SERVERLESS DISTRIBUTED APPLICATION FOR BACKING UP AND FILE SHARING

PROJECT PROPOSAL - 24/04/2015

ANDRÉ PIRES
FILIPE BATISTA
FILIPE MIRANDA
JOÃO BANDEIRA

Purpose of the Application

For this second assignment the group proposes the implementation of an application based on the first project, capable of backing up files (chunks) on different computers (serverless system). On top of the first project, we will implement some enhancements regarding security, fault tolerance, among others.

Besides backing up files, this application could also be used to share files (entire file) with a group of users, who would have access to the most recent version of that file.

Main Features

Features from the previous assignment (improved as explained in the follow section):
- Backup files
- Recover files
- Delete files
- Reclaim space

Plus new features:
- Sharing complete files with circles of friends
- Editing simple files (.txt) in real time collaboration (extra)
- Files version control

Enhancements for the first assignment Distributed Backup Service:

1. Implement the basic protocols for an Internet version (peer-to-peer)
2. Fault tolerance (survive peer crashes)
3. Security

   o Secure connections
   o Peers authentication
   o Chunk confidentiality
   o Possibility of choosing to whom the chunks would be sent - trusted peers would be less likely to try to hack your files

Web Services

- 3rd party web services

    - API 1: Facebook
        - authentication
        - download profile information and friends list

    - API 2: Google+
        - authentication
        - download profile information and friends list

- Provided services by our server

    - Peers Finder

        - Find peers to connect, either randomly or from social networks' friends lists. Everytime a peer logs in, a list of logged-in people will be updated on the server. Everytime a user wants to send a file to a group of friends, the server will verify if those friends are currently online.

    - Temporary storage

        - Ideally, every transfer should be done P2P, however, if one peer chosen to receive a shared filed is currently offline, the server will store a temporary copy until that peers logs in, and then immediately sends that file.

Target Platforms

- Java standalone application for PC
- **Possibly** for Mobile Devices (Android only)

Additional Services and Improvements (if time permits)

- Development of a GUI
- Real-time editing