Anonymity Online

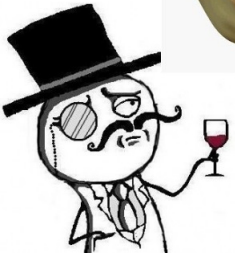
194.144 Privacy-Enhancing Technologies

Dr. Markus Donko-Huber

Outline

- Online anonymity
- High-latency anonymity systems
  - Remailer
- Low-latency anonymity systems
  - Onion routing
  - DC nets
  - Broadcast traffic
- Private file-sharing

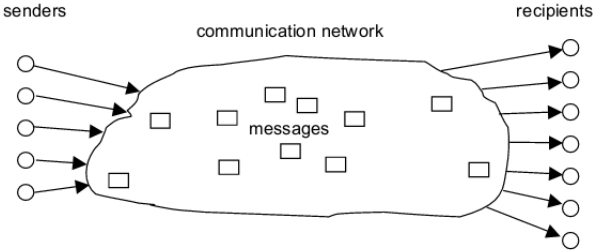
Anonymity online I



Anonymity online II

Anonymity Systems

- anonymity
  - From ancient Greek “without a name” or “namelessness”
  - Meaning: *not being identifiable*
- anonymity systems: unlinkable communication

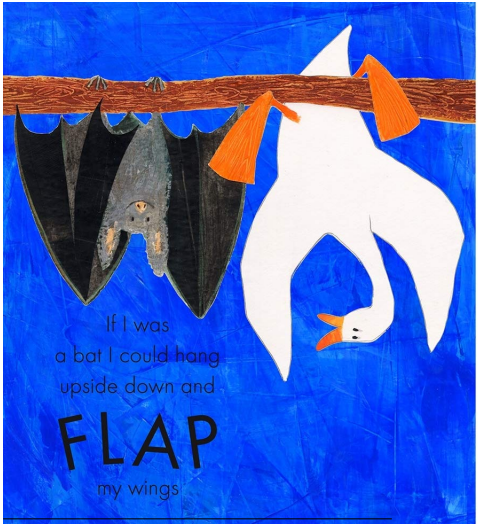


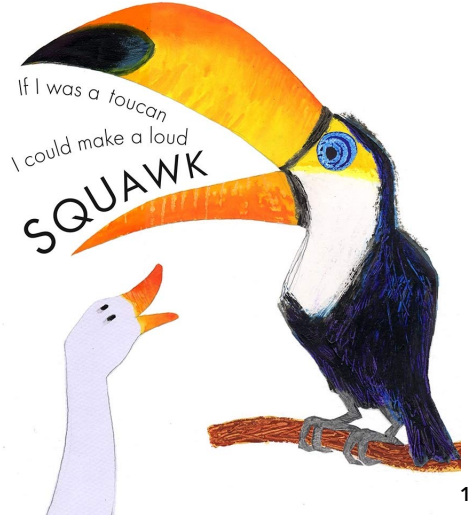
Anonymity Set

"Anonymity is the state of being not identifiable within a set of subjects, the anonymity set."
[*Pfitzmann, 2000*]

- Anonymity requires a **peer group/set**
- Well-defined group of individuals
- The bigger the better
  - Example: general suspicion if you use Tor
    http://sz.de/1.2029100

Anonymity Set: Suzy Goose

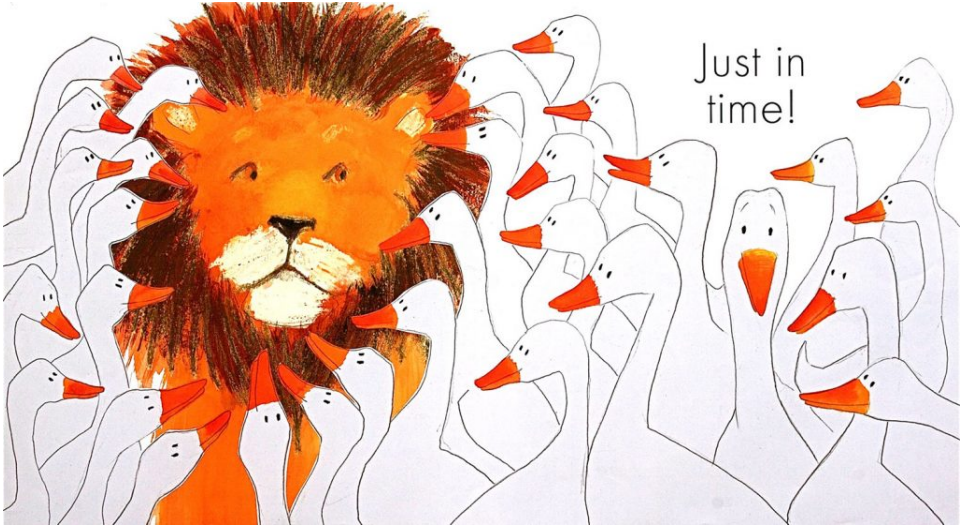




[1]

[1](C) Petr Horácek

Anonymity Set: Suzy Goose



basic terminology

- Pseudo-anonymity
- Unlinkability
- Sender anonymity
  - Sender of a message cannot be determined
- Recipient anonymity
  - Recipient of a message cannot be determined
- Relationship anonymity
  - Messages cannot be attributed to a pair of users
- Unobservability
  - Cannot be determined if specific user sent messages at all [2]

[2] see "anonymous" threat at Harvard

Pseudo-anonymity

- Fake names = pseudonyms
- Examples
  - nicknames, alias (showbiz name)
  - student number
  - email addresses, telephone numbers
  - usernames, user IDs
  - bank account / credit card numbers

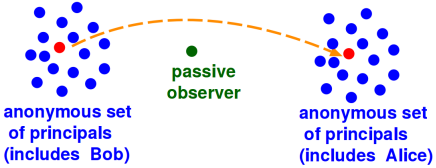
Unlinkability

- Can (pseudonymous) information be linked for de-anonymization?
- Example: Wikipedia entries
  - Anonymous entries at Wikipedia (no user account / pseudonyms)
  - IP addresses of entries $\rightarrow$ organizations
  - See https://en.wikipedia.org/wiki/WikiScanner
  - Possible solution: use proxies or Tor
- Example: Anon Ops press release
  - File released on one-click hoster, IP address unknown
  - metadata of file: name of its author
  - See Spokesman leaves name in PR meta data

Demo: Linkability

Attacks on anonymity systems

- Active attacks
  - Compromise of software (security/implementation bugs)
  - Malicious users (sibyl attacks) : denial of service, modification of messages
  - Infiltration (also for passive attacks)
- Passive attacks
  - Global observer
  - Correlation over timing/content (also when encryption is used)



Internet anonymity

- Complex challenge
  - Multitude of attack vectors
  - Hard to be done correctly
- Needed by reporters, whistleblowers, average user ...
- Two broad classes of online anonymity systems
  - High-latency systems
  - Low-latency systems

High-latency systems

High-latency systems

- For message-based communication
  - E-Mail, Usenet postings, ...
- Anonymous e-mail messages
- Slower message transmission is acceptable (asynchronous)
  - Not possible for e.g. web browsing
  - Messages might be delayed for hours/days

Penet Remailer **anon.penet.fi**

- Founded 1993 by Johan Helsingius
- Pseudoanonymous remailer for email and usenet
- How did it work?
  1. User sends email to remailer
  2. Remailer removes sensible information
     (sender, ip address of used SMTP Servers etc.)
  3. Remailer forwards E-Mail with different sender address
     e.g. `pseudo@anon.penet.fi`
  4. Remailer stores link for answers:
     e.g. `leet@tuwien.ac.at` $\rightarrow$ `pseudo@anon.penet.fi`

Penet: Singe Point of Failure

- Database with links from real addresses to pseudonyms
- Compromise of anon.penet.fi server $\rightarrow$ compromise of (pseudo)anonymity
  - First rumors at DEFCON III (1994)
- 1995 – Scientology
  - Internal documents released at `alt.religion.scientology`
  - User '-AB-'
  - Scientology contacts FBI and Interpol
  - Finnish police enforces release of user information
  - Full story: "What Really Happened in INCOMM" (2003)
- Project terminated 1996

alt.anonymous.messages (amm)

- Usenet group from 1994
- All users post to the same mailbox
- Use of asymmetric cryptography
  - Alice gets all messages of the usenet group but can only read messages from Bob
- DEFCON 2013
  - De-anonymization of *amm* users (PGP Key-ID, outdated crypto)
  - https://ritter.vg/blog-deanonymizing_amm.html

Chaum Mix

Chaum Mix

- Base of modern anonymity systems
- David Chaum: ”***Untraceable electronic mail, return adresses, and digital pseudonyms***”, Communications of the ACM, 1981
- Chaum Mix
  - Order of sent messages != order of received messages
  - Messages are split into equal chunks and padded
  - Hinders de-anonymization based on analyzing the network traffic

One-hop mix

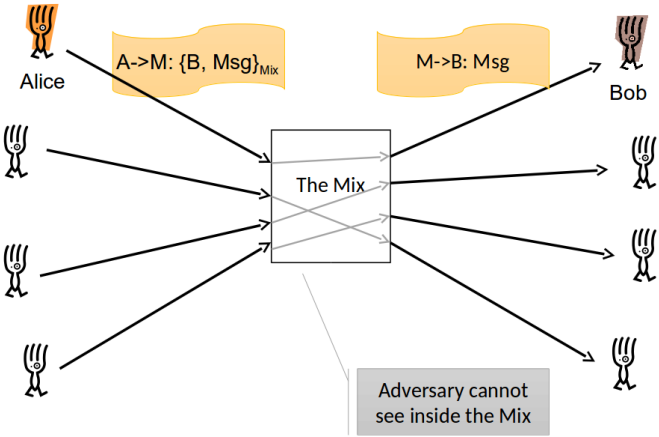
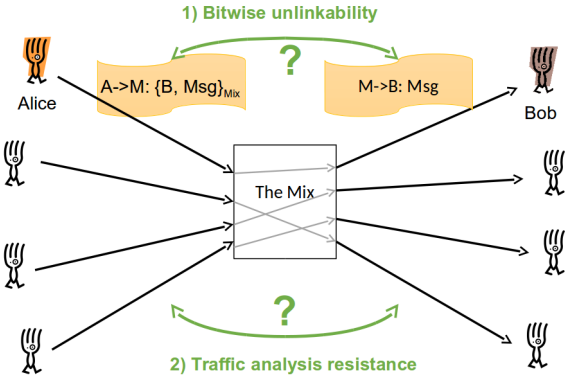


Figure: Basic Mix, George Danezis, UCL

Chaum mix properties



- unlinkability: use of cryptography, message chunking
- traffic analysis: reordering of messages

Broken Mix: no reordering but FIFO

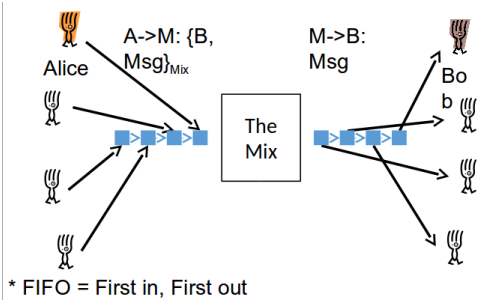


Figure: FIFO Mix, George Danezis, UCL

Attackers can link senders to recipients by observing the mix.

Protection against traffic analysis

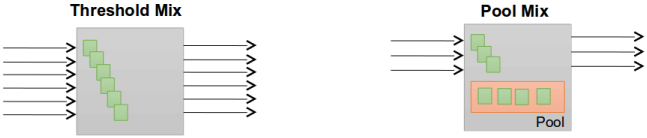


Figure: Mix Batching / Pooling, George Danezis, UCL

Threshold sends all messages once certain number of messages reached (Chaum), pooling: some messages are kept back.

Statistical disclosure attacks on threshold mixes

For each round where Alice sends a message, mark down recipients. After `n` round, which users received the most messages?

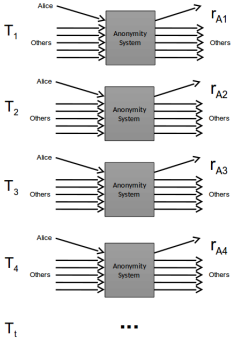
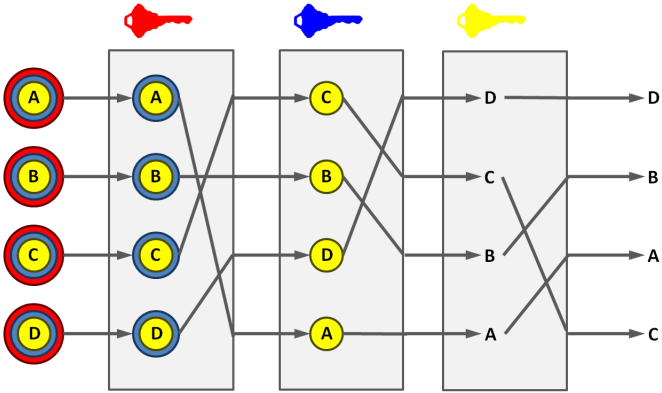


Figure: Mix Batching / Pooling, George Danezis, UCL

Chaum Mix Network

- Chaum Mix Network (Mix networks)
  - Chaining of multiple Chaum mixes (Russian dolls)
- Made possible with Public Key Cryptography (1976)
  - Diffie, Hellman: ”New Directions in Cryptography”



Remailer

Remailer based on Chaum mixes

- Cypherpunk anonymous remailer, beginning of ’90s (Remailer type I)
  - Messages are sent encrypted to the remailer
  - Remailer decrypts messages and forwards it to the recipient
  - Cascading remailers is possible
  - No answers possible
- Mixmaster, mid ’90s (Remailer type II),
  - Last version released 2008
  - Each message is split into equally sized chunks and messages are reordered
  - Basically an implementation of a Chaum Mix

Remailer based on Chaum mixes II

- Mixminion, 2003 – 2007 (Remailer type III)
  - Reordering of messages (pooling)
  - Batch-based sending (batching)
  - Equally sized messages
  - All communication between mixes is encrypted (TLS)
  - Answers possible with “single-use reply blocks“
  - Protection against denial of service attacks
- Open problems
  - Traffic analysis with denial of service attacks (attacker overloads mix with dummy traffic)
  - No active development / further projects, spam

~~Mixmaster 3.0 Demo~~

mixmaster et al. seem defunct :(

Youtube: Mixmaster Demo by Steven Murdoch

Low-Latency systems

Low-Latency systems

- Delays for a couple of hours are unacceptable for many of today's Internet services
  - Examples: HTTP, SSH, instant(!) messaging
- Low Latency systems
  - Enable interactivity
  - Goal: anonymity with as little delay as possible
- Challenges
  - Factor time cannot be used to hinder traffic analysis
  - Multitude of different applications
  - near real-time applications such as VOIP

Simple low latency systems

- Protocol-dependent services
  - Open HTTP proxies for web surfing
  - Chaining of proxies possible
  - Websites that act as proxies (e.g. `http://anonymouse.org`)
  - Bouncer (BNC) for FTP, IRC
  - Discard email addresses
    - `http://www.spambog.com/`
    - `http://www.mintemail.com`
    - `http://www.filzmail.com`
- Problem: Single Point of failure (cmp. penet remailer)

Proxies and VPNs

Problems with proxies

- Users can be de-anonymized via JavaScript / Java / Flash
- Anonymity-software (e.g. Tor browser bundle) therefor uses modified web browsers
- Example online test to check for leaking information:
  - http://ip-check.info/

Example Web Proxy: Ultrasurf

- Initially developed to provide an anti-censorship service for Chinese web users
- ”Ultrasurf: Privacy, Security, Freedom”
- Client-software for MS Windows
- Connection via TLS to ultrasurf proxies
- Not a privacy tool!
  - Ultrasurf logs, certain websites blocked, Flash/JavaScript
  - https://media.torproject.org/misc/2012-04-16-ultrasurf-analysis.pdf

VPN Services

- VPN = protocol-independent service
- How to decide on a provider
  - Logging policies
    - http://goo.gl/DQ1uo5
  - Supported encryption protocols
    - PPTP: simple to crack (e.g. Moxie's CloudCracker)
    - IPSec completely broken by NSA?
  - https://thatoneprivacysite.net/
- VPN via own VPS
- Issues
  - Provider trust: Linkable via payment provider, IP addresses
  - Leaks: IPv6, DNS, WebRTC

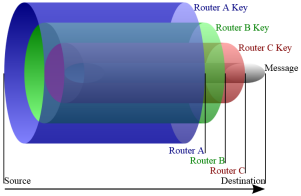
VPN Services: recent incident(s)

- “UFO VPN” incident July / 2020
  - “Zero logs” VPN service based in Hongkong
  - 894 GB on unsecured Elasticsearch cluster
  - Passwords (in plain), IP addresses, connection timestamps.
  - for details see: blog post
- Brief summary of issues with (free) VPN provider
  - Free VPNs are bad for your privacy

Mix Networks

Onion-Routing

- Based on the research of Chaum
- Cascade of onion routers (mix network)
- Most popular use of onion routing: Tor
  - Covered in detail in separate lecture
- Onion Routing != Tor
  - Tor: global network view
  - Hidden services
  - Out-proxies



Example Mix Network: Java Anon Proxy (JAP) aka JonDonym

- Based on research project of TU Dresden, Universität Regensburg
- Java implementation for all common operating systems
- Static set of mix nodes, new mixes must get certified
- Difference to Tor: not everybody can operate a mix
- Free version and payment model
  - https://www.anonym-surfen.de/
  - Free version limits bandwidth

JonDo Demo

Law enforcement and JonDonym

- 2003 the German Bundeskriminalamt enforced the collection of network information by JonDonym
- Logs for a given set of websites
- 2006 a JonDonym server had been seized
- Overview of requests by law enforcement
  - https://www.anonym-surfen.de/strafverfolgung.html
- ~~Data retention law (2006/24/EG)~~
  - ~~Adapted by a number of JonDonym mixes~~
  - Timestamps and connection logs
  - De-anonymization requires data from all three mixes of a path

Garlic Routing

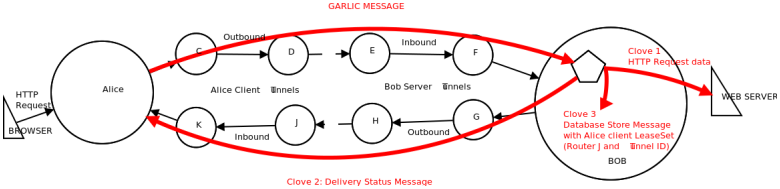
- Foundation of the Invisible Internet Project (I2P)
- Layer-based encryption
  - Basic idea: Chaum mixes / Onion Routing
- Messages are bundled
  - Messages are merged into Bulbes/Cloves
- ElGamal/AES + SessionTag
  - Combination of asymmetric and symmetric encryption methods

Example Garlic Routing: I2P

- Based on Java, active development since 2003
- Available for common desktop OSs and Android
- I2P Router creates local proxy (4444/TCP)
- I2P Applications
  - Filesharing (BitTorrent, eMule, Gnutella)
  - E-Mail (Postman, I2P-Bote)
  - Instant messaging (I2P Messenger)
  - Publishing (Syndie)
  - Distributed file-system/storage (Tahoe-LAFS)

Garlic Routing: I2P

- Tunnels to other I2P nodes are created (incoming / outgoing tunnels)
- Use focuses on “Darknet“ applications (as opposed to Tor)



I2P Demo

Problems with Onion/Garlic Routing

- Global Observer
  - Entities that can monitor the ”entire“ Internet (cmp. Snowden Leaks)
- Analysis of each mix node’s traffic
- Correlation of user traffic entering and leaving the mix network
  - Which user has seen which website at which time?
- Correlation is practical with little resources (e.g. Tor: 100MBit + six month time, see http://goo.gl/267e9z)

Experimental Anonymity Systems

Anonymity-systems with protection against global observers

- Simple ”Broadcast Ring”
  - Every user creates a key-pair and public key if accessible to everyone (public key must not contain personal information)
  - Users publish encrypted message or random message in fixed time intervals
  - **Large network overhead**
- Dining Cryptographers
  - Information-theoretical safe, ”simple system” based on e.g. RSA
- Broadcast / Cover Traffic systems
  - Similar to ”Broadcast Ring” with optimizations for network overhead

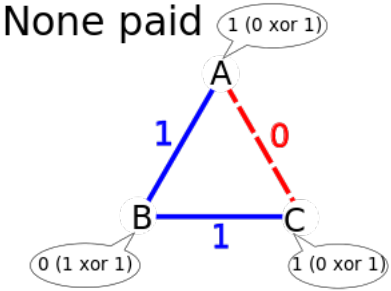
Dining Cryptographers Problem

- A group of cryptographers goes out for dinner, once they want to pay, the waiter tells them that their food has already been paid for.
- How can they determine if one of the cryptographers or the NSA paid, without knowing who paid?
  (Dining Cryptographers, Chaum 1988)

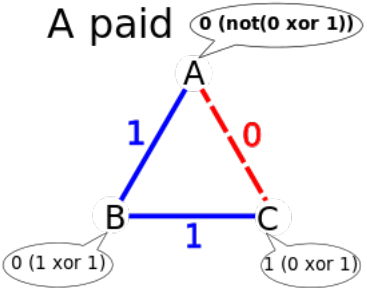
Dining Cryptographers (DC) algorithm

- Every pair of cryptographers shares a secret
  - Coin-flip behind menu
- Every cryptographer
  - XOR over all shared secrets
  - If he/she did not pay: result of XOR
  - If he/she paid: negates result of XOR
- Final result
  - XOR over all cryptographer's results
  - 1 : Somebody at the table paid
  - 0 : Nobody at the table paid

Example: Dining Cryptographers



1 xor 1 xor 0 = 0



1 xor 0 xor 0 = 1

Advantages of DC-Nets

- The simple DC-net algorithm can be used to transmit any type of information
  - e.g. in most ineffective way per Bit
- DC-nets provide strong anonymity
  - Analysis of network traffic does not reveal anything
  - Senders and recipients can not be linked

Drawbacks of the original DC-net algorithm

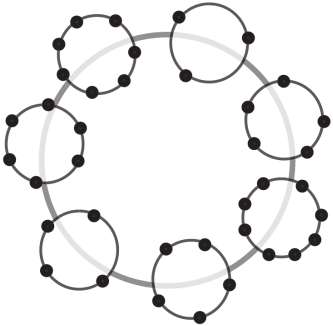
- Collisions
  - Invalid results if two cryptographers pay
- Disruptions
  - Malicious cryptographer could send random bits
- Complexity/Scalability
  - DC-net algorithm with multiple participants problematic
- Practical use
  - Secure channel for group-wise exchange of secrets
  - Long messages

Example DC Net: Herbivore

- Response to FBI program “Carnivore” (2000)
  - Carnivore (Fleischfresser), Herbivore (Pflanzenfresser)
  - Project of Cornell University
  - https://www.cs.cornell.edu/People/egs/herbivore
- Strong anonymity
  - Senders, recipients can not be detected (active/passive attacks)
- Scalability
- Robust against attacks

Herbivore Design

- Network is collection of smaller/local DC-Nets
- Algorithm for global topology
- Network is segmented into ”anonymity cliques”
- Slots to send messages
- Out-Proxy / Darknet



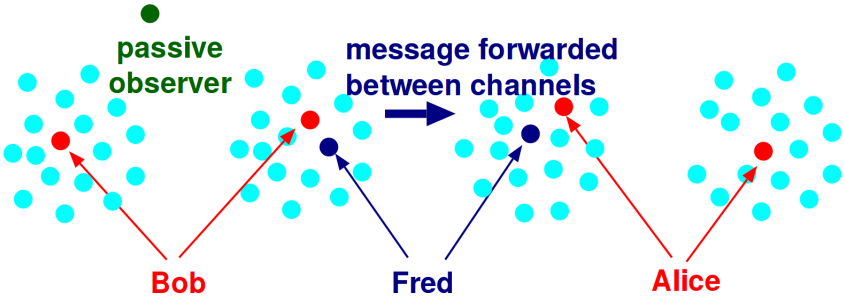
Example DC Net: Dissent

- Project at Yale University, published around 2010
- http://dedis.cs.yale.edu/dissent/
- Scales up to $\sim$1000 participants
- Detection of malicious participants
  (with cryptographic shuffles)
- Main use for broadcast applications
  (wikis, auctions, e-voting)

Broadcast/Cover Traffic systems: P^5

- P^5 (Peer-to-Peer Personal Privacy Protocol)
- http://www.cs.umd.edu/projects/p5/
- Optimization of the simple "broadcast ring"
- Users are split into broadcast groups
- Noise / Cover Traffic
  - If no message is send, random cover traffic is transmitted

P^5: Broadcast Groups



P^5 Broadcast Groups: if users are not in the same broadcast group, messages are forwarded

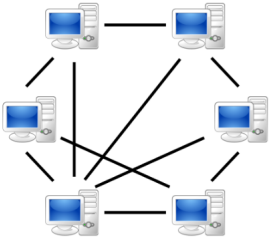
DC nets and broadcast systems

- Until today limited to research prototypes
  - No active software development
- DC-Nets scale up to a couple of 1,000 users
- Not in widespread use
- Network / performance overhead

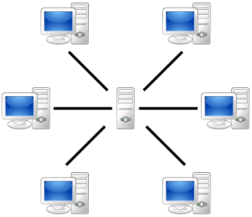
Private File-Sharing

Private P2P systems

- Decentralized applications for filesharing (Peer-to-Peer, P2P)
- Goals
  - Censorship resistance (motivated by Napster)
  - Anonymity
- Challenges
  - Distributed search index
  - Detection of other peers



P2P network [Wikimedia]



central network [Wikimedia]

Anonymous P2P systems

- Freenet
  - Ian Clark, University of Edinburgh, first version 2000
  - https://freenetproject.org/
- Goals of Freenet
  - Protection against Internet censorship
  - anonymity (sender, recipient)
  - availability, robustness, scalability
- Freenet architecture
  - Every freenet user contributes storage
  - Static websites (freesites) and files

Freenet: Architecture

- GUID
  - Files are identified via GUID (globally unique identifiers)
  - Content-hash keys: SHA1 of a certain file
  - Signed-subspace keys: personal namespace (world readable, writable only by creator) , e.g. surveillance/us/snowden-leaks
- anonymity
  - Chaum Mix nets
- Opennet mode
  - Automatic peering with other freenet users
  - Relatively simple to block
- Darknet mode
  - Friend-2-Friend networks
  - Higher anonymity and harder to block (no central components)

Freenet Demo

InterPlanetary File System (IPFS)

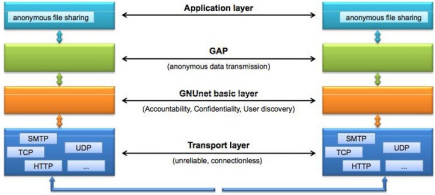
- IPFS[3]
  - distributed file system
  - content-addressing in global namespace
  - Web gateways for access without local node
  - Native support in Brave, Opera for Android
- IPFS vs. Freenet
  - IPFS is optimized for performance vs. anonymity
  - IPFS nodes only store content explicitly
  - IPFS as a PET: anti-censorship[4]

[3] https://ipfs.tech

[4] 2017 Catalan independence referendum

GNUnet

- Framework for decentralized P2P networks
  - https://gnunet.org/
- Different transport plugins
  - TCP, UDP, HTTP, HTTPS, WLAN, Bluetooth
- Encrypted communication links (RSA + AES 256)
  - Messages: 1kB blocks
- anonymity via “GAP”



[fidis.net]

Self-Hosting (web services)

Self-hosting web services

- Major online services controlled by few companies
- Self-hosting == “private cloud“
  - Nextcloud for Files, Calendar, Contacts
  - Tahoe-LAFS for encrypted, distributed storage
  - matrix decentralized communication
- Self-hosting challenges
  - (steep) learning curve
  - Bandwidth / VPS costs
  - Privacy will always depend on your peers

Anonymity systems overview

method	example	use-case	advantage	disadvantage
pseudonyms	penet.fi	E-Mail	sender pseudonymity	single point of failure
Remailer (Type I,II,III)	mixmaster	E-Mail	sender-anonymity	No active development, active attacks
proxies	zend2.com	Web	simple usage	Provider trust, plugin leaks ...
VPN	Mullvad, NordVPN	Web	simple usage, protection (public WiFi)	Trust in Provider, network leaks
onion-routing	Tor	Web	simple usage, large community	global observer, speed
garlic-routing	I2P	Darknet (Filesharing, E-Mail, etc.)	inbuilt applications	global observers, small community
DC nets	Herbivore, Dissent	broadcast applications	strong sender/recipient anonymity	No user base, academic prototypes
Decentral P2P	GNUnet, freenet	filesharing	censorship resistance, anonymity	performance, community

Conclusion

- High-Latency anonymity systems
  - For message-based communication
  - Chaum mix networks
  - protection against global observers
  - asynchronous communication (delays)
- Low-Latency anonymity systems
  - VPN, Proxies
  - Onion Routing
    - Tor, I2P $\rightarrow$ popular anonymity networks
  - DC-nets/ Broadcast systems
    - strong anonymity
    - Performance / usable implementations
  - Anonymous Filesharing systems
    - de-central, anonymous, censorship resistant