

Bit Torrent Based Downloader for Cloud Based Storage

Project Proposal

by
Nishant Mehta
Punit Sharma
Shandeep K Murugasamy
Vivek Chauhan

Project Advisor
Professor Ashok Banerjee

Ashok Banerjee
Ashok Banerjee (Nov 9, 2014)

October 31, 2014

Table of Contents

Chapter 1 Introduction

- 1.1 Problem statement, motivation, and needs
- 1.2 Project application and impact
- 1.3 Expected project results and deliverables
- 1.4 Market research
- 1.5 Project report structure

Chapter 2 Background and Related Work

- 2.1 Background and used technologies
- 2.2 State-of-the-art
- 2.3 Literature survey

Chapter 3 Project Description

- 3.1 Project goals and objectives
- 3.2 Project approaches
- 3.3 Project solutions and justification
- 3.4 Project evaluation and validation

Chapter 4 Project Plan and Schedule

- 4.1 Project tasks and schedule
- 4.2 Project resources, budget, and cost analysis

Abstract

Bit Torrent Based Downloader for Cloud Based Storage

Nishant Mehta, Punit Sharma, Shandeep K Murugasamy and Vivek Chauhan

High availability, speed and security are the major constraints in any cloud based file storage solutions. The cloud servers are the single point of failure and downtime will stop all the services. Also, the current systems don't provide a support to pause or resume the download. The other important factor is the security of user data. The proposed system will leverage the power of peer to peer sharing combined with encryption techniques to solve these problems.

The proposed system will use BitTorrent as the base to distribute the files with the cloud storage service. The cloud storage will act as a highly available seeder which can be used whenever the seeder ratio goes down. The files will be distributed into chunks, encrypted and distributed over internet to provide highly scalable and secured storage. Also it will allow resuming of download from where it was left. A tracker-less system will be developed where each peer has information about the neighbors having similar data and where each of them will acts as a tracker itself. The file chunks will be retrieved using an algorithm that ensures cloud storage is hit only for the rare bits of data and other chunks can be downloaded from a nearby peer, increasing the overall speed. Security features to avoid leakage or alteration of data will be implemented.

The proposed solution would provide high availability and speed while file sharing. The users will have more control on their data. Users can create torrents for their file without registering with any tracker, providing more privacy and security. Users would experience greater speed due to the power of peer sharing network. The system will provide a new scalable, high available and secure cloud file storage based solution.

Chapter 1 Introduction

1.1 Problem statement, motivation and needs

Currently Cloud based storage are most reliable way to store data. It is highly scalable, highly reliable and most convenient way to take backups. But Cloud computing is an efficient and highly scalable technology that enables enterprise information, software applications and currently all online data storage relies on ftp or raw http to download/upload the files/data to the server. This is not only unreliable and inefficient way to distribute huge amount of data but also insecure in many ways. Today data security remains the biggest hindrance to adoption of cloud technology.

Apart from this, clients who use traditional synchronization software often have problems with downloading huge files. Disconnection of network will lead to situation where client has to download the entire file again. It can be very frustrating when files are really huge can take hours or days to download. Also, some downloaders come with pause feature, but it can be paused only for short period of time or until the machines turned off. This again is very inconvenient for users.

The final problem was the environmental factor. Today data centers are the 3rd largest contributor of greenhouse gases in USA. Its impact is so huge that it is expected to surpass the greenhouse emission of airline industry by 2018. We need to focus on optimizing the data distribution to reduce the environmental impact.

1.2 Project application and impact

To tackle the above stated problems we decided you use the time tested protocols like BitTorrent to securely and efficiently distribute the data. Being encrypted, BitTorrent provides extra layer of security to the users. Also since it is peer-to-peer based technology, that allows one to download the data from its peers, one need not rely only on the cloud server to get the data. This allows one to download at much greater speed that the traditional way of downloading from server.

This will also help to reduce the overall load on servers. One can use much lesser resources to give much more efficient technology. This results into better economic returns. Apart from this, our system increases the efficiency of the content distribution system and allows much less carbon footprint due to less number of servers being used.

There are many applications of this system that still remains to be explored. It can help in creation of systems which required fast replication of data over a large network. Example of these systems can be network of servers which share real-time climate data over the world, requiring highly scalable and fast content distribution systems. Implication for the concepts stills remains to be determined.

1.3 Expected project results and deliverables

The proposed solution would provide highly scalable, reliable and secure data distribution system with high average download speed for end users. The users will have more control on their data. Users can create torrents for their file without registering with any tracker, providing more privacy and security. The software will comprise of the BitTorrent downloader for cloud with a tracker built inside along with mechanism to connect to public cloud services.

Chapter 2 Background and Related Work

2.1 Background and used technologies

BitTorrent protocol was introduced in 2001. The protocol deals primarily with the limitation of downloading speed in a peer to peer connection during data transfer. At this point of time the data was downloaded from a single source and the data transfer speed was governed by the internet speed at the source of data. BitTorrent protocol works in a different manner and allows users to download data from multiple sources. Parts of file requested by the user are collected from different users, transformed in usable format and transferred to the requesting user. It lead to revolutionary changes in internet history as it enables high download speed even if users can upload data at a slower speed.

Our application will be developed on similar rules with additional features of cloud storages, security etc. which will take BitTorrent protocol to next level. Our application will be more secure as there is no direct connection with other peers. This will ensure limited or no contact between peers. User will interact with a cloud which itself acts a peer and who enable user to interact with other peers indirectly though the cloud in focus.

2.2 State-of-the-art

There are some developments which makes use of **bittorrent** protocol to implement to implement similar ideas. Products like “Put.IO” which enables data download using torrent and works with torrent files rapidshare file etc and “Bittorrent Sync” which syncs data downloaded using BitTorrent to multiple systems. Both of these products have their own advantages. “Put.io” allows user to play with multiple file formats while “Bittorrent Sync” allows user to sync data over multiple machines. Other similar protocol is the “Gossip Protocol” which assumes that there’s no reliable communication and also the number of interactions among systems over network is low.

2.3 Literature survey

There are few papers return on cloud based flexible storage based on BitTorrent. Few such studies were done by, Rahma Chaabouni, Marc Sanchez-Artigas and Pedro Garc ´ ´ia-Lopez ´ at Universitat Rovira i Virgili, Tarragona (Spain) on personal cloud and enhancement using BitTorrent, a study on Cloud-based Content Distribution on a Budget was done by Francesco Albanese, Damiano Carra, Pietro Michiardi and Azer Bestavros. Another interesting paper was written on BitTorrent on Mobile Phones – Energy Efficiency of a Distributed Proxy Solution by Imre Kelényi, Ákos Ludányi and Jukka K. Nurminen. But none of these studies suggest an secure and flexible way of implementing file sharing system over cloud.

Though these ideas revolve around BitTorrent protocol for data transfer but they fulfill a different requirement. One of the primary differences between these ideas from our project is, they still make direct connections with the peers. This can lead to security

breach and unwanted interference from unreliable peers. In our project the idea is to let the cloud interact with the peer on behalf of the client. This adds another layer between the peer and the client that requested the data.

Chapter 3 Project Description

3.1 Project goals and objectives

The goal of the project is to overcome the limitations of current cloud based storage solutions. High availability, security, speed and cost are the major constraints in all of the present cloud based storage system. The idea of the project is to leverage the power of peer to peer communication combined with encryption techniques to address all these issues.

The main objective of the project is to provide an infrastructure which provides highly efficient, fast and secured mechanism for the storage solutions in the next generation cloud systems.

3.2 Project approaches

The project falls under the category of prototyping project. The present cloud based solution is based on Client-Server architecture, where only cloud server takes request from multiple clients. The project idea is based on a completely different architecture. The focus is on peer to peer networks which is far more powerful, scalable and efficient than traditional server side architecture. The viability of the project idea can only be tested by having a prototyping system.

The software methodology of object oriented design would be used to develop the system. The object oriented design has underlying principles of encapsulation, inheritance, abstraction and polymorphism. The advantages of these principles include:

- Code reusability
- Software maintenance
- Fast development
- Good quality software

3.3 Problem solutions and justification

The solution to the problem is building a peer to peer communication channel using BitTorrent protocol. The file to be uploaded will be first uploaded to the cloud storage using an uploader. Download of the file will be done using the downloader which will implement peer networking protocols. The file will be broken into chunks and will be distributed across multiple peers.

The downloader will have a tracker which will look for the nearest peers to download the file. The speed of downloading the file from a nearby peer is way better when compared with remote cloud server. For example, the request to download comes from San Francisco and the cloud server is in Oregon. The time to complete this download request will be more due to the network bandwidth and IO latencies. On the other hand, if the proposed approach is used, the tracker of the client would search for the nearest seeder (peer) which can be as near as 100 meters. The speed in this case would be high as the download is on local network.

The files would be broken into chunks and MD5 hashing would be done to ensure data integrity. By chunking the files into logical blocks, it is ensured that file can be distributed across various machines. The client can then download different chunks from different clients. Even if a peer goes down, the client can still resume the download. Also, the client can easily stop and resume the download of the file due to the logical chunks as the resume will be continued from the next chunk of data. The RSA algorithm will encrypt and decrypt the logical chunks. The process of encryption helps in securing and avoiding any data tempering.

The top most seeder will be the cloud server. The other seeders will first become leechers and then seeders. The idea is to ensure availability of data at any point of time. Even if all seeders go down, cloud server would still server the request to the clients.

3.4 Project evaluation and validation

There are various approaches to evaluate and validate the project.

Checksum : The checksum of MD5 hash received and expected should be same. If the checksum is same, it means that file has all the data as expected.

Regular Download vs Peer Based Download : The performance and outcome can be evaluated by comparing the system against traditional systems. The IO time and file integrity are crucial criteria in this evaluation.

Availability Test: The seeders will be stopped in a random fashion. This test would ensure that file download continues seamlessly regardless of the amount of seeders.

Server Load Test: The performance of server would be monitored by doing the load testing. The load test would send multiple requests to the server and would then analyze the CPU usage pattern.