NAR Kickoff Document

Description

Our product will provide free, secure, private and limited-by-contribution file cloud storage system to its users. Main problem that we want to solve is centralization of file clouds. We want to create a system that is not traceable by any company or group and secure and unlimited by design.

Users will push files to the system. Upon push request, the file will be encrypted by a key that is known only to the user and will be fragmented into little pieces. The user will ask for a list of peers from the central entity to push his pieces to them. The central entity will coordinate the connections between peers. To provide high availability of files, the central entity will create redundancy by instructing peers to share pieces with other peers.

Authentication of individual users will be provided by the central entity. Users will provide proofs to the central entity in order to show that they are indeed them.

When a user wants to get his files back, he will connect to the central entity with his proof-of-identity and requested file and piece id. The central entity will provide a connection with a peer that has the requested piece of file. After connection, the piece will be downloaded directly from the peer. When the user collects all pieces of a file, the pieces will be defragmented and decrypted. The original file will be available to the user.

When a user wants to list the files that he has in the system, he will ask the central entity to give a list of his files providing proof-of-identity. The central entity will the return with the list of files that user has in the cloud.

When a user wants to delete a file from the cloud, he will ask to the central entity to remove the requested file. Upon receiving the request, the central entity will remove the pieces related to the deleted file by instructing peers to delete pieces.

When a user wants to update a file, the delete and add file procedures will be followed one after another.

Work Packages

WP	Term	WP Title	Estimated
1	491	File Preparation and Reconstruction	4
2	491	Connection Establishment	4
3	491	Connection Protocols Definitions	6
4	491	Database Design and Operations	4
5	492	Peer Selection with Redundancy	6
6	492	Redundancy Maintenance	12

WP	Term	WP Title	Estimated
7	491	Implementing DHT as Database [BONUS 1]	8
8	492	Decentralizing Central Entity [BONUS 2]	8

Detailed Description of High-Level Work packages

WP1 – File Preparation and Reconstruction

1. Encryption of file
2. Compression of encrypted file
3. Fragmentation of compressed file
4. De-fragmentation of pieces
5. Decompression of de-fragmented pieces
6. Decryption of decompressed file

WP2 – Connection Establishment

1. NAT Traversal Server Side
2. NAT Traversal Client Side
3. Direct Connection Server Side
4. Direct Connection Client Side

WP3 – Connection Protocols Definitions

1. Peer Registration
2. Peer Authentication
3. Peer-to-Central Entity and Central Entity-to-Peer Coordination
4. Peer-to-Peer Piece Transmission

WP4 – Database Design and Operations

1. Mappings: piece-to-peer mapping, file-to-file owner mapping, available disk space-to-peer mapping, available network quota-to-peer mapping.
2. Uptime-to-Peer Mapping: This mapping will define the estimated uptimes of peers.

WP5 – Peer Selection with Redundancy

1. Selecting peers depending on the number of pieces that will be pushed and uptimes of peers.
2. Assigning multiple peers to pieces to establish redundancy.

WP6 – Redundancy Maintenance

1. Redistribution of pieces in case of peer leave.
2. Redistribution of pieces in case of a new peer join.

WP7 – Implementing DHT as Database [BONUS 1]

1. Distributing database to peers and connecting peers
2. Handling peer churn
3. Implementing a query system

WP8 – Decentralizing Central Entity [BONUS 2]

1. Redefining connection protocols
2. Distributing peer selection
3. Distributing redundancy maintenance

Risk Assessment

Risk #	Description	Possible Solutions
1	Not being able to implement NAT Traversal.	Flood the traffic through a centralized server.
2	Not being able to authenticating users when the central entity is decentralized.	Implement a distinct authentication server.
3	Not being able to detect malicious users in decentralized central entity case	Stick with centralized model.
4	Not being able to select peers that provide maximum availability of files.	Choose them heuristically.