P2P Networking Seminar Series:

FREE HAVEN

--- An Anonymous Storage System

By

Xiaoyang Guan & Quanhong Wang

Project Outline

1. Introduction
2. Anonymity for Anonymous Storage
3. System Design
4. Attacks on Free Haven
5. Comparisons
6. Conclusion
7. Future Work
8. References

Introduction

- Background/Origin:
  - **Began in December 1999**
  - **a research project by several MIT students**
- Developers:
  - **Roger Dingledine**
  - **Michael J. Freedman**
  - **David Molnar**

Goals of Free Haven

- Anonymity:(for all parties)
- Accountability:
- Persistence: the publisher of a document determine its lifetime
- Flexibility: servers can join and remove from the system smoothly

Some Useful Definitions:

- Author : **the entity who initially created the document**
- Publisher: **the entity who places the document into the system**
- Reader: **entity who retrieve the document from the system**
- Servers: **participants who provide special services required to keep the system running, such as disk space or bandwidth**

Anonymity for Anonymous Storage

- Why?
  - Allow individuals to speak freely without fear of persecution
- What is?
  - No link between documents and users
- How to realize?
  - Using pseudonyms to give each server a reputation, influences how much data a server can store and provides an incentive to act correctly

Different kinds of anonymity:

- Author Anonymity
- Publisher Anonymity
- Reader Anonymity
- Server Anonymity
- Document Anonymity
- Query Anonymity

Partial Anonymity

"is the system anonymous enough" instead of
"is the system anonymous?

Design Requirements

- Robust
- Simple
- Modular
- Decentralized
- Content-neutral
- Flexibility
- Free and open source (components)

- Supporting operations:
  - inserting documents
  - retrieving documents
  - expiring documents
  - adding servers
  - recognizing inactive or dead servers

System Overview

Publication system

- Servnet
  - distributed storage
  - peer-to-peer
  - dynamic
    - data moves from one server to another
    - servers join and leave

Communication channel

- Remailer network
  - Cypherpunk and Mixmaster remailers(Mixnets)

Elements of Publication System

- agents: author, publisher, server, and reader
- server
  - a public key and one or more remailer reply blocks(addresses)
    - provide secure, authenticated, anonymous communication
  - database of some other servers in the servnet
    - public key, reply block, trust

Publication

- Encrypt the file as desired
- Break the file into *shares* $f_1, \ldots, f_n$ where any k shares are sufficient to recreate the file (Rabin's information dispersal algorithm)
- Generate PK_{doc}/SK_{doc} for signing each share
- Build and sign a data segment for each share
- Insert those segments into the local server's space

Retrieval

Documents are indexed by $H(PK_{doc})$

- Reader locates a server for the query
  - generates a key pair PK_{client}/SK_{client} and a one-time remailer reply block
  - broadcasts a request $\{H(PK_{doc}), PK_{client}$, reply block$\}$
  - optionally specifies the desired share index
- Server receives the query
  - checks if it has any shares with $H(PK_{doc})$
  - If yes, encrypts each share using PK_{client}
  - sends the encrypted share through the remailer to the enclosed address

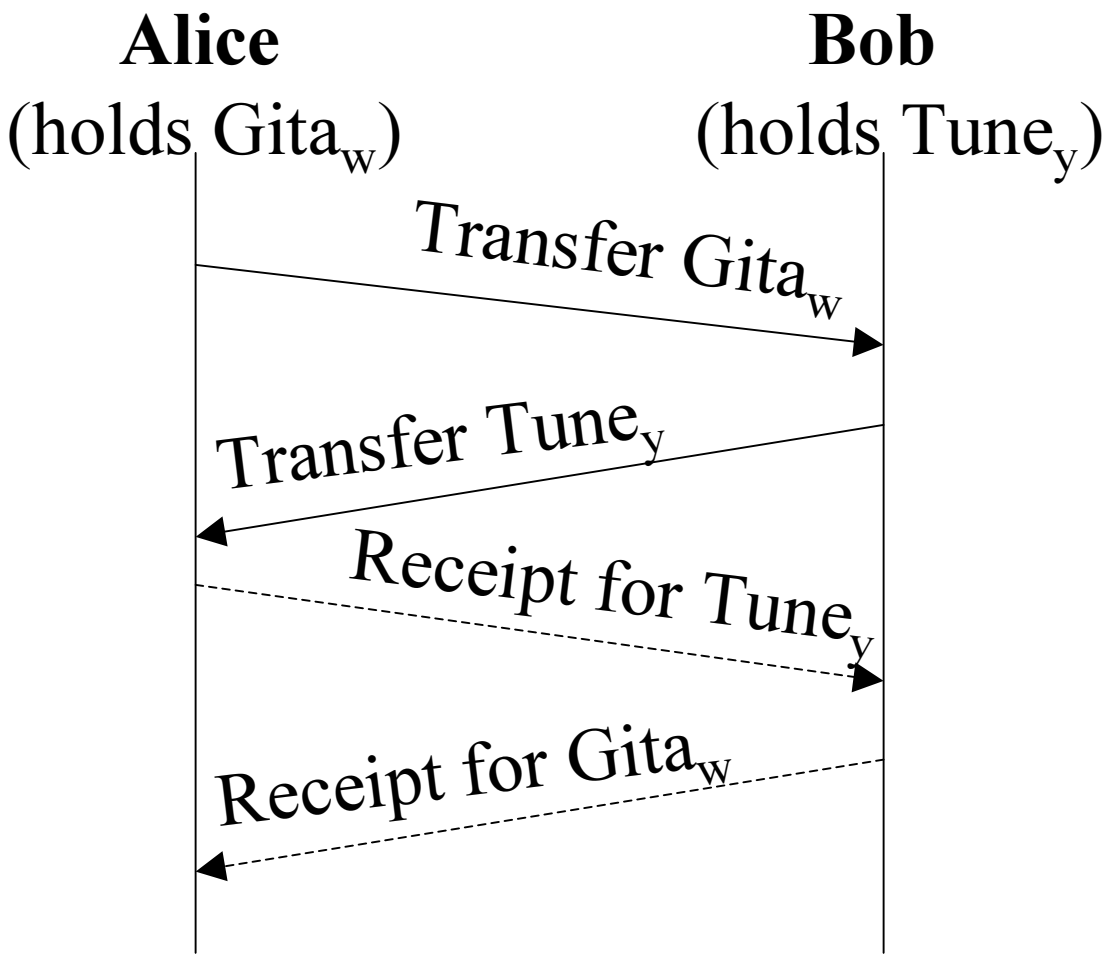
Share Expiration

- Absolute timestamp
- Determined by the publisher
  - how long the publisher wants the data to last
  - file size
  - likelihood of trading
- Wait until all local shares expire to quit the servnet

Trading

- To provide a cover for publishing
- To let servers join and leave
- To permit longer expiration dates
- To accommodate ethical concerns of server operators
- To provide a moving target

Trading Handshake



Issues in Trading

- Frequency
- "Fairness"
  - size * duration
  - trust between trading parties
  - build up reputation
- Keeping copy of share during negotiation
  - increases both overhead and robustness
  - response to query
  - optimum amount of time

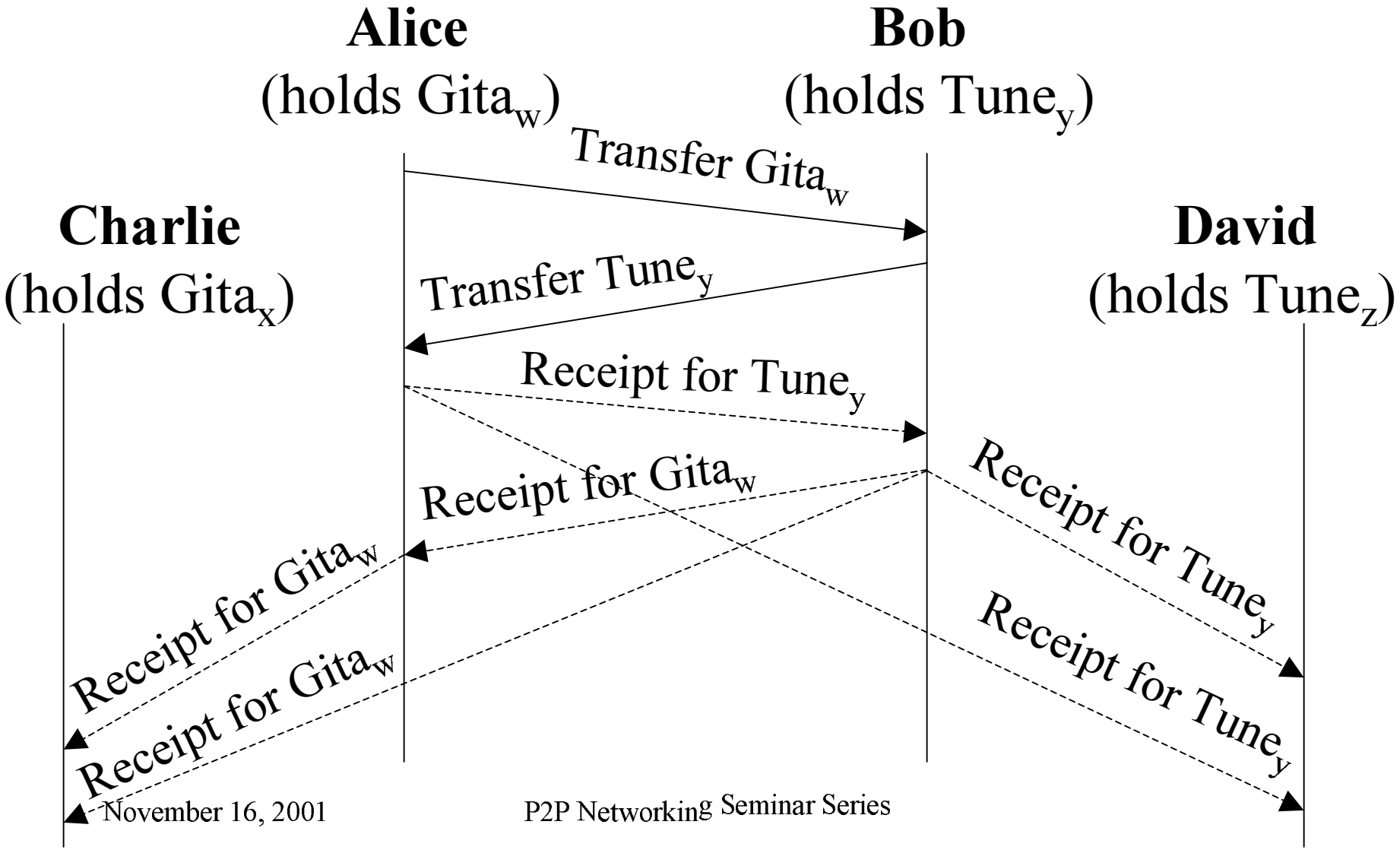
Accountability

Malicious nodes accept but do not store shares

Buddy system

- Association between pairs of shares (*buddies*) within a given document
- Each share maintains information of its buddy
- Each share periodically query for its buddy
- Buddy notifications when a share moves

Trading Handshake Revisited



Latency of Communication Channel

- Two *buddy notifications* sent to the buddy when a share moves
- Notification arrives after buddy also traded
- Forwarding buddy notifications using information in the receipt of the buddy
- Receipts kept until share expiration
- Forwarding *not* done for document requests

? *Buddy query*

Reputation System

- Misbehaviors
  - Neglecting to send a receipt after a trade
  - wrongly accusing another server of losing a share
- Trust management
  - *reputation* and *credibility*
  - confidence rating
  - broadcasting *referrals*

How do servers discover each other

Introducers: servers with a good reputation

- New servers contact introducers
- Introducers broadcast referrals of the new server
- Servers are marked *dormant* given some threshold of unanswered requests
- Dormant servers are not included in broadcasts or trade requests

Attacks on Free Haven

- Attacks on **documents** or the **servnet**
  - physical attack, legal action, social pressure, denial of service, data flooding, etc.
- Attacks on the **reputation system**
  - simple betrayal, buddy coopting, false referrals, trading receipt games, entrapment, etc.
- Attacks on **anonymity**
  - reader anonymity, server anonymity, publisher anonymity, etc.

Comparisons

- Napster
  - no anonymity
- Gnutellar
  - efficiency, flexibility
  - no anonymity(querying?), no persistence
- Mojo Nation
  - efficient
  - centralized digital cash system
  - document anonymity, no guarantee of persistence

Conclusion

- Addresses all the requirements at one time:
  - decentralized storage service
  - anonymity of publishers, readers, and servers.
  - dynamic network
  - publisher-specified lifetime.
- Weakness:
  - sacrificing efficiency and convenience

Future work

- efficiency
- formal definition of anonymity
- accountability and reputation
- modeling and metrics
- deployed free low-latency pseudonymous channel
- usability requirements and Interface

References

- www.freehaven.net
- Roger Dingledine, Michael J. Freedman, David Molnar,
  - *Free Haven, In Peer-to-Peer, an O'Reilly book*
  - *the Free Haven Project: Distributed Anonymous Storage Service*
  - *A Reputation System to Increase MIX-net Reliability*