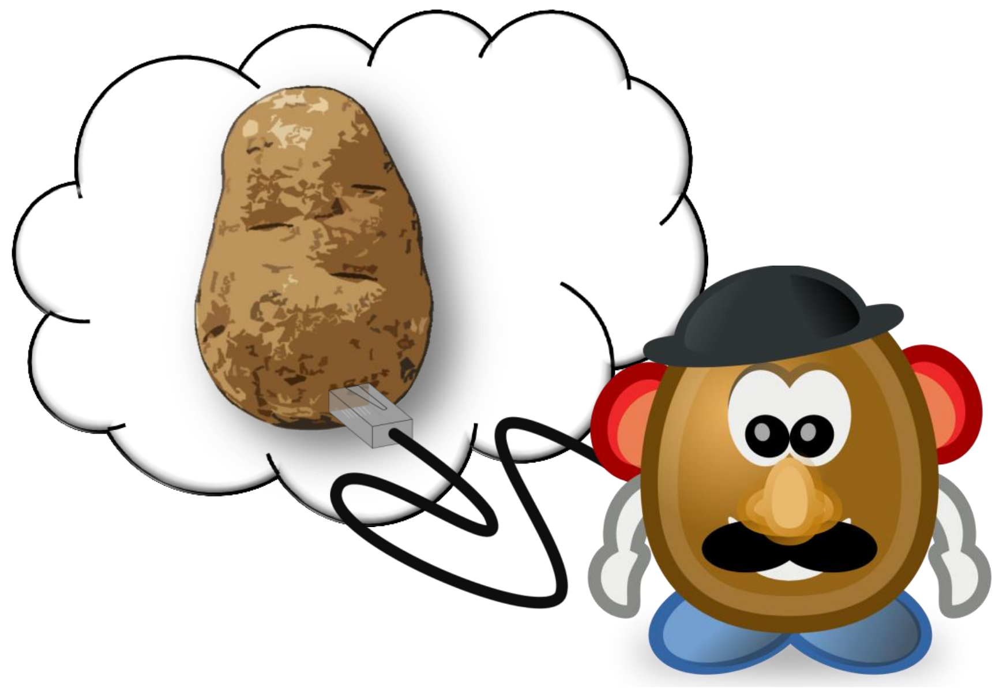
P_2Potato

A Distributed Dropbox using Isis^2

Michael Nazario and Jason Wang



Security

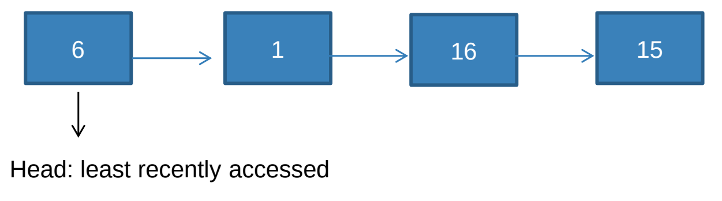
- Password-based authentication
  - Hashed using SHA-512 and a randomized salt
- All files are encrypted
  - Uses AES with password-salt-hash
- Password hashes are encrypted
- Shared folders accessible through common password
- All files stored have their filenames hashed and salted

Consistency

- Timestamp based concurrency
  - Latest timestamp wins
  - Arbitrarily greater address breaks ties
  - Prevents inconsistency
- Assumes generally synchronized clocks
- Allows one hour of wiggle room between clocks
  - Prevents malicious user from preventing future changes

Availability and Replication

- Using Isis^2 DHT
  - Enforces replication of files
  - Available as long as enough computer are on
- Server-side cache
  - Most recently-used policy
  - Other heuristics could be implemented
  - Ensures often read data such as folder contents are available



MRU Queue of Keys

Key	Value
16	Data…
1	Data…
6	Data…
15	Data…

Cached Data Table

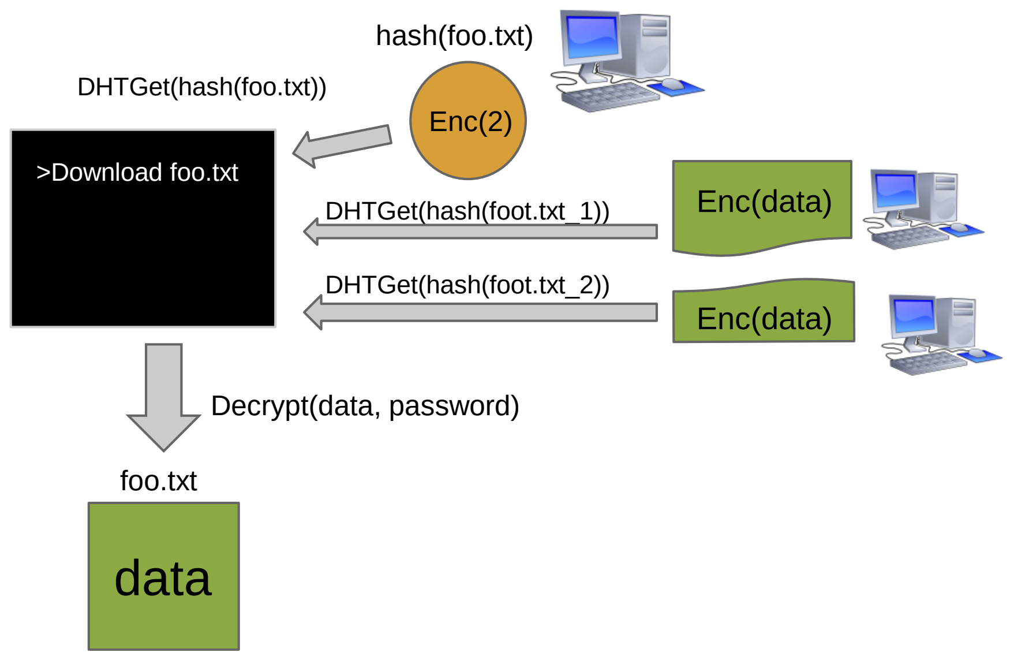
Overview and Features

- Using the Isis^2 DHT to create a distributed cloud storage system
- Main goals are Consistency, Security, and Availability
- Offer users a virtual filesystem
  - One private partition accessible only to the user
  - Many shared partitions (folders) secured with a separate password
- Console and GUI interfaces for a Unix or Windows feel respectively

File Storage in the DHT

- An abstraction above the Isis2 DHT
  - Not just a simple key/value store
  - Store all files on disk and not memory
- Files are broken up and stored
  - Broken up into ~4MB fragments and stored as key/value pairs
  - Lookups for a file go to a base node that specifies how many fragments to look up
  - Uploads that shrink the file will clean up leftover fragments

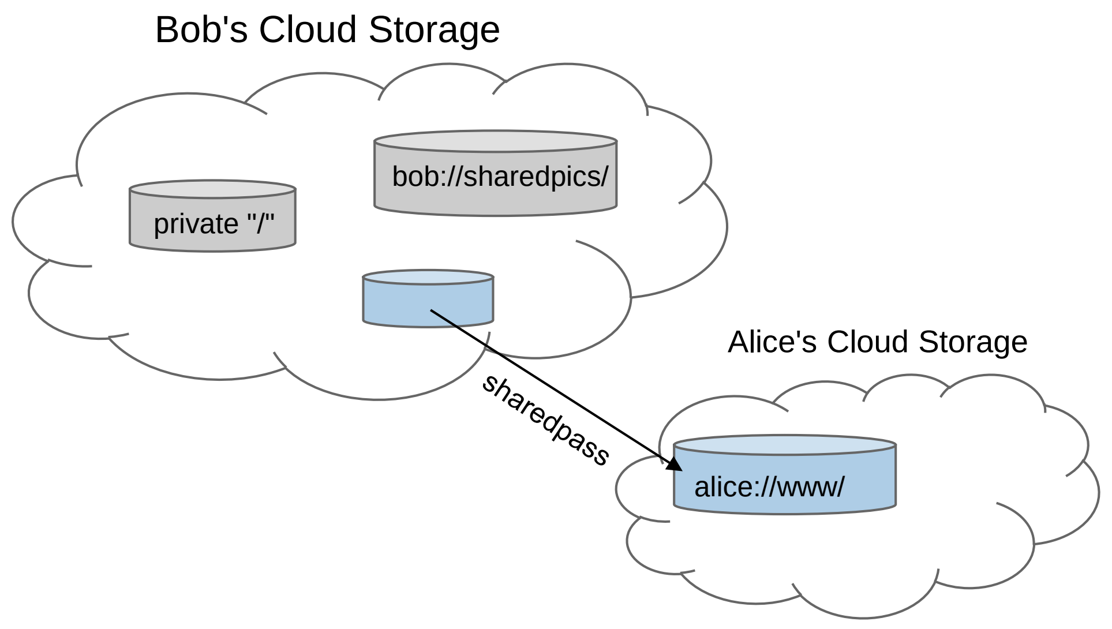
Download File Example



Virtual Filesystem

- Root directory
  - "/" for private directories
  - "user://foldername/" for shared directories
- Directory structure
  - Current implementation is just a list of other directories and files in that directory
  - For better scalability, a Unix inode implementation would be desired
- Shared folders (partitions)
  - Users enter a shared directory by giving a host username, folder name, and a shared password
  - Leaves own private partition and enters "shared mode"

Filesystem Visualized

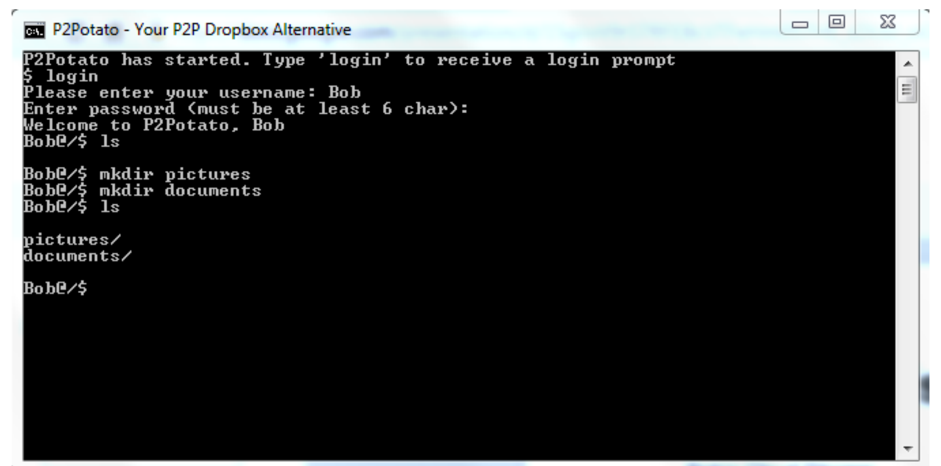


Isis2 Modifications

- Added DHTSetHandlers function
  - Allows arbitrary storage locations (e.g. disk)
  - Integrates into the Isis2 API easily
- Discovered bugs in DHT
  - Simple log calculation issues
  - Inability to update new members of DHT
- Future modifications
  - Kelips-like gossip protocol to ensure consistency in updates in the affinity group

P2Potato Implementation - Terminal

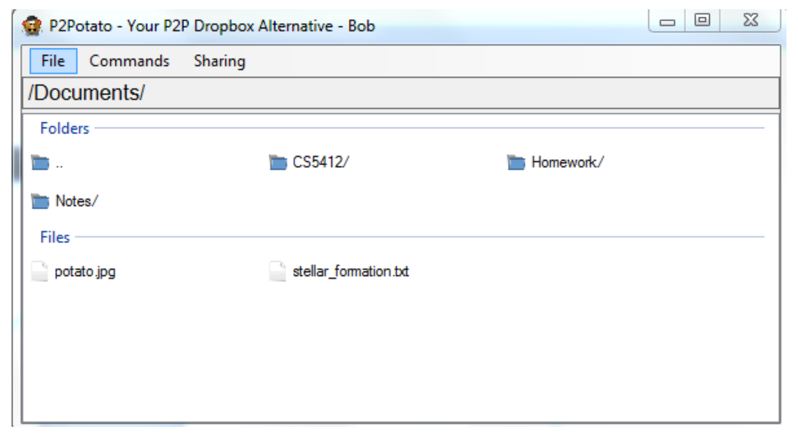
- Command line
  - Familiar filesystem to linux users
  - ls, cd, mkdir, etc.



```
P2Potato has started. Type 'login' to receive a login prompt
$ login
Please enter your username: Bob
Enter password (must be at least 6 char):
Welcome to P2Potato, Bob
Bob@/$ ls

Bob@/$ mkdir pictures
Bob@/$ mkdir documents
Bob@/$ ls

pictures/
documents/

Bob@/$
```

P2Potato Implementation - GUI

- P2PotatoGUI
  - Emulates a Windows explorer experience
  - Runs in the background when closed
  - Easy for the general consumer


P2Potato - Your P2P Dropbox Alternative - Bob
File
Commands
Sharing
/Documents/
Folders
..
CS5412/
Homework/
Notes/
Files
potato.jpg
stellar_formation.txt