Vapor-Ware ™ : Data, up in the Air !

Karthik Narayan , Rohit Koul, Sandeep Dhoot, Vinod Ramachandran

Application Description

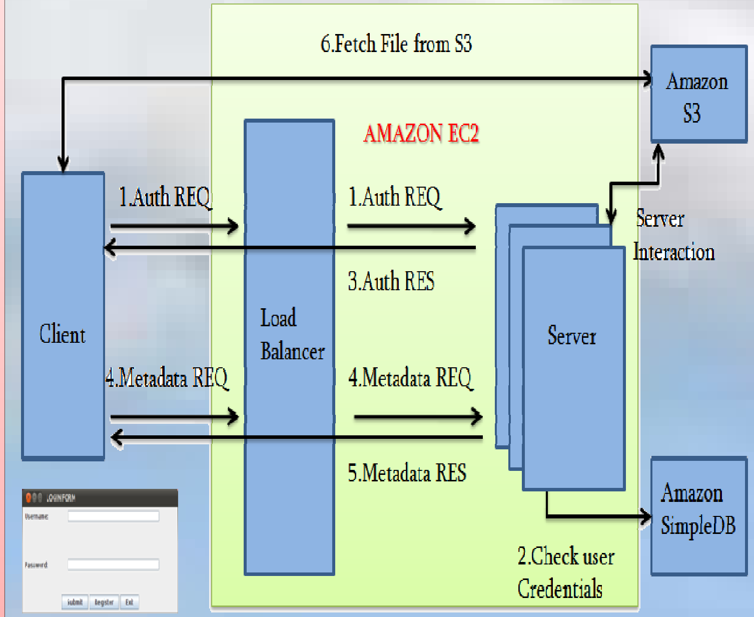
- File backup and sharing service served using Cloud Infrastructure
- Server runs inside Amazon EC2 AMI
- User files are stored in Amazon S3
- Authentication database stored in Amazon SimpleDB
- Client and Server interact using HTTP

Features

- **Metadata Server :** Client talks to the Server ONLY for Metadata
- **Stateless Server :** All metadata is stored in S3 which means that any request can go to any server
- **Direct File Transfer to S3** : Client directly communicates with S3 for file transfer, there by reducing the load on the server
- **Client-side Conflict Avoidance :** Client avoids conflicting updates to S3's copy of file using Version number returned by server, any conflicting files are renamed and notified to user through email
- **File Sharing :** Sharing / Un-sharing supported at granularity of Directories
- **No Redundancy :** Only single copy of shared directories stored in S3
- **Encryption :** Support for 128 bit AES and 64 bit DES Encryption

Reliability and Fault Tolerance

- Stateless server simplifies Fault tolerance
- Periodically saving of VersionMap to persistent storage allows client to gracefully handle crashes
- Amazon's Load Balancer detects server failures and distributes all requests among healthy server instances
- Auto Scaling helps to scale up and down in response to changing load on the system
- User is linked to a client thereby allowing the client to resume its activities on crashing.



Evaluation Metrics

- Resource consumption by desktop client and the Server
- Fault tolerance (Client and Server Crashes)
- Time taken to synchronize files

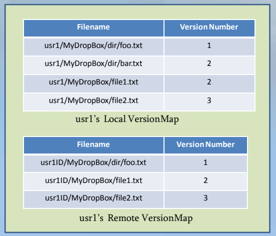
Metadata Structures

NO SHARING

- VersionMap (.versionMap) stores version number for each file
- Local VersionMap is modified by client, Remote VersionMap is modified by server on every write request to server
- Periodic syncing with server done using it.

SHARING (usr1 shares with usr2)

- On sharing, entry for shared directory added to usr2's VersionMap
- usr2 maintains SharedVersionMap to track version numbers of shared files
- All shared files stored(modified) remotely in usr1's MyDropBox directory in S3
- On un-sharing, entry for shared directory removed from usr2's VersionMap, SharedVersionMap cleared as necessary, all previously shared files newly created in usr2's MyDropBox directory

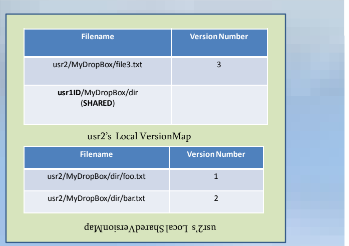


Filename	Version Number
usr1/MyDropBox/dir/foo.txt	1
usr1/MyDropBox/dir/bar.txt	2
usr1/MyDropBox/file1.txt	2
usr1/MyDropBox/file2.txt	3

usr1's Local VersionMap

Filename	Version Number
usr1ID/MyDropBox/dir/foo.txt	1
usr1ID/MyDropBox/file1.txt	2
usr1ID/MyDropBox/file2.txt	3

usr1's Remote VersionMap



Filename	Version Number
usr2/MyDropBox/file3.txt	3
usr1ID/MyDropBox/dir (**SHARED**)	

usr2's Local VersionMap

Filename	Version Number
usr2/MyDropBox/dir/foo.txt	1
usr2/MyDropBox/dir/bar.txt	2

usr2's Local SharedVersionMap

Future Direction

- Sharing at a file level
- Scripting support and better GUI
- SSL Support
- Better Flexibility and Config options
- Optimizations (Batch Query / Caching)

CS 739– Project 2