DecenHash Project v1.0

By Arthur S. Sacramento

decenhash@gmail.com

Introduction

The objective of this work is to show concepts that can be used in the construction of a decentralized file storage system, where each server earns according to its computational power that is used to perform validations, storage, etc. Unlike others, its difference from competitors is that the file sharing layer is different from the monetization or property layer. In other words, the user does not need to run a heavy system, they can run only the file transfer layer where there is no monetization or rewards or run only the monetization or rewards layer where there is no transfer of the original files but only their hash and other metadata.

Empirically, create a system that facilitates file searching even across different servers. Develop a means or mechanism where a file server receives profit according to its computing power. And the user can run their own server or node, even if it is a simplified version and in HTML.

SHA256 standard for naming files: Files are saved using their SHA256 hash as the name plus the original extension. Files can be saved within the 'files' directory or in the subdirectories of 'data' (where each directory stores the original file itself and only the hash addresses of other files that relate to that file).

File flow: A large server can search for files even if they are hosted on other servers. A server's profits or earnings are proportional to its computing power.

Each file is associated with a hash: This hash can be a category, the identification of the user who sent the file, or any other data or criteria related to the file.

Each hash can be associated with another hash: Each hash, which can represent, for example, an image or conversation, can be associated with other hashes that correspond to similar images or responses to a conversation, resembling graphs or trees with branches.

Storage based on directories and hashes: Each directory corresponds to a hash and it will store several hash files (for example, the directory can represent the hash of a file and all the hashes inside the directory indicate comments about that file or related files).

Search by hash: When searching for a word, for example, it will be converted into a sha256 hash and in each user or node will be searched a directory with the name of that hash. This directory will contain an "index.html" file with a list of hashes or original file names.

Simple decentralized search: The objective is to offer a means to perform simple decentralized searches, even if the user's server does not support a database and only HTML. Thus, it will already be saved in a directory (whose name is a hash) the file or category that the user searched and a list of hashes and files that are related to that file or category.

System optimized for searches in nodes or by category: The way in which the files are organized does not prevent a database from being used to make the search more complete. But, for example, if the user only wants to download a file and all the hashes associated with it then he does not need to query the database, but only directly open the directory and all the files within it. The same happens with the search, he directly opens the HTML file without the need to query the database. In other words, this architecture aims to make it more efficient for the user to download a list of files from a directory or consult a list of hashes than to make a query in a database with many entries or records.

Metadata layer: The user can register a hash address, adding additional information to the file such as name, description, etc.

Monetization layer: The user can receive rewards according to the number of nodes that store their file and the relevance of those nodes, among other criteria.

Minimalist sharing option: The user can download all files from a server and upload to it. This process can be done on a list of servers.

Challenges and difficulties of common servers:

1) Monetization X Transfers - The lack of incentive for the user to participate or sharing. Generally, systems excel at offering a service but fail to integrate a good file transfer system with one of their monetization. In other words, there is a lack of incentive for the user to share a file in a centralized system or host it on their own computer as a node.

2) Lack of a standard for search: Although it may be useful for a server not to have a search tool to increase user privacy, it would be an interesting idea for a large server to be able to search for files on others through a common interface for search.

3) Excessively complex and heavy systems: There is no doubt that nowadays there are many decentralized solutions for file storage. But, generally, they are heavy because they try to do everything in one thing, being a barrier to entry for new users or clients. Our solution to this is that there is a layer only for file transfer and others for more advanced operations, giving the user the flexibility to choose which layer they want to run.

4) Absence of a verification system: As there is no broad standard for storing files and their names, it is difficult to verify which servers host the same file. And to build upon this idea, a monetization or reward model that favors the file owner or user the more it is distributed.

Conclusion:

Many of the limitations we currently have seem to arise because we still don't have the right tools to deal with them, whether at the software or hardware level. It is a mistake to try to solve a problem with the wrong tool. However, such questions or limitations do not prevent a conceptual model from being built, illuminating at least part of the problem. Nowadays, there are already several solutions that can serve as direction or inspiration, such as graph-oriented databases, decentralized technologies, or even the rise of Bitcoin. We want something new to emerge like Bitcoin, but focused on file transfer. A kind of Über or Bitcoin for file transfer, combining practicality with an efficient monetization model.

References

Nakamoto, S. (2008). ***Bitcoin: A Peer-to-Peer Electronic Cash System***.

Nelson, T. H. (1981). ***Literary Machines***. Sausalito, CA: Mindful Press.

Lévy, P. (2001). ***Cyberculture*** . Minneapolis: University of Minnesota Press.

Deleuze, G., & Guattari, F. (1987). ***A Thousand Plateaus: Capitalism and Schizophrenia***. Minneapolis: University of Minnesota Press.