Peer-to-Peer Systems and Security
Introduction

Christian Grothoff

Technische Universität München

April 14, 2013

"They seem to have forgotten that, and are back saying the only purpose of P2P networks is for illegal trading of owned goods. We claim part of the reason for P2P is for legal trading of what ought to be in public domain. And what is in public domain in many cases." – John Perry Barlow

Peer-to-Peer Systems

Definition:

- A Peer-to-Peer (P2P) system is a system where participants work together as equals, with symmetric roles, rights and responsibilities.

- A *pure* P2P system is a P2P system where *all* (important) services are realized by peers.

This course is about P2P systems that use the Internet for communication between peers (also known as *overlay* networks).
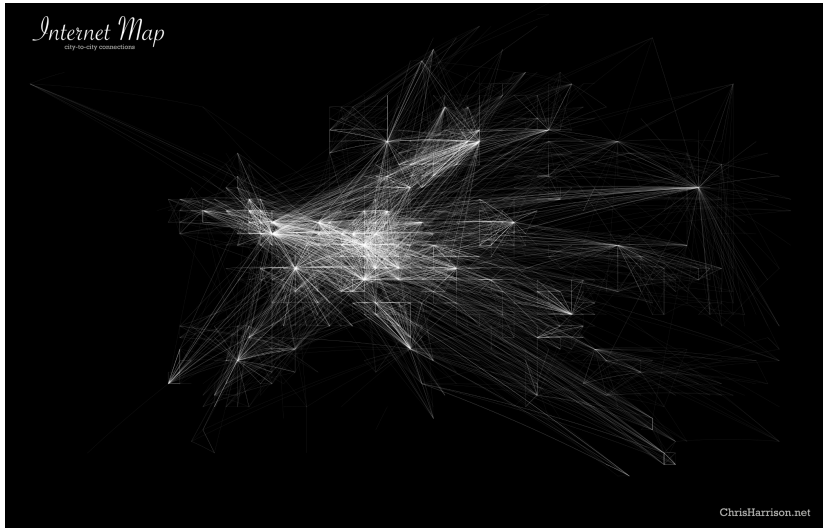
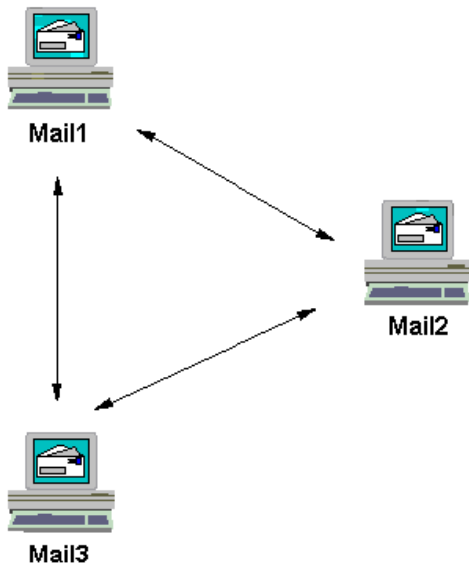# Famous P2P Systems: Democracy

Famous P2P Systems: Ethernet
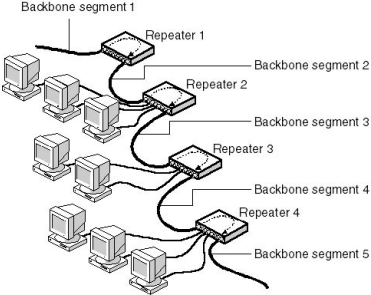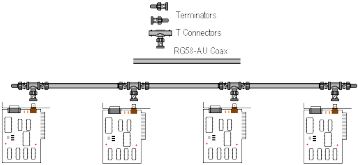
Famous P2P Systems: Internet (IP/BGP)

Famous P2P Systems: SMTP

Famous P2P Systems: World Wide Web

Famous P2P Overlay Systems

- Napster
- Gnutella
- Freenet
- Bittorrent
- Tor

Client-Server Benefits

Client-server architectures make it easy to:

- establish trust, and restrict access
- manage resources, and charge fees
- deploy updates, and remove features
- collect data, and sell it

Why not use Client-Server architecture?

If you use a server, you give up control of your:

- data
- computation
- free software

Why study *overlay* P2P Systems?

- Easier to develop and deploy
- Layered architecture: make use of existing abstractions
- Envision the future of the Internet!

Distributed Systems

An overlay P2P network is a distributed system. Deutsch
formulated "The Eight Fallacies of Distributed Computing":

- The network is reliable
- Latency is zero
- Bandwidth is infinite
- The network is secure
- Topology does not change
- There is one administrator
- Transport cost is zero
- The network is homogeneous

Questions?

?

Learning Goals

In this course, you will learn about:

- ▶ Protocol design
- ▶ Distributed algorithms & data structures
- ▶ System programming
- ▶ Game theory / Reputation Systems
- ▶ Network security & privacy

Learning Methods

- Lectures on existing designs and implementations
- Study current research papers
- Present (and discuss) your own ideas
- Implement your own protocol / extension

Deliverables

- Quizzes
- Written reports (design document, progress report, final report)
- Individual presentation on group project
- Project code
- Final individual interview
- **NO** final exam

Details at
http://grothoff.org/christian/teaching/2013/2194/.

The Project

- Webiste gives *suggestions*
- Teams of one or two students
- One project-related presentation per student
- Joint project reports
- Individual interview

Using GNUnet for the project is a suggestion, not a requirement.

Schedule

- Introduction & GNUnet architecture
- Security & unstructured protocols
- Structured Routing Algorithms & NAT traversal
- Game theory & Anonymity
- Attacks & Evil P2P networks
- Visions for the future

Schedule

- Introduction & GNUnet architecture
- Security & unstructured protocols
- Structured Routing Algorithms & NAT traversal
- Game theory & Anonymity
- Attacks & Evil P2P networks
- Visions for the future

- Presentations

Schedule

- Introduction & GNUnet architecture
- Security & unstructured protocols
- Structured Routing Algorithms & NAT traversal
- Game theory & Anonymity
- Attacks & Evil P2P networks
- Visions for the future

- Presentations

- Tor Hacker Meeting (July 22-26)

Project Ideas

- Tor-like OR in GNUnet
- In-network monitoring to detect attacks
- Distributed search engine [1]
- Cubit DHT [4] (or other "interesting" DHT [3])
- P2P over DNS, SMTP [5], SCTP, Satellite, ...
- Improved NAT traversal [2]
- M2M applications
- Asynchronous messaging
- Distributed constraint optimization [6]
- ...

Remember

- Study assigned reading before each class
- Review previous lectures before each class
- Form teams, e-mail team information to Andreas Korsten
- Prepare design documents, first presentation due in 6 weeks!

Questions?

?

References

Yacy.
http://yacy.net/, 2013.

A. Müller and A. Klenk and G. Carle.
Behavior and Classification of NAT devices and implications for NAT-Traversal.
IEEE Special issue on Middleboxes, pages 14–19, September 2008.

Ioannis Aekaterinidis and Peter Triantafillou.
PastryStrings: a comprehensive content-based publish/subscribe DHT network.
In *Proc. 26th IEEE Int. Conf. on Distributed Computing Systems (ICDCS '06), Lisboa, Portugal*, page 23, 2006.

Aleksandrs Slivkins Bernard Wong and Emin Gn Sirer.
Approximate matching for Peer-to-Peer overlays with Cubit.
Technical report, Cornell University, Computing and Information Science, 2008.

Ronaldo A. Ferreira, Christian Grothoff, and Paul Ruth.
A Transport Layer Abstraction for Peer-to-Peer Networks.
In *Proceedings of the 3rd International Symposium on Cluster Computing and the Grid (GRID 2003)*, pages 398–403. IEEE Computer Society, 2003.

Pragnesh Jay Modi, Wei-Min Shen, Milind Tambe, and Makoto Yokoo.
Adopt: asynchronous distributed constraint optimization with quality guarantees.
Artificial Intelligence, 161(1–2):149–180, January 2005.