Anonymous communication

Tariq Elahi[1]
School of Informatics
University of Edinburgh

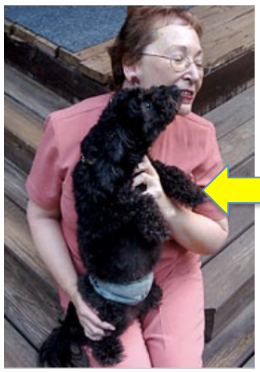
February 28, 2020

[1]With slides developed by Myrto Arapinis

Context

- The Internet is a public network:
  - network routers see all traffic that passes through them
- Routing information is public:
  - IP packet headers contain source and destination of packets
- Encryption does not hide identities:
  - encryption hides payload, but not routing information

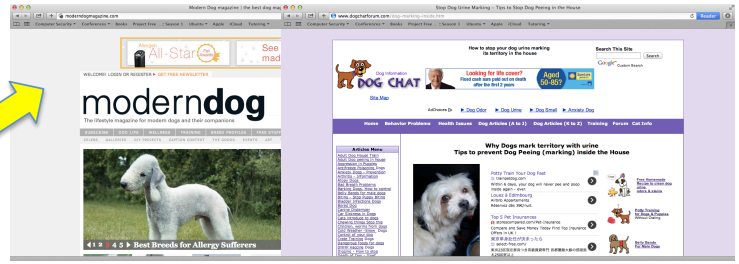
Routing information can reveal who you are!



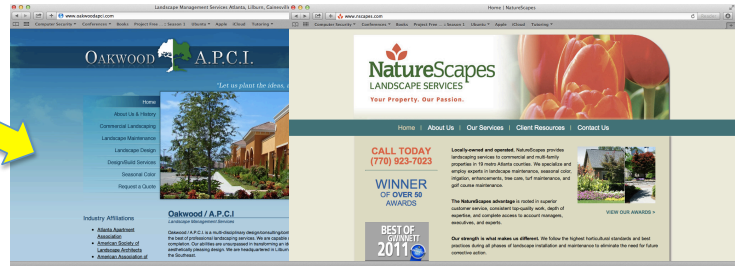
$X_1.X_2.X_3.X_4$

INTERNET

66.154.48.145 79.170.40.232



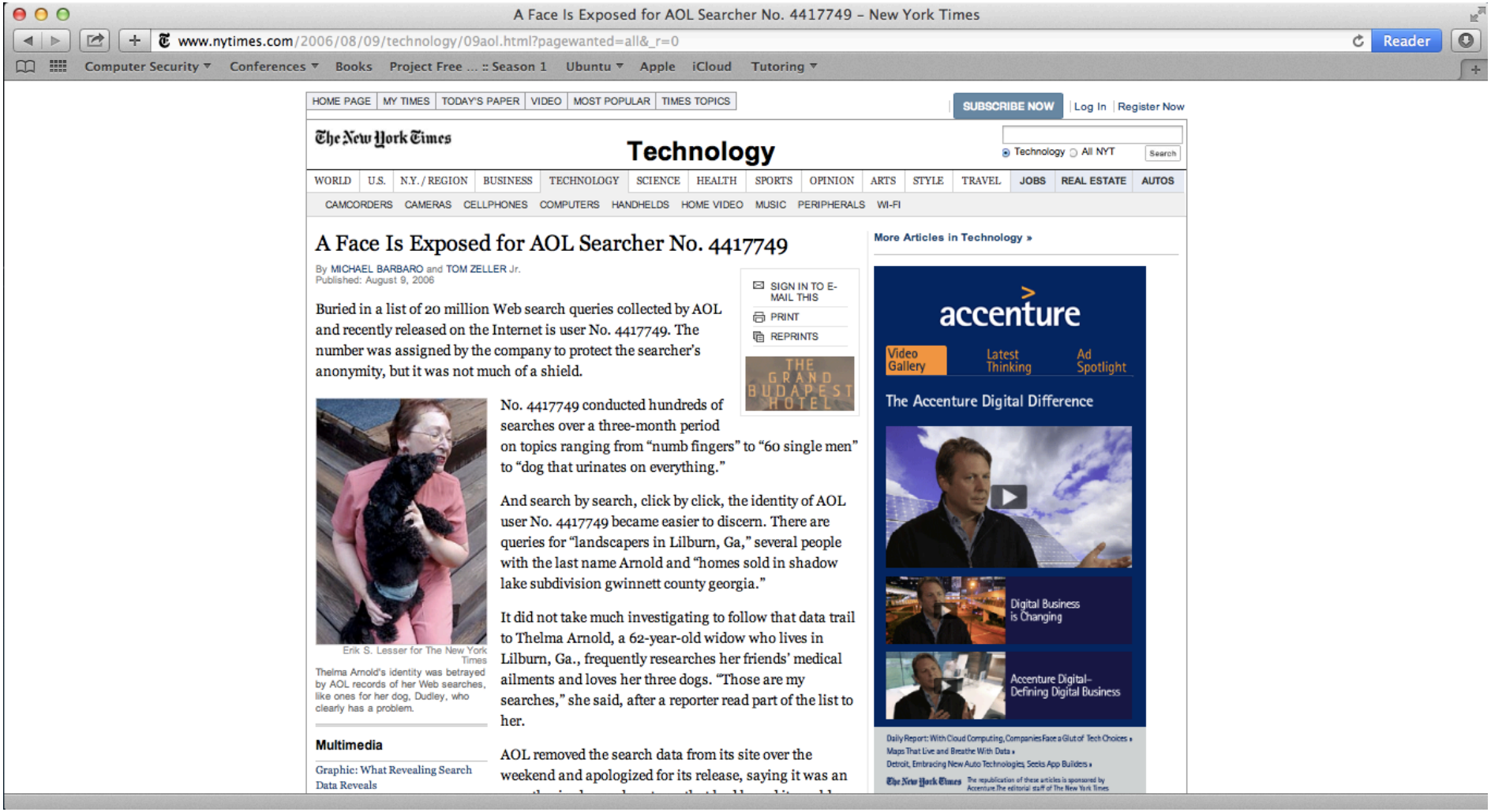
64.131.67.188 192.185.48.150



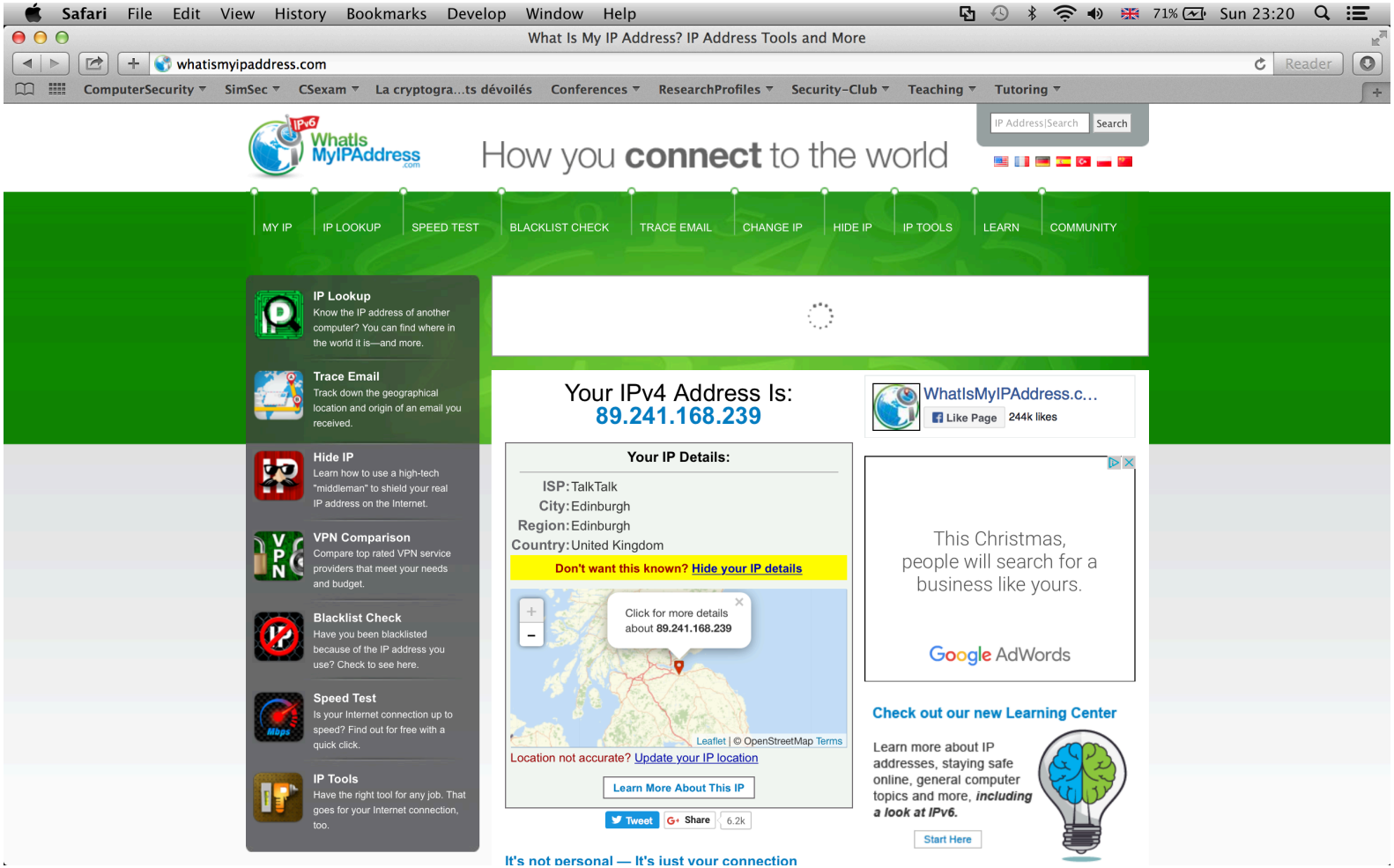
108.168.213.84 192.254.187.105

216.240.189.1 69.25.28.142

Routing information can reveal who you are!



Routing information can reveal who you are!



Routing information can reveal who you are!

"With your permission, you give us more information about you, about your friends, and we can improve the quality of your searches. We don't need you to type at all. We know where you are. We know where you've been. We can more or less know what you're thinking about."

Eric Schmidt, CEO Google, 2010

Your IP address is your ID

Your IP address leaves behind digital tracks that can be used to identify you and invade your privacy

The McNealy argument

"You have zero privacy anyway. Get over it"
Scott McNealy, CEO Sun Microsystems, 1999

The Schmidt argument

"If you have something that you don't want anyone to know maybe you shouldn't be doing it in the first place"

Eric Schmidt, CEO Google, 2009

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

Anonymity

Definition (ISO/IEC standard 15408)

A user may use a service or resource without disclosing the user's identity.

$\longrightarrow$ this can be achieved by hiding one's activities among others' similar activities

- Dining cryptographers
- Crowds
- Chaum's mix
- Onion routing

Three-party dining cryptographers (3DC)

Three NSA cryptographers are having dinner. At the end of the dinner they are informed that the dinner has already been paid for. Now, either the NSA paid for the dinner, or one of the cryptographers did. They want to know if it is the NSA that paid, or one of them, **without** revealing the identity of the paying cryptographer.

3DC protocol:

1. **Phase 1:** Each pair of cryptographers flips a coin (that only they can see), where heads $= 1$ and tails $= 0$
   - each cryptographer will see two coin flips.

2. **Phase 2:** Each participant publicly announces the result as follows:
   - **Did NOT pay:** the XOR of the two coin flips they observed
   - **Did pay:** the *negation* of the XOR of the two coin flips they observed

3. **Resolution:** If the XOR of the three announcements is:
   - 0**:** The NSA paid
   - 1**:** One of them paid (but only the payer will know this.)

Superposed sending

- 3DC protocol generalises to any group size $n > 2$ (nDC)
- Sender wants to anonymously broadcast a message m. For each bit, m_i, of m:
  1. every pair of users generate a random bit $(0, 1)$
     - every user observes $n - 1$ bits.
  2. each user (except the sender) announces (XOR of all $n - 1$ observed bits)
  3. the sender announces (XOR of all $n - 1$ observed bits and m_i)
  4. XOR of all announcements $= m_i$
     - every randomly generated bit occurs in this sum twice (and is cancelled by XOR)
     - m_i occurs only once

Limitations of the DC protocol

The DC protocol is impractical:

- Requires pair-wise shared secret keys (secure channels) between the participants (to share random bits)
- Requires large amounts of randomness
- Any one can launch a denial of service attack by either not performing the protocol properly, or by transmitting at the same time as some one else. Only one person can transmit at any given round.

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

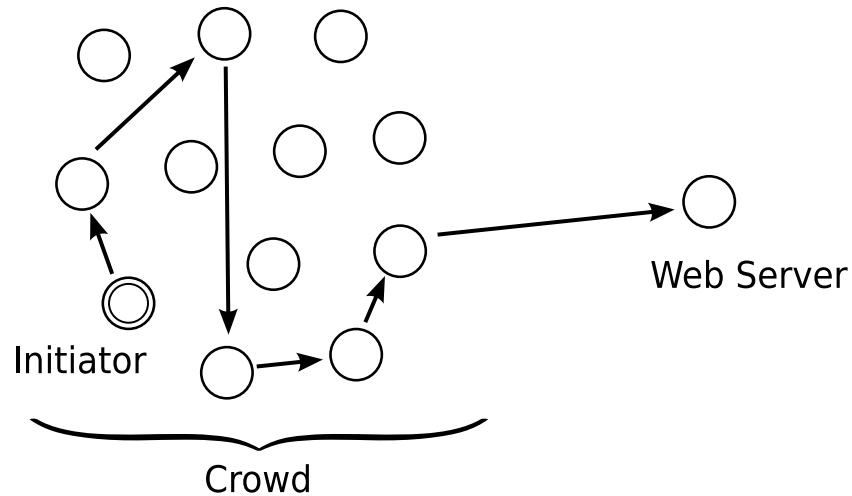
Idea: randomly route the request through a crowd of users

Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- a crowd is a group of m users; c out of m users may be corrupted

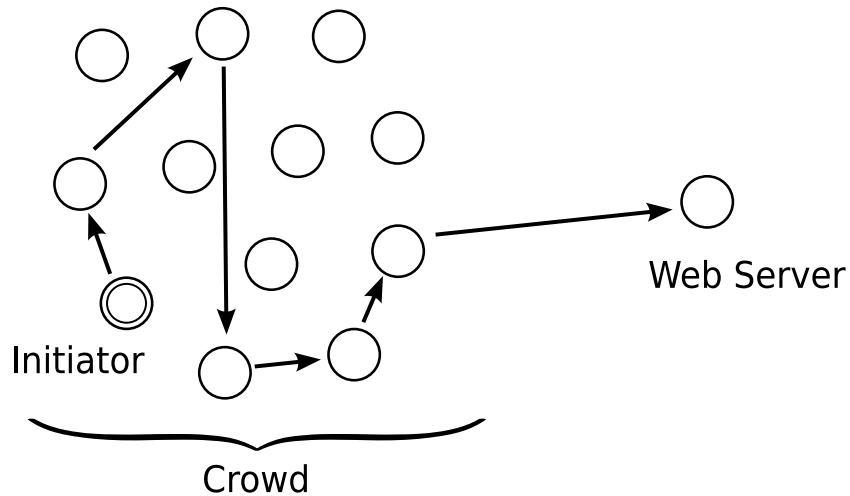


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- a crowd is a group of m users; c out of m users may be corrupted
- an initiator that wants to request a webpage creates a path between him and the server:

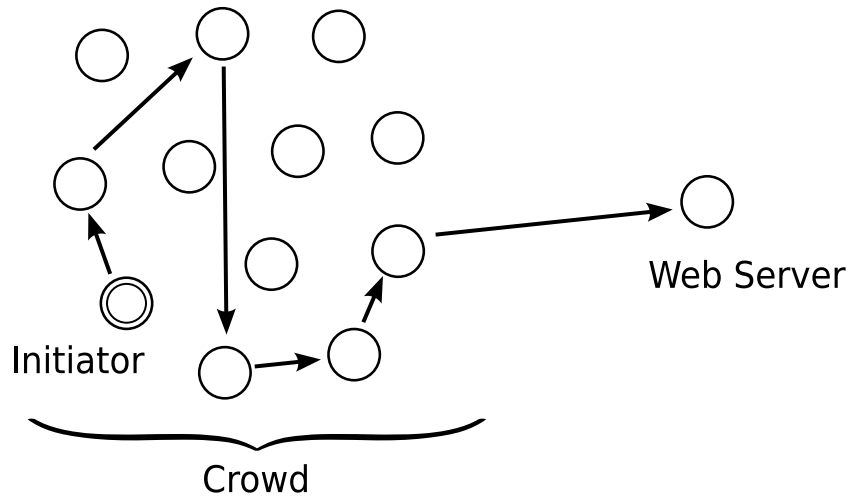


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- a crowd is a group of m users; c out of m users may be corrupted
- an initiator that wants to request a webpage creates a path between him and the server:
  1. the initiator selects a forwarder from the crowd and sends him his request

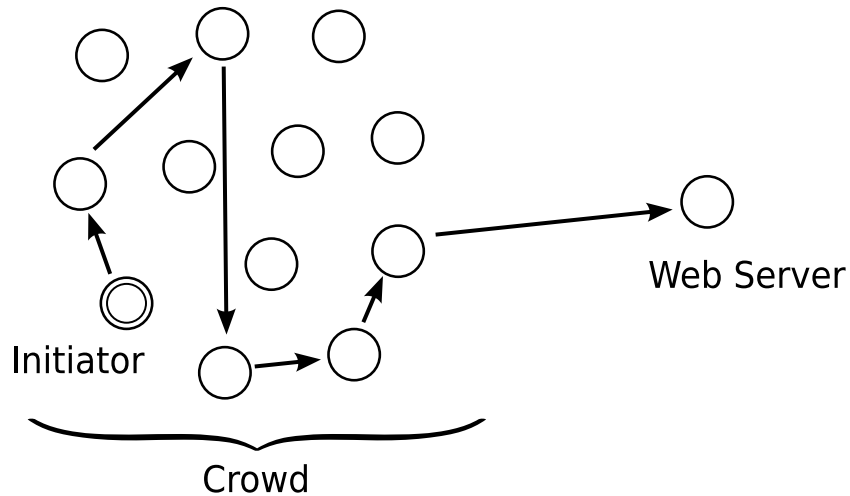


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- a crowd is a group of m users; c out of m users may be corrupted
- an initiator that wants to request a webpage creates a path between him and the server:
  1. the initiator selects a forwarder from the crowd and sends him his request
  2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f; the new forwarder repeats the procedure

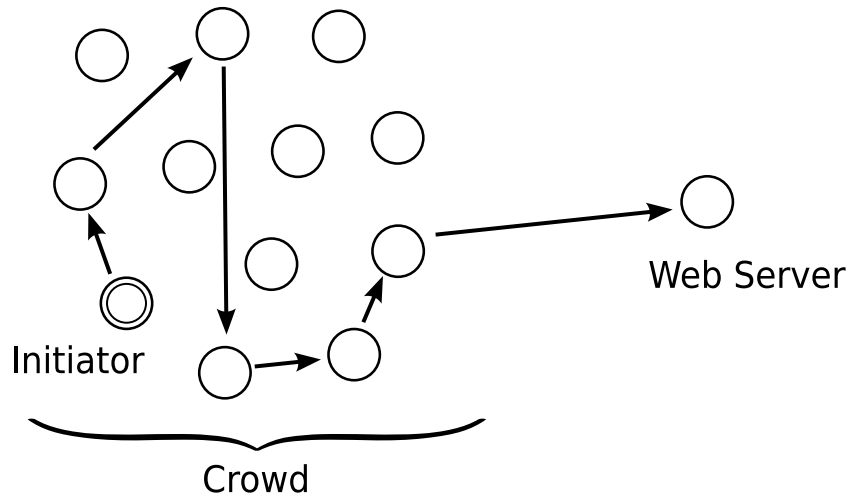


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

- a crowd is a group of m users; c out of m users may be corrupted
- an initiator that wants to request a webpage creates a path between him and the server:
  1. the initiator selects a forwarder from the crowd and sends him his request
  2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f; the new forwarder repeats the procedure
  3. the response from the server follows same route in opposite direction

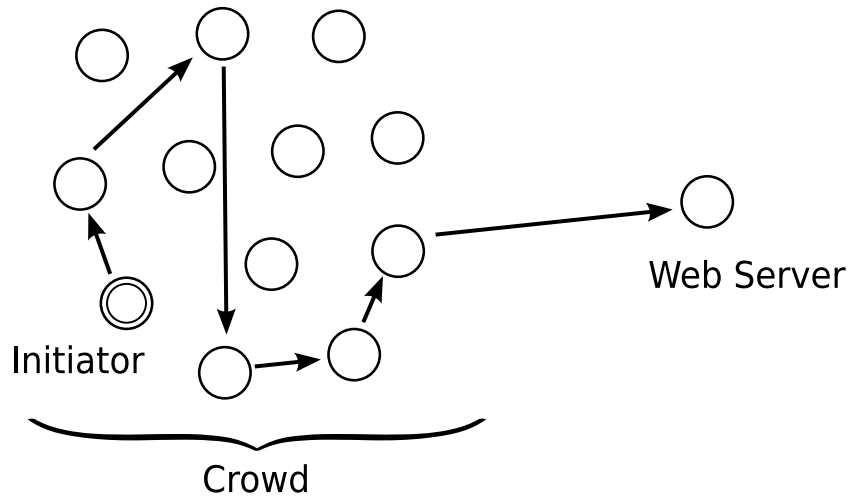


Crowds

[M. K. Reiter and A. D. Rubin, "Crowds: anonymity for Web transactions". ACM Transactions on Information and System Security.]

Idea: randomly route the request through a crowd of users

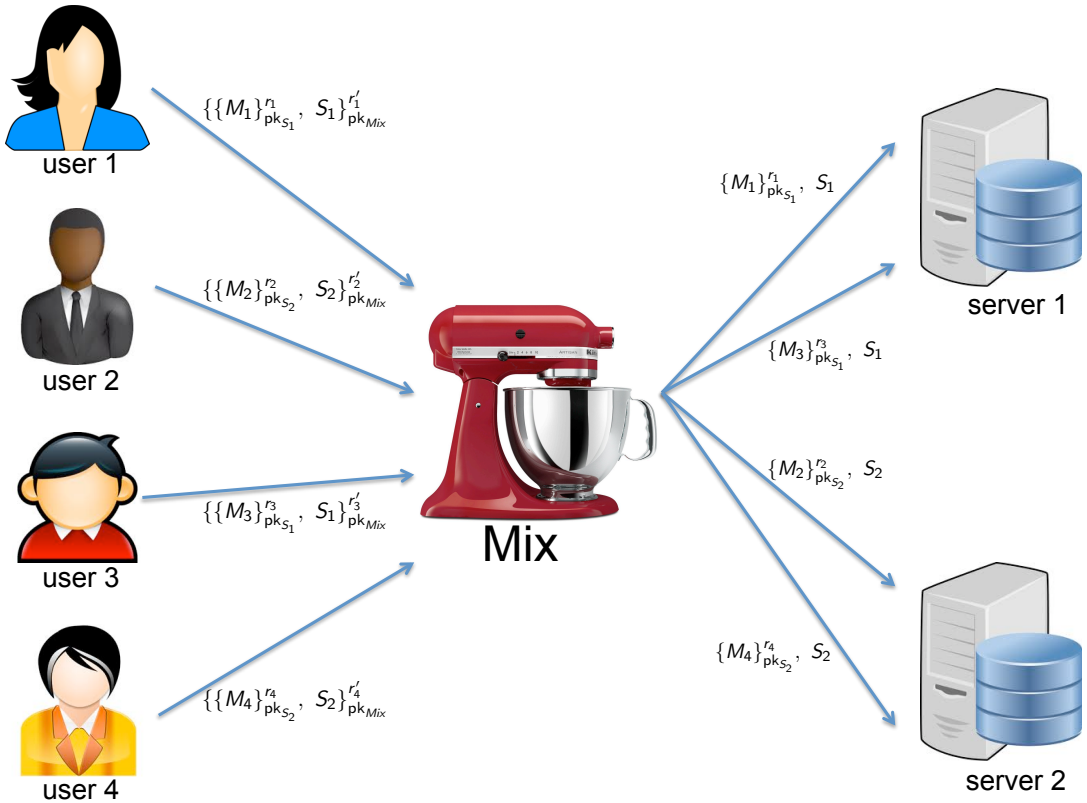
- a crowd is a group of m users; c out of m users may be corrupted
- an initiator that wants to request a webpage creates a path between him and the server:
  1. the initiator selects a forwarder from the crowd and sends him his request
  2. a forwarder delivers the request directly to the server with probability $1 - p_f$; he forwards the request to a randomly selected new forwarder from the crowd with probability p_f; the new forwarder repeats the procedure
  3. the response from the server follows same route in opposite direction



Crowd IS NOT resistant against an attacker that sees the whole network traffic!

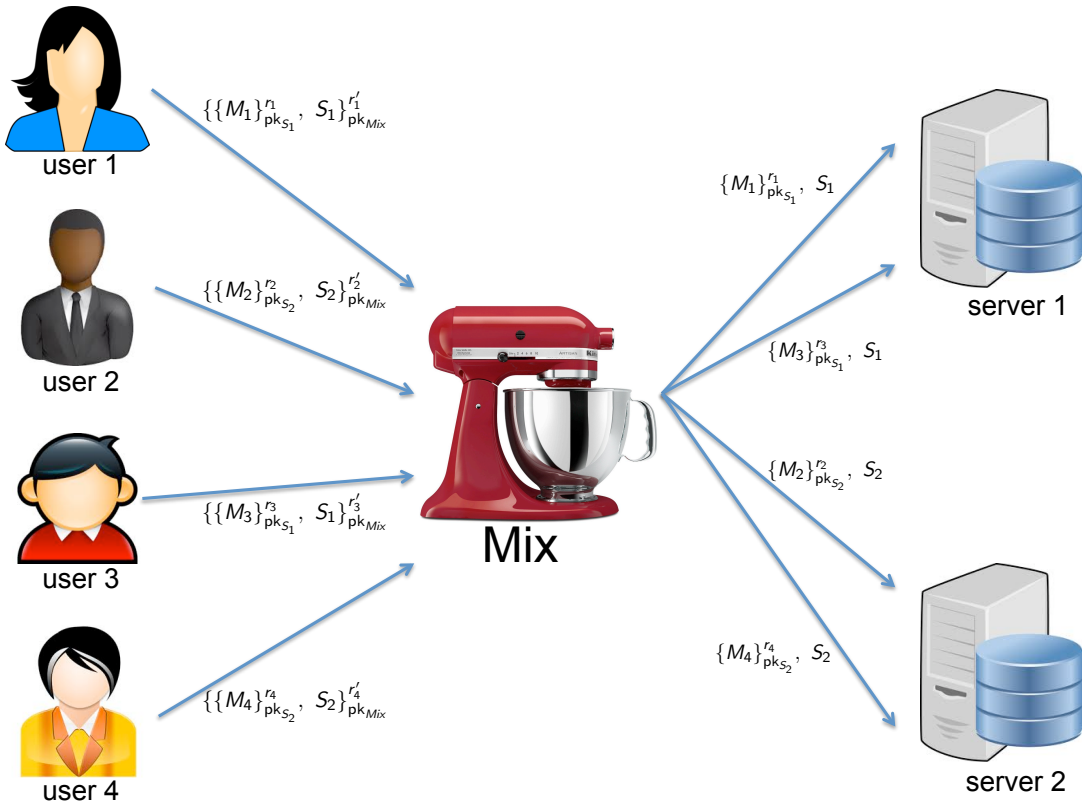
Chaum's mix

[D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", Communications of the ACM, February 1981.]



Chaum's mix

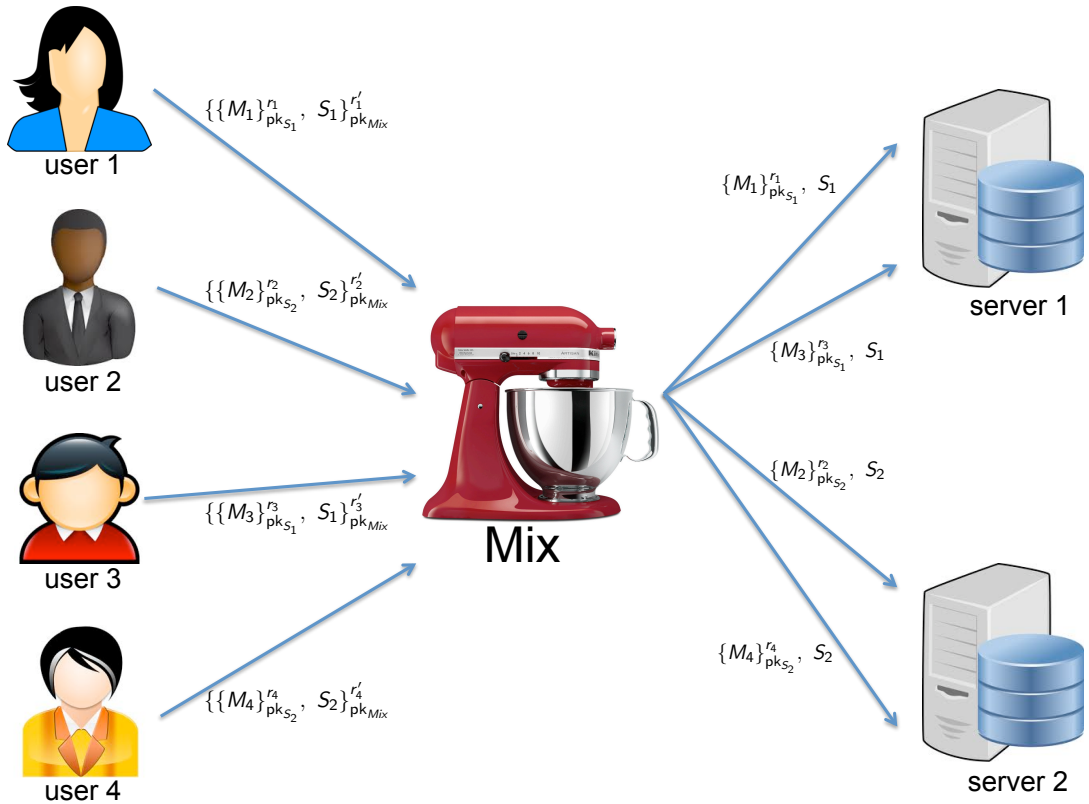
[D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", Communications of the ACM, February 1981.]



- message padding and buffering to avoid time correlation attacks

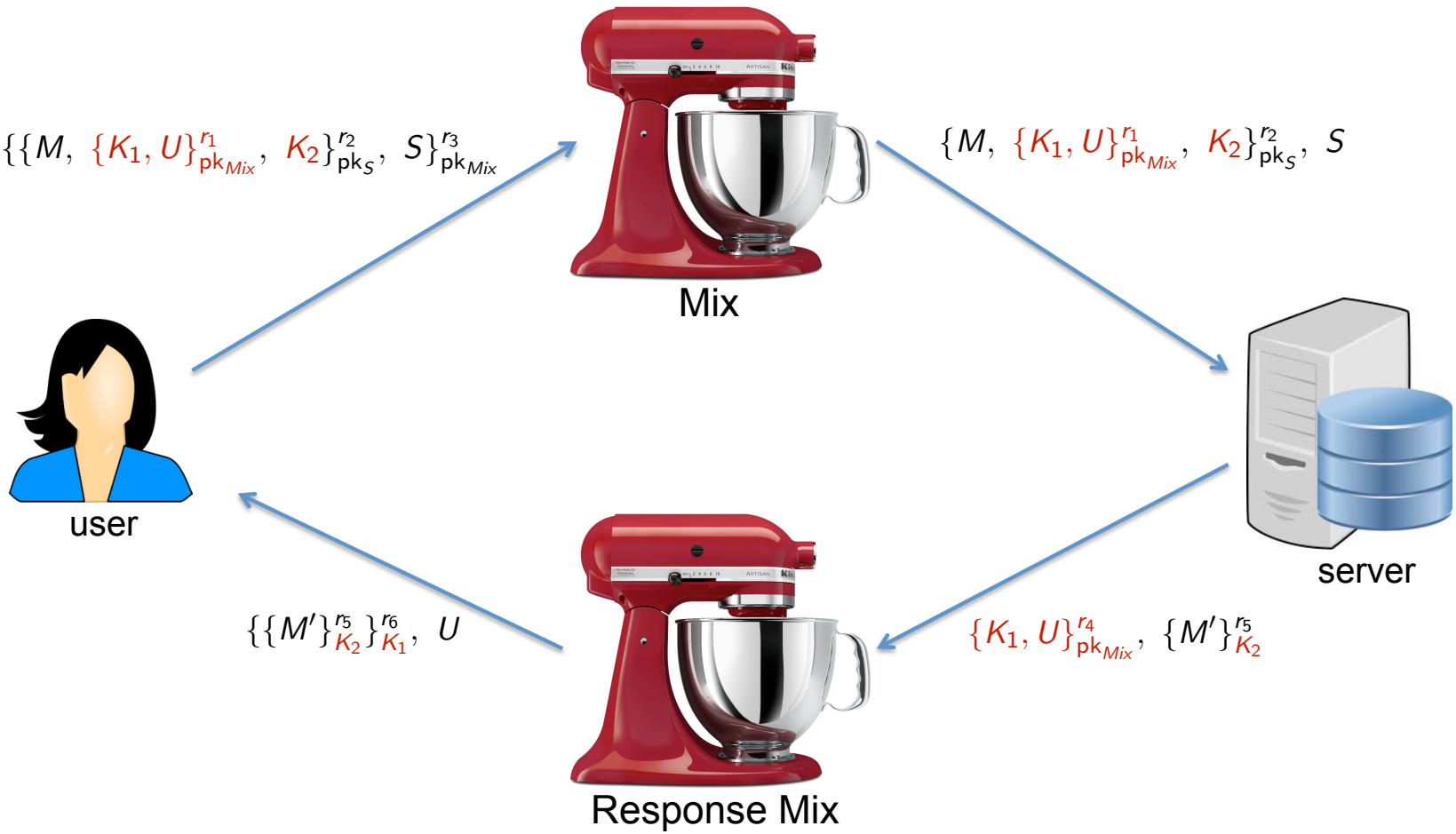
Chaum's mix

[D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms", Communications of the ACM, February 1981.]

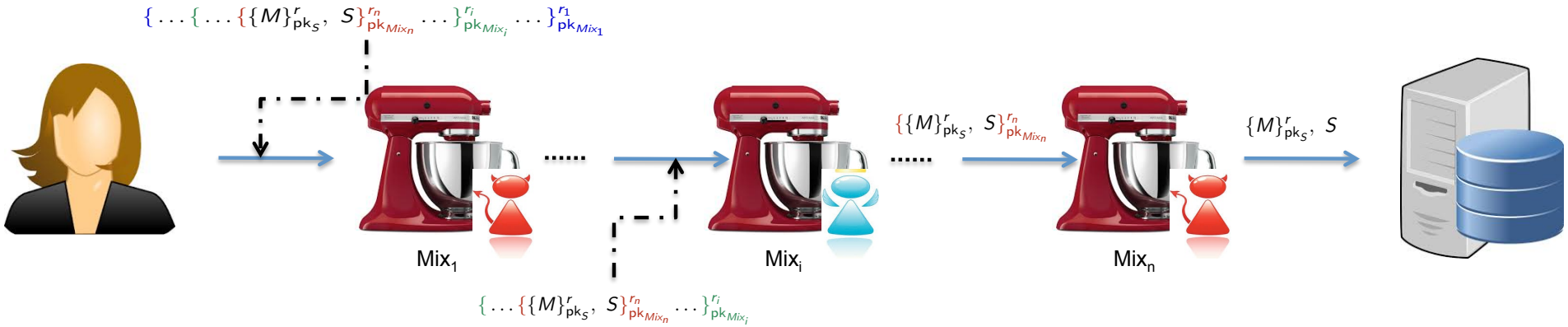


- message padding and buffering to avoid time correlation attacks
- dummy messages are generated by the mixes themselves to prevent an attacker sending $n-1$ messages to a mix with capacity n, allowing him to then link the sender of the n^{th} message with its recipient

Anonymous return addresses



Mix cascade



- messages are sent through a sequence of mixes
- some of the mixes may be corrupted
- a single honest mix guarantees anonymity against an attacker controlling the whole network provided it applies:
  - message padding
  - buffering
  - dummy messages

Limitations of Chaum's mixnets

- Asymmetric encryption is not efficient
- Dummy messages are inefficient
- Buffering is not efficient