**IPFS**

A blackchin can store data, however, the distributed ledger technology is not designed to do so. The biggest problem with storing data in a blockchain is an issue with scalability and related transaction cost. In addition, query data from a blockchain can be slow due to the mentioned scalability issue. IPFS is an alternative secure and immutability storage solution, which was designed to store data.https://malcoded.com/posts/storing-data-blockchain/

IPFS is a peer-to-peer (p2p) storage network. Content is accessible through peers located anywhere in the world, that might relay information, store it, or do both. The main idea of the IPFS is to have many nodes all over the world synchronize their data, therefore the storage solution is immutable.

**CID**

IPFS knows how to locate what you ask for utilizing its content address (CID) instead of the location of the data. CID is a label used to point to material in IPFS. The CIDs are based on the content’s cryptographic hash function, therefore Any difference in the content will produce a different CID. The same content added to two different IPFS nodes using the same settings will produce the same CID. IPFS splits content into blocks and verifies them through directed acyclic graphs (DAGs), SHA file hashes will not match CIDs. However, the CID is still deterministic, uncorrelated, unique and a one-way function.

https://docs.ipfs.io/concepts/content-addressing/#identifier-formats

**Immutability**

An immutable object is an object whose state cannot be altered or modified once created. Once a file is added to the IPFS network, the content of that file cannot be changed without altering the [content identifier (CID)](https://docs.ipfs.io/concepts/content-addressing) of the file. However, when it comes to content that needs to be altered or updated, immutability becomes a problem. Currently, the IPFS system has no version control(https://github.com/ipfs/notes/issues/23) opened in 2015

**Source:** [**https://docs.ipfs.io/concepts/immutability/**](https://docs.ipfs.io/concepts/immutability/)

**Install:**

This document will present how to set up a privet-IPFS network. The privet IPFS (Inter Planetary File System) network does not connect to public IPFS nodes.

Use of docker for easier system administration.

**Security:**

In the real-world application of this project, many nodes would be used. To have Geo-redundant storage. In order to establish a connection between nodes the internet would be used, which could open the system for cyber-attacks. Therefore, the system needs to be hardened, our recommendation is two uses of Defense-in-depth and the zero-trust model.

Firstly, IPFS has a private key system called swarm key to protect the network. Only nodes which the right swarm key can connect to the IPFS network, however, if the key leaks any bad actors can connect to the network. It might be a good idea to change the key regularly in a secure way.

Secondly, nodes should have a firewall rule which blocks all incoming traffic unless the packet is originated from a known node IP address, furthermore, the same rule should be implemented in a network firewall. To implement a zero-trust type environment all outbound traffic needs to be filtered, only connections to known IPFS nodes should be allowed.

For a higher level of security, additional security systems are recommended such as IDS/IPS and an application-level firewall as well which should be monitored 24/7 in a SIEM. (Physical security of the servers)site to site VPN?