Stensitive Whitepaper

1 Abstract

Stensitive is an open source and free to use technology to interact with the Stellar blockchain in order to store sensitive data on it.

By sensitive data, we intend every kind of data that the user does not wish to have publicy available.

Even though data on blockchain is publicly available by design, with stensitive, your data will be encrypted in a way you, and you only, will be able to read and modify it by signing transactions through your wallet. The technical workings of Stensitive are further described in the ”How it Works” section.

1.1 Example

Carlos wants to store sensitive data in a way he will be the only one with access to it, and so that no entity will put its data at risk. He starts the stensitive app (or any app that uses the Stensitive npm package), enters the data he wants to store (along with a name and a pin for additional security), and signs two transactions. His data is now stored encrypted on the blockchain.

2 Encryption

Encryption is the key concept of Stensitive, the goal was to build it in a way that only the user, by signing a transaction with their wallet, would be able to read and modify their data.

This was achieved by leveraging the nature of a transaction's signature: **unique and unpredictable depending on the user's secret key and the signed transaction's body**. By this definition, encrypting data with a transaction's signature as the encryption key allows the encrypted data to be only retrieved by signing the transaction again to re-obtain the signature. This process obviously assumes that the transaction used to retrieve the signature to encrypt the data is never submitted to the network.

To wrap it all up, the encryption process under the hood to upload the data is:

1. The User sings a transaction that has a security pin as transaction sequence number.

2. The signature from the signed envelope is used as encryption key to AES encrypt the User's data.
3. The transaction will not be sent, rather the user will now sign and submit to the network a new transaction that adds the ipfs hash that contains the user's data to the user's account data attributes. This way, the encrypted data is stored on IPFS and linked to the user's Stellar account.

When the user wants to read this data:

1. The user gets the encrypted data from IPFS
2. The user signs a transaction that has the same sequence number as the pin used to encrypt it (see step 1 for uploading the data). Basically, we are creating the same transaction the user had signed to get the secret key that encrypted the data, then the user signs it to get the decryption key.
3. The data is now decrypted.

3 Security

By design, data stored with this technique is secure as long as the stellar blockchain is. However, there are still a couple of things to take into account before using the technology:

- when implementing this technology in your app, the transaction that will generate the secret key for the AES encryption **does not have to get submitted to the stellar network**.
- the usage of the pin prevents the user from phishing attacks, it is recomended to make it a non-optional paramenter. That's because, if a malicious actor ever asked the user to sign an empty transaction without a pin, it would obtain the encryption key for the user's documents. With the pin as additional parameter, the user can notice better if the website is trying to steal their information.

4 Using Stensitive

This very document has been edited and stored using Stensitive's LaTeX live editor, meaning that the document was stored securely encrypted on Stellar before being published.

The usage of a wallet to store and load data every time is not user-friendly for users that are not used to the world of Cryptocurrencies. However, that is the price to pay if you don't want to store your data relying on a third party.

To help spreading user-owned data to the Stellar community and not only, we have developed a simple helper npm package: `stensitive`. This helper will be well-documented in our docs, it allows to speed up the process described in section 2 and integrate it with every Stellar wallet.

5 Open Source & Free-To-Use

We believe that user-owned data will be very important in the next years (where we are going towards an excessive centralization of corporations owning user-data). Furhtermore, our concept is not perfect and was developed in a really short time-window, we believe that making Stensitive open-source will help the stellar community to maintain and make it even better.

6 Conclusion

We have discussed the principles of how we see user-owned data on the Stellar Blockchain. We encourage you to visit Stensitive web app to start using it, or if you are a developer, visit our documentation for the stensitive npm package.