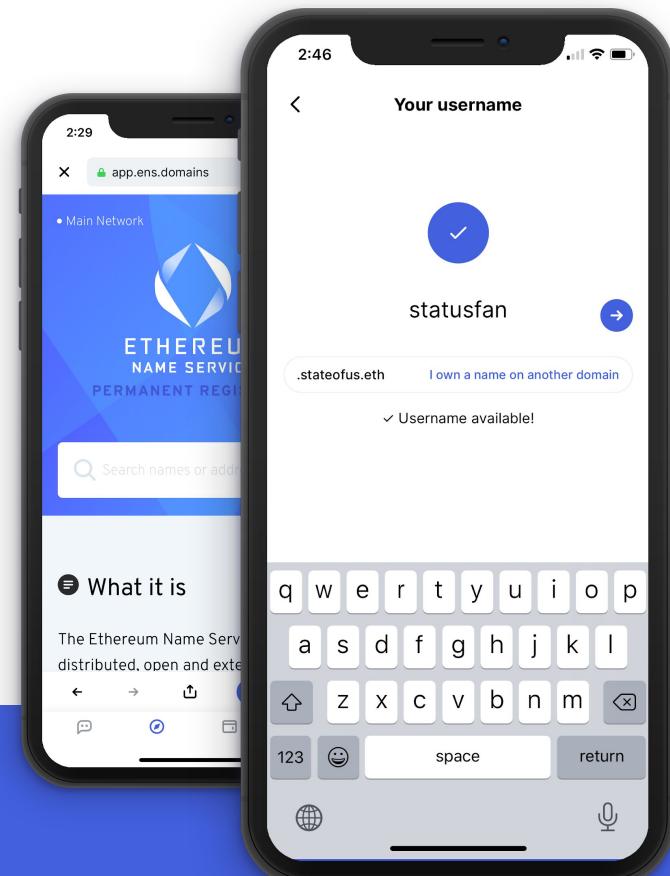
SNT UTILITY

Ethereum Name Service (ENS Registration)

Status users can register a stateofus.eth user name for easy searchability and recognition within Status. ENS names replace the long hexadecimal hash of your chat key with a human readable name of your choosing.

ENS names can be reserved from the profile screen for a nominal cost in SNT. You can display your ENS name in place of your three-word chat name once registered.

Any .eth address can be used within Status, regardless of whether you register your name through Status or another service. The ENS section of the profile will help you get set up.



SNT Utility

Decentralized Sticker Market

Status App includes a decentralized Sticker Market. Stickers offer a fun way for users to visually interact with their friends and family, and have been shown to increase engagement levels within messaging apps, particularly among millennials.

The first version of the Status Sticker Market includes one Status created pack and one community sourced pack. The community pack is free to download and use in chat, while the Status pack costs a small amount of SNT.

Packs can be obtained from within the chat UI and the commands tab.

Creators keep 100% of the proceeds from their stickers. You can submit a pack using [this form](#).



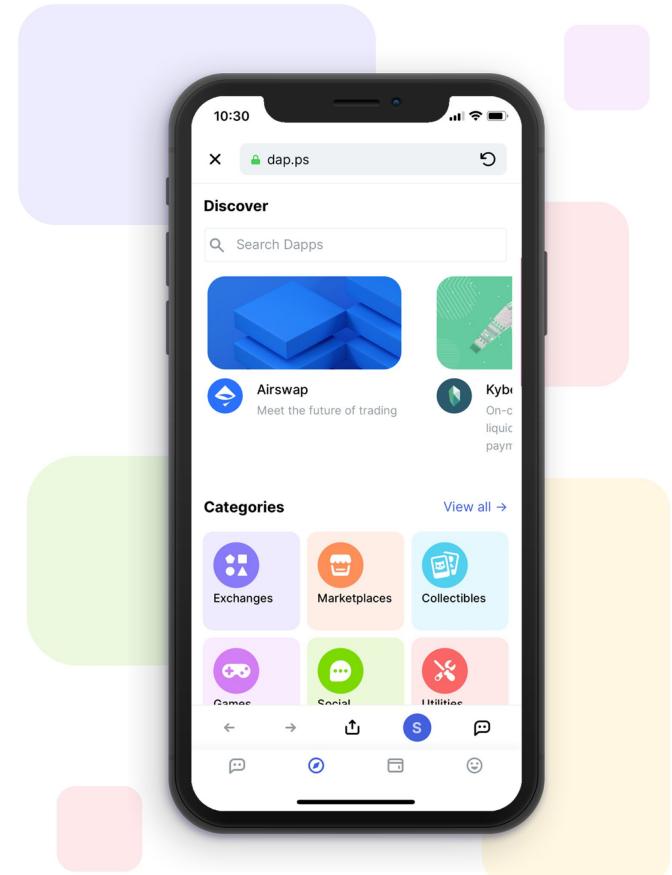
SNT Utility

Dap.ps

Dap.ps is a third party application that is integrated into the Status Mobile App. Dap.ps provides access to a list of the best, most useful dapps to Status users.

Dapp developers can submit their dapp to Dap.ps and stake SNT to be ranked higher in the listings. However, there is a catch to staking and ranking: the more SNT a developer stakes, the cheaper it is for a community member to downvote that dapp and the less SNT they can withdraw.

With Dap.ps, users can upvote and downvote their favorite dapps with SNT. Creating an game-theoretically stable, open, and fair ranking system.



SNT Utility

Status Voting Dapp and Community Governance

DAP.PS is a third party application that is integrated into the Status Mobile App. Dap.ps provides access to a list of the best, most useful dapps to Status users.

Dapp developers can submit their dapp to Dap.ps and even stake some SNT to be ranked higher in the dapp listing. However, there is a catch to staking and ranking, the more SNT a developer stakes, the less SNT they can get back and the easier/cheaper it is for a community member to downvote that dapp.

With Dap.ps, users can upvote and downvote their favorite dapps with SNT. Creating a fair, community managed dapp listing.



Technical Specifications

Technical Specifications

Serverless Tech Stack:
Status-react & Status-Go

Bootnodes

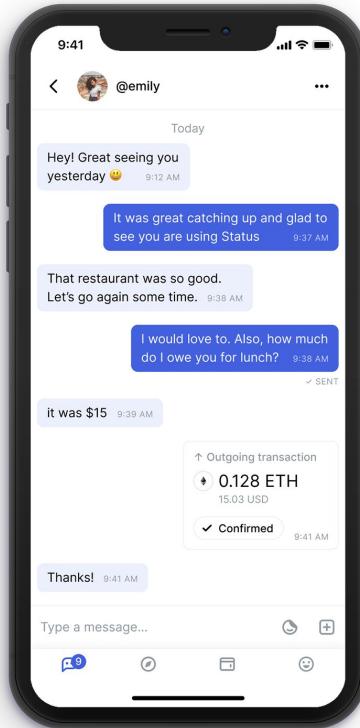
Mailservers

Serverless Tech Stack

status-react
UI, dApps, UX logic

Status-react - [React](#)

- iOS/Android: ClojureScript -> ReactNative
- Desktop: react-native-qt
- UI code (screens, chat input, widgets, DApp webview etc.)
- Chatbot API
- Interactions with DApps
- Think of it as the “frontend”



Status-go - [Go](#)

- Go, uses go-ethereum, can be compiled into standalone server or static libraries to link with iOS/Android, Node.js etc.
- All blockchain-related code (sync, validation, data storage etc.)
- Takes care of “heavy” underlying logic of being an actual blockchain node.

status-go
Blockchain logic, transaction, JS VM

Specifications Repo

For more information about the technical specifications

- see the Specifications repo on [Github](#)
- Join us in the #status public channel

```
<?php if (have_comments()) : ?>
<?php printf(_nx('One response to %2$s&rdquo;',
'comment-list">
<?php wp_list_comments(['style' => 'ol', 'short_ping'
/>);

<?php if (get_comment_pages_count() > 1 && get_option('
    <ul class="pager">
        <?php if (get_previous_comments_link()) : ?>
            <li class="previous"><?php previous_comments_li
        <?php endif; ?>
        <?php if (get_next_comments_link()) : ?>
            <li class="next"><?php next_comments_link(__('Ne
        <?php endif; ?>
    </ul>
</nav>
<?php endif; ?>
<?php endif; // have_comments() ?>
<?php endif; // comments_open() && get_comments_number() != '0
<div class="alert alert-warning">
    <?php _e('Comments are closed.', 'sage'); ?>
</div>
```

Forward Thinking

Forward Thinking

Chat improvements

Whisper Upgrade

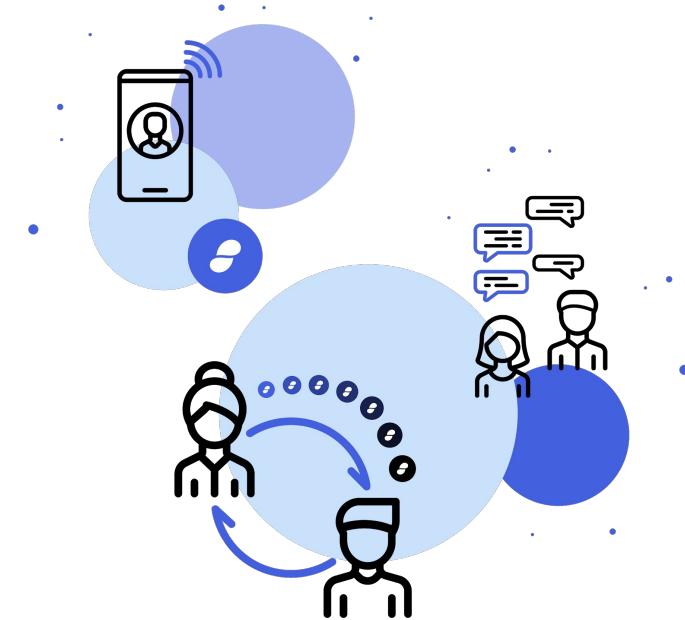
SNT Utility Features

Performance Optimization

Chat Improvements

Status is a social communication tool and one of the primary features is a peer-to-peer messenger. Status v1 enables users to chat with their friends and communities in a decentralized manner but the team plans on adding a number of new features to achieve feature parity with the most widely used messengers available today. These features include:

- Group Chats
- Sending images in chat
- Push notifications for Android
- Mark as read feature with unread message counter



Replacing Whisper with Waku

Whisper is not without its limitations and challenges. Status is the only instance of Whisper being used in a real production environment, so we've made some ad hoc modifications to support our use case.

What are these fundamental issues? In short:

1. Whisper is not scalable, most pressingly when it comes to bandwidth usage—there are not enough active Whisper nodes to support great volume of use.
2. It has low spam resistance: proof-of-work is a poor mechanism for Whisper's heterogeneous nodes.
3. Lack of incentivized infrastructure leads to reliance on Status' own nodes as centralized choke points.

4. Lack of formal, clearcut specification makes it hard to analyze and implement.
5. Whisper runs over devp2p, which limits where it can run and how.

In the coming months, Status will supplement the current implementation of Whisper with a fork of a protocol called Waku.

Waku is being developed by the Vac team at Status, which conducts R&D on a modular, peer-to-peer messaging stack for private, secure, censorship resistant communication.

Learn more about Vac and Waku [here](#)

SNT Utility Features

Status V1 includes a number of SNT utility features as stated earlier in this document. We will be introducing some new features the bring further use to the token including

- **Network Incentivization:** a framework and incentivization structure for people to deploy and use Status nodes. SNT is used to incentivize and reward network participants to run a node in order to ensure the app continues to work if all nodes in the Status-hosted cluster are down. This includes paying for forwarded messages and paying for offline messages.
- **Tribute to Talk:** an economics-based anti-spam filter – in this case for receiving messages and “cold” contact requests from users. This enables stakeholders to set a minimum amount of SNT that must be deposited in order for someone outside of their network contact him/her directly.
- **User Acquisition Engine:** Using the Status Network Token, we can design mechanisms for growth that have been tried and tested in Web 2.0, whilst providing greater transparency and better alignment of incentives for participants. The Status Acquisition Engine provides a cryptographically provable and transparent means of growing the user base.



Performance Optimization

We will continue to make improvements to the Status Mobile app that improve the overall user experience of the application such as:

- Optimize battery consumption
- Reducing network data consumption
- Improving wallet fetching transaction history
- UI performance, specifically on older Android devices



Join Us in Status

<https://status.im/get/>

Join our Community Discord Server

<https://discordapp.com/invite/3Exux7Y>

Status Press Kit

<https://statusnetwork.com/press-kit/status/>

