Peer-to-Peer Networks

Chapter 2: Initial (real world) systems
(part # 3)

Thorsten Strufe

Chapter Outline

- Overview of (previously) deployed P2P systems in 3 areas
- P2P file sharing and content distribution:
  - BitTorrent, Napster, Gnutella, KaZaA
  - Differences, strengths, weaknesses
- P2P Communication
  - Typical instant messaging setup
  - Skype
- P2P Computation
  - SETI@Home example

KaZaA: Spyware

- KaZaA included spyware in their program
- Spyware does things like:
  - Sends all DNS queries to a tracking server
  - Monitors visited websites
  - Additional popup windows on “partner” sites
- KaZaA originally denied existence of spyware
- In theory, possible to disable spying functions
  - Removal software reportedly failed often…
  - Spyware-free versions available for download (sometimes for sale)
  - “Spyware-free KaZaA” (malware) for download…

KaZaA: Strengths

- Main strength of KaZaA: Combines good points from Napster and Gnutella
  - Efficient searching under each supernode
  - Flooding restricted to supernodes only
  - Result: Efficient searching with “low” resource usage

- Most popular network (globally)
  - Lots of content, lots of users
  - Some networks more popular in some areas (e.g., eDonkey in Germany, now aMule all over Europe)
  - Currently most big file sharing networks have been shut down

KaZaA: Weaknesses

- Search not comprehensive
  - Can still miss a file even though it exists
  - But better reach than Gnutella
- Easy to create collisions for UUHash (feed-in corrupted files)
- Single point of failure?
  - Lawsuits against KaZaA eventually successful
  - Software comes with list of “well-known” supernodes
    - Increases robustness?
    - More targets for lawyers?
- In general, solves many problems of Napster and Gnutella

Hierarchical Systems, Annex

- Many other systems followed FastTrack
  - Gnutella 0.6 extension of Gnutella to supernodes
  - „Gnutella2“ (publicity stunt) „Leaves“ „hubs“ (iterative queries at SN) „walks“
  - eDonkey network (eMule, aMule, iMule, jMule, lMule, etc.)
    - hierarchical: users can run server or client
    - eDonkey2000 Closed source (caused by shame, because it was coded so badly…)
    - Free/Open Source Software (FOSS) (aMule/eMule) included DHT later on

Napster vs. Gnutella vs. KaZaA

	Napster	**Gnutella**	**KaZaA**
Type of Network	Centralized	Distributed	Hybrid
Efficient Searching	+++	---	+
Resilience to Attacks	---	++?	+
Open Protocol	N	Y	N
Spyware-free	Y	Y	N?
Popularity	+++	-	+++

For Your Long Term Memory

- Problems that file sharing systems solve well
  - Connectivity (find neighbors, stay connected)
  - Name- and location services
  - Request delegation/routing
- Name space (index) implemented
  - Centralized index (full namespace, complete index)
  - Distributed index (full namespace, partial index)
  - “Hybrid” index (full namespace, partial but aggregated index)
- Delegation/“Routing” depending on the impl. of namespace

Putting the Systems into Perspective

- Entirely different approaches exist
  - Information dissemination: Gossiping / Epidemic routing
  - Stochastic: Random Walks / Percolation Search

- Questions remain
  - Hit rate: search not comprehensive (routing not deterministic unless Napster)
  - Neighbor selection?
  - Load balancing?
    - Requests / storage of namespace?
    - Forwarded messages?
    - Stored resources (“shared” files)?
  - Effectiveness (load on the underlying network)?
  - Fairness?
  - Resilience to malicious behavior?

Next Generation P2P...?

- P2P networks seen so far are called unstructured
  - Content can be placed anywhere in the network
  - Centralized/fully distributed sometimes called 1st generation P2P
  - Hybrid, Gossiping, Random Walk st. called 2nd generation P2P
- Contrast: Structured networks
  - Every file has a well-defined place (distribution of the namespace)
  - Sometimes called 3rd generation P2P (will there be a 4th?)
  - See DHTs in Chapter 3

File Sharing: Current State

- Most bigger file sharing networks sued into submission
  - Napster, Kazaa, eDonkey, (the pirate bay)…
- Many networks still up and running
  - Because many open clients are available
  - New target are the users (hadopi/three strikes, ask Piratenpartei…)

- Future is uncertain
- Content owners (record companies and movie studios) are moving into online delivery of content
  - iTunes and others for music
  - iTunes, Amazon for movies and TV content
  - "If it was for a fair price…" – but what is a fair price?
  - (*Plus:* we are hunters and gatherers)

- Many different kinds of file sharing networks
- Old ones go, new ones come (pace slowing down?)
- Remains to be seen… Stay tuned!

P2P File Sharing: Summary

- File sharing networks extremely popular
  - Different networks come and go
- File sharing based on keyword searches
  - Keyword matches either file name or metadata
  - Must use same keywords (or pattern matched) as provider
    - Usually not a problem
- No guarantees about file being what it claims to be
  - Record companies inject files with dummy content
  - Solution: Each file has hash, make public list of “bad files”
- Future looks uncertain

P2P Communication

- P2P communication is a communication architecture based on the P2P principle
- Examples: Email, network news, instant messaging, telephony

- Current email and news systems are P2P to some degree
  - See below for details

- Generally possible to implement any communication using P2P
  - Remove central management
  - Remove any dedicated servers

P2P Communication Example: IM

- Typical instant messaging system is P2P
  - Centralized server has buddy lists
  - User logs on to server, sees buddies on-line

- Chatting directly between peers
  - Including audio, video, and file transfers

- Role of centralized server: (similar to Napster)
  - Bring people together
  - Centralized server also helps with firewalled clients

- Jabber, P2P IM
  - jabber/xmpp connects distributed servers (encrypted, if wanted)
  - Servers interconnect, making use of DNS for location/routing!

P2P Communications: Email and News

- Current email and news systems have P2P components
- In Email, Mail Transfer Agents (MTA, mail servers) exchange email directly between them
  - No central coordination, except through DNS
  - Automatic transfer of messages, according to DNS MX records

- ***By the way: which problems do we experience here? :-)***

- In News, NNTP servers exchange articles between them to build news feed
  - Again, no central coordination except DNS
  - Feeds typically set up through agreements between admins (Gossiping)

From user's point of view, P2P is hidden

- User always has to access the same mail server to get her mail
- Same for news (although technically this could be avoided…)

P2P Communications: Skype

- Skype is a popular Internet telephony software
- Allows the user to
  - Make calls to other computers on Internet
  - Make calls to real phone network (costs money)
  - Have calls made to a real phone number forwarded to Skype (also costs money)
- Skype developed by same people as KaZaA
  - But: Skype is perfectly legal (the affected industry is “only” telcos, they sell DSL...)
- Architecture of Skype very similar to that of KaZaA
  - Supernodes and ordinary nodes
- Very popular, ~300 million downloads, ~15 million concurrent users online
- Clients for Windows, Mac, Linux, Smartphones...

Skype: Details

- Skype is a proprietary and encrypted protocol
- No real details available :-(
- Best study about Skype: “it sends 48 bytes over TCP to some IP address, then 512 bytes to this address”…
- What is known from Skype:
  - One central server for login and billing
  - Everything is encrypted, key-pair allocated by centralized server
  - Supernodes behave much the same way as in KaZaA
    - Normal nodes connect to SN, SN are not firewalled, etc..
  - Directory of who is online is spread over the peers
    - Details unknown, Skype claims that system knows all users who were online in the last 72 hours
    - Keeping state of 20m nodes (160 bit ID?) → 381MB of IDs
  - Skype goes through firewalls (it does, try! :)
    - As long as firewall allows (some) outgoing connections

(*) Baset et al.: An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol, INFOCOM.

Skype: Supernodes and Calls

- Supernodes (and some other nodes?) have more responsibilities in Skype than in KaZaA
- Supernodes are responsible for forwarding actual data traffic (calls) between (firewalled) peers
- No (easy) way to disable this in client software
  - Configure your own firewall, restrict Skype’s connectivity, …

Why is skype closed-source? Keys assigned, not much known?

- Advantage of Skype is high call quality
  - Better than POTS in many cases (*don’t blame skype for your WLAN)*!
- Skype has highly efficient voice codec
  - About 5 kB/s of traffic generated (even during silence)

P2P Computation

- P2P computation: share computation between many peers
- Computationally intensive problem to solve
  - For example, crack encryption or find messages from aliens
  - Problem: needs to be easy to parallelize and distributed to peers
- Typically, centralized server manages work distribution and aggregation
  - Distributes work to peers
  - Peers only perform their computation, send back result
  - Each peer contributes at its own speed
  - Results verified somehow (problem dependent)
- Usually no special reward for participation
  - Common goal for all peers
- Uncontrolled and un-administered
  - Peers free to join and leave when they wish, contribute what they want

Why does P2P Computation Work?

- P2P computation works because common goal appeals to people running peers
  - Read: People do it because they think it is worthwhile
- People participating are “techno-nerds”
  - Cracking encryption and SETI@Home are “cool”
  - Common, non-profit purpose
  - Often run on campuses and dorms (= lot of “free” computers)
- What if run by a private company for proprietary purposes?
  - For example, a car company wants to model a wind tunnel
  - Or military wants to simulate a nuclear detonation
  - (Usually hidden behind other, “philantropic” tasks)
- *Is it possible to build a P2P computation system where users are paid for their contributions?*

P2P Computation: Example

- Several P2P computation projects active
- SETI@Home and distributed.net were the first
- SETI@Home project run from UC Berkeley
  - Now a project in BOINC (Berkeley Open Infrastructure for Network Computing)
- Search for Extraterrestrial Intelligence (SETI):
  - Goal of SETI project is to discover signals from extraterrestrial civilizations
  - SETI@Home uses P2P computation to identify those signals
- Why is P2P (distributed) computation needed in SETI?
  - Signal parameters are unknown, sensitivity of search depends on available computation power
  - Need to scan large frequency bands, correct for Doppler shift, filter out local interference (from Earth)

SETI@Home

- SETI scans 2.5 MHz wide band around 1,420 MHz
  - Assumed to be universally of interest (hydrogen line)
- Total amount of data from survey expected to be around about 39 TB of data
- Data divided into work units at UC Berkeley
  - Work units sent to clients (picked up by clients, actually)
  - Client can work offline, takes several hours per work unit
- Clients reply with results from computation
  - Each work unit calculated by several clients
    - Undetected errors occur once every 10^{18} machine instructions (SETI would see several such errors per day!)
    - Communication errors
    - Would people cheat for the Kudos? ;-)
- Communications over HTTP
  - Consider clients behind a firewall

SETI@Home: Some Old Numbers

- Most importantly: No alien signals detected yet ☹ (or ☺?) (*)
- Client available for 47 OS/hardware combinations
- Millions of users, many with multiple machines
  - Users organized in teams
  - Teams “compete” against each other
  - SETI relies on volunteers, no rewards offered
    - Except prestige from being in “leading team”
    - And the distant possibility of finding a signal…
  - Grain of salt: 330k “stable” users for seti@home, 200k for folding@home
- Total throughput: 10.3 petaFLOPS (2009, 8.8: folding@home)
  - “RoadRunner” aimed: 1.7petaFlops (1.4), 133Million US$ (2009)
- And still: new signals added faster than they are processed

(*) http://www.telegraph.co.uk/science/space/8875780/White-House-denies-alien-cover-up.html

Chapter Summary

- P2P systems in active use in many areas
  - Main focus in content distribution (file sharing networks)
- Show well properties of P2P principle
  - Autonomous
  - Exploit edge resources
  - Intermittent connectivity
- Different types of system (content distribution, communication,computation)
  - Several different file sharing networks, each with good and bad points
  - Several communication networks
  - Many computation projects
- No single solution, approach or system ruling over others

Additional Readings

BitTorrent

[1] Piatek et al.: Do incentives build robustness in BitTorrent?, NSDI'07

KaZaA

[2] Chawathe et al.: Making Gnutella-like P2P Systems Scalable, SIGCOMM (2003)

[3] Liang et al.: Understanding KaZaA

[4] Leibowitz et al.: Deconstructing the Kazaa Network, WIAPP '03

Napster & Gnutella

[5] Ripeanu .: Peer-to-peer architecture case study: Gnutella network, P2P computing 2001

[6] Acosta et al.: Understanding the Practical Limits of the Gnutella P2P System: An Analysis of Query Terms and Object Name Distributions ACM/SPIE Multimedia 2008

[7] Sarolu et al.: Measuring and analyzing the characteristics of Napster and Gnutella hosts, Multimedia systems 2003.

Skype

[8] Baset et al.: An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol, INFOCOM 2006

[9] Ford et al.: Peer-to-Peer Communication Across Network Address Translators, USENIX 2005.

Power Law Networks

[10] Albert et al.: Error and attack tolerance of complex networks, nature 406.

[11] Brinkmeier et al.: Methods for Improving Resilience in Communication Networks and P2P Overlays