Anonymous Communications (I)

George Danezis

Security Group,
Computer Laboratory,
University of Cambridge.

George.Danezis@cl.cam.ac.uk

Objective

- This is not just academic
- Objective:
  - Empower people to **use**.
  - Help **implement** anonymous comms.
  - Help **design** and **analyze** systems.
- Present where we are, and where we are going with the Mixminion remailer.
- Onion routing derivatives – next talk!

Outline

- Introduction to anonymous comms.
- Basic principles.
  - Do not reinvent the wheel.
- Current research.
  - Do not reinvent the rocket either.
- History of remailers.
- What is to be done?

Introducing the problem

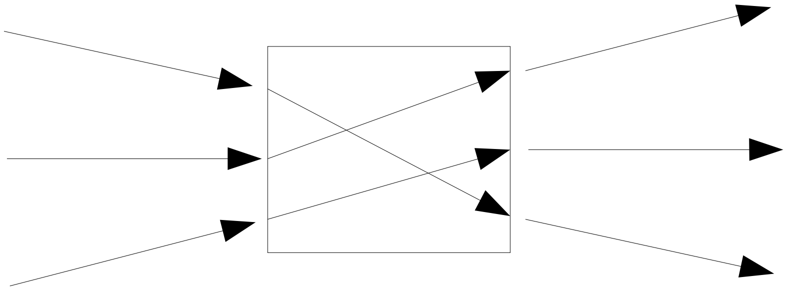
- Real world: whistle blowers, human rights work, elections, e-cash, political speech, …
- Anonymous communications: what is it?
  - Alice wants to talk to Bob without anyone, including Bob, knowing her identity (sender anonymity).
  - She wants Bob to reply without anyone knowing her identity (receiver anonymity).
  - The two can be combined to provide bi-directional anonymity.

Meet the adversary

- We assume:
  - **Eve** can observe all the network links.
  - **Mallory** can modify, delete, inject messages as they travel on any network links.
  - **Bob** is working with them, not Alice.
  - Some trusted **third parties** are corrupt, and misbehave.
- Stage of clinical paranoia makes designers sleep well at night.

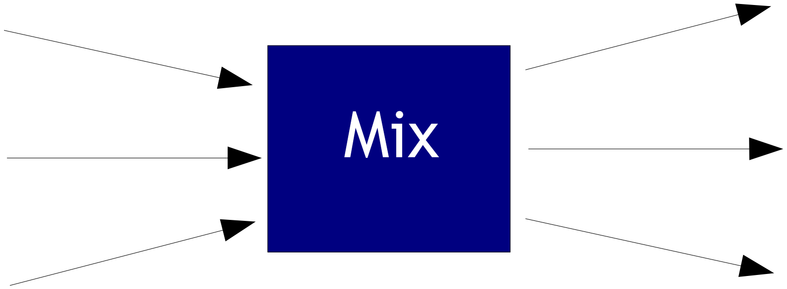
How do we do this?

- At the beginning there was David Chaum's (1981) **mix.**
- What is a mix:
  - Router that takes messages and send them out.
  - Mixes hide the correspondence between inputs and outputs – hence anonymity!



How do we do this?

- At the beginning there was David Chaum's (1981) **mix.**
- What is a mix:
  - Router that takes messages and send them out.
  - Mixes hide the correspondence between inputs and outputs - hence anonymity!

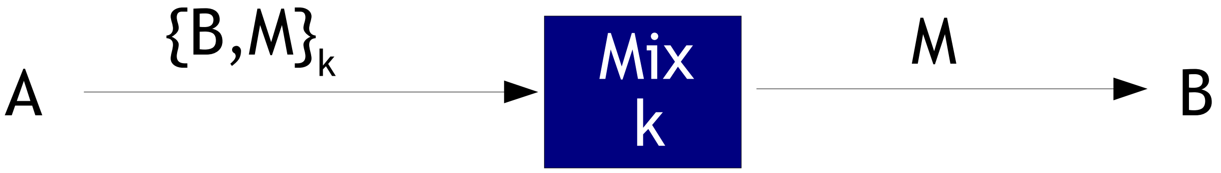


How to design a mix?

- Messages in and out have to look different.
  - Bitwise unlinkability: use cryptography.
- Timing of arrivals and departures must not link messages.
  - Traffic analysis resistance: use batching strategies, and dummy traffic.
- Other attacks: flooding, DoS, network discovery, sting attacks, … black magic!

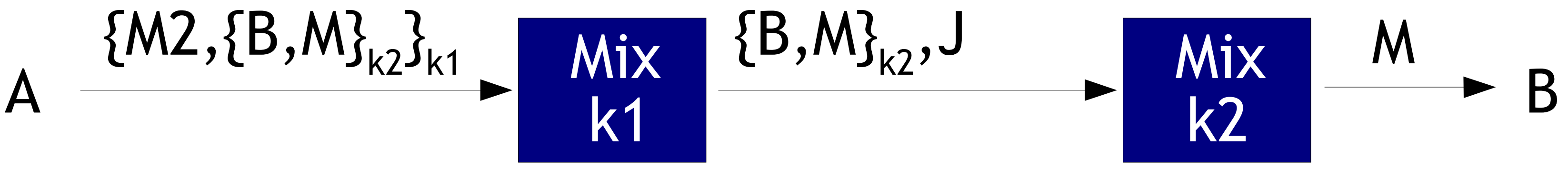
An insecure example

- A simple construction:



A $\xrightarrow{\{B,M\}_k}$ Mix k $\xrightarrow{M}$ B

- Chaining mixes:

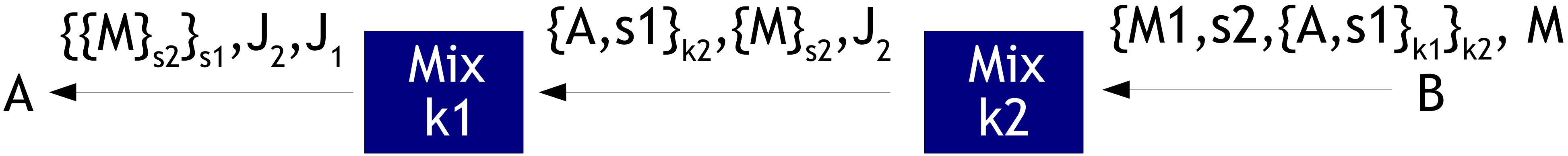


A $\xrightarrow{\{M2,\{B,M\}_{k2}\}_{k1}}$ Mix k1 $\xrightarrow{\{B,M\}_{k2},J}$ Mix k2 $\xrightarrow{M}$ B

More requirements: select honest routes, hide total number of hops, hide from corrupt mixes, Topology, …

How to reply anonymously?

- Alice sends **reply blocks** to Bob, so that he can route messages back.


$\{\{M\}_{s2}\}_{s1},J_2,J_1$
A
Mix k1
$\{A,s1\}_{k2},\{M\}_{s2},J_2$
Mix k2
$\{M1,s2,\{A,s1\}_{k1}\}_{k2}$, M
B

- More requirements:
  - Path length of reply blocks not leaked.
  - Intermediaries do not know their positions.
  - Replies must not be distinguishable from normal.

After Chaum …

- Three main branches of anonymous comms:
  - **Remailers** - mixing email-like traffic.
  - Onion Routing – ISDN, JAP, Tor, … (Roger's talk)
  - Provable schemes – elections (hardcore crypto)
- Non-mix based systems:
  - Simple proxies / Crowds (weak!)
  - Dining Cryptographers networks (very strong!)
  - Cool hacks: wireless, steganography, …

Theoretical schemes

- Schemes:
  - Babel - remailer
  - Sg-mixes – to combat (n-1) attacks
  - Moller's provable mix
  - Minx – Very efficient packet format.
- Analysis:
  - Measuring anonymity (information theory / covert channel analysis).

Theoretical schemes (cont)

  - Mix strategies and dummy traffic.
  - Topologies (cascades, restricted routing, synchronous batching, …)
  - Tagging attacks - original Chaum mix fails!
  - Simulation
- Analysis of attacks (we are good at it now):
  - Disclosure and statistical disclosure.
  - Traffic analysis
  - Network discovery attacks.

Stone age remailer: penet.fi

- 1993 - penet.fi by Johan Helsingius
- Simple email proxy:
  - Strips identifying headers.
  - Substitutes an nym address, to route back replies.
  - Correspondance is kept in a large file!
- 1996 – Legal attack – penet.fi loses.
- Impact on anonymity community.

Type I “cypherpunk” remailers

- Appears on the cypherpunk list
  - At the time cypherpunks wrote code :-)
- Fixes the “one large file” problem.
- Uses PGP 2 for crypto (weak!) - tagging & no padding.
- Many remailers can be chained.
- Reply blocks can be used (more than once) to reply to messages. Still in use!

Type II “Mixmaster” remailer

- Lance Cottrell (1995), Ulf Moller,Peter Palfrader, Len Sassaman++
- Custom crypto to avoid tagging attacks and replays.
- Fixed size payload & split messages.
- No **reply blocks**.
- Overall secure and maintained.

Type III “Mixminion” remailers

- A serious effort: Dingledine, Mathewson, Danezis, Zooko, Hopwood, Mazieres, Mixmaster crew …
- Allows anonymous sending (32kb).
- Indistinguishable single use reply blocks (4kb).
- Implements all features described.
- Forward secure custom transport (not SMTP)
- Can do better but it is state of the art!
  - Do not reinvent it!

Mixminion – a bit more technical

- Written in Python with a bit of C.
  (praise Nick Mathewson!)
- In alpha but stable and useable.
- Good documentation: design documents, specifications, documented code,
- Responsive and archived mailing list.
- Around 30 volunteers running servers.
- But more to do …

What is to be done?

- Infrastructure work:
  - Trust management as network grows.
  - Reliable two way anonymity.
- Integration and services work:
  - Usable clients.
  - Nym servers and other protocol gateways.
- The stuff no one likes doing:
  - User documentation, FAQ, evangelism, website, logo, …

Trust infrastructure - Directories

- Directory services: to disseminate key material about **all** remailers. High-availability, high integrity!
- If some are missing there might be a pattern.
- Adversary to populate the directory (sybil attacks.) / get the honest ones out.
- How do we distribute this function? How do we allow nodes to trust different subsets?

Trust infrastructure - Reliability

- Adversary will try to disrupt communications to put people off using Mixminion.
- Pingers constantly test the state of the network (Peter Palfrader – echolot).
- Open questions:
  - Can we do better? More efficient?
  - Is it safe? (false sense of traffic, lots of info).
  - How can clients use it – without attacks?
  - Reputation? Aaaahhhh…

Reliable transmission

- Mixminion cannot guarantee that messages arrive.
  - Use forward error correcting codes.
  - Make sure not prone to traffic analysis.
- Need to include SURBs for replies.
  - Standard way to do so does not leak info.
  - How to make sure one does not run out.
- Combine the two to have reliable two way anonymous comms.

Mixminion to email: nym servers

- Nym servers act as a bridge between normal email and anonymous email.
- Can send normal email and it is sent anonymously to recipient (David Mazieres).
- Many architectural options:
  - Use a list of SURBs per nym.
  - Poll by sending a bunch of SURBs.
  - Use private information retrieval.
- Specs available, waiting to be implemented.

Usable clients

- Users have to be attracted – usability is security (the more the merrier).
- Option A: write them from scratch.
  - Advantages: security design from the beginning, no unforeseen feature interaction.
  - Disadvantages: A lot of work, slow development, unfamiliar and not integrated environment.

Usable clients (cont.)

- Option B: Client integration (plug-in to provide anonymity check box)
  - Advantages: quicker development, more infrastructure there, familiar environment.
  - Disadvantages: Feature interaction, some filtering required, how to make sure the user does not do something silly?
- Who knows how to write Thunderbird extensions or … outlook plug-ins?

Usable clients (cont.)

- Option C: Anonymous Proxies
  - SMTP server that sends anonymous mail
  - POP3 server that receives anonymous mail.
  - Advantages: very familiar environment, can configure a proxy for whole VPN/intranet, easier to code.
  - Disadvantages: Heavy filtering required, can users configure an SMTP/POP client?
- Prototype already available with Mixminion.

Wild/Research ideas

- Integrate the aTCP with nym servers to provide a peer-to-peer nym service.
- How do we secure large (100s Mbs) downloads over mixminion? (back to traffic analysis).
- How do we make Mixminion SURBs forward secure?
- How do we integrate Mixminion and other (Tor?) into an a Linux distribution?

In conclusion

- High latency type III remailer is the most secure anonymous communications medium we have.
- Mixminion is a robust protocol jet more work is needed in areas surrounding it.
- A lot of integration work has to be done.
- **You** are the people **you** have been looking for!

I want more!

- State of the art in anonymity research:
  - Bibliography http://www.freehaven.net/anonbib/
  - Privacy Enhancing Technologies Workshop http://petworkshop.org
- The real thing:
  - Mixminion http://mixminion.net
  - Tor http://tor.eff.org/