

## Anonymous Systems

## Today's Lecture

- Theoretical anonymity.
  - Dining Cryptographers Protocol
  - Definitions of Anonymity
  - The Crowds protocol
  - Mixes
  - Onion Routing
- Practical anonymous systems
  - Tor system
  - Mixminion
  - MUTE
  - Freenet

**“You have zero privacy anyway, get over it”**

Scott McNealy, CEO of SUN Microsystems.

## The Theory of Anonymity



**“On the Internet nobody knows you're a dog”**

Anonymity means different things to different users.

The right definitions are key to understand any system.

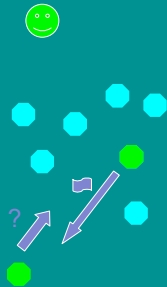
## The Theory of Anonymity

- Anonymity is a difficult notion to define.
  - Systems have multiple agents
  - which have different views of the system
  - and wish to hide different actions
  - to variable levels.
- Sometimes you just want some doubt, sometimes you want to act unseen.

## The Theory of Anonymity

- In a system of anonymous communication you can be:
  - A sender
  - A receive / responder
  - A helpful node in the system
  - An outsider (who may see all or just some of the communications).
- We might want anonymity for any of these, from any of these.

## Example: Anonymous File-Sharing



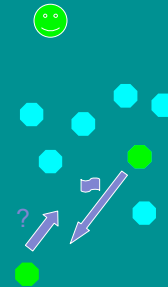
One node sends a request for a file (sender)

Other nodes receive this request (the nodes)

Maybe one of the nodes replies with a file (receiver/responder).

The attacker may be any of these or an outside observer.

## Example: Anonymous File-Sharing



- The user may wish to hide
  - that they are offering files
  - that they are taking part in data transfer
  - that they are running the software at all.

- The user may want to have plausible deniability or go complete unnoticed.

## Levels of Anonymity

Reiter and Rubin provide the classification:

- **Beyond suspicion:** the user appears no more likely to have acted than any other.
- **Probable innocence:** the user appears no more likely to have acted than to not to have.
- **Possible innocence:** there is a nontrivial probability that it was not the user.

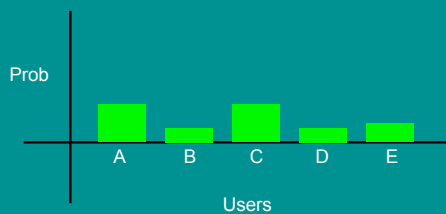
## Beyond suspicion

- A is Beyond suspicion:



## Beyond suspicion

- A is not Beyond Suspicion:



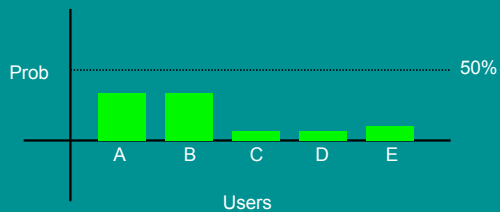
## Probable Innocence

- A is Probably Innocence



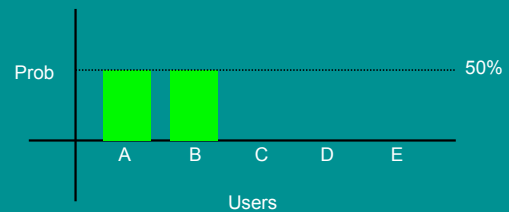
## Probable Innocence

- A is Probably Innocence



## Probable Innocence

- A is Probably Innocence



## Probable Innocence

- A is Probably Innocence



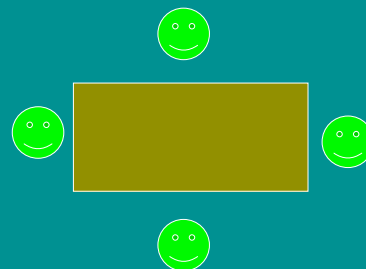
## Summary: The Theory of Anonymity

- There are many agents in a system each of which have different views.
- There are a number of different actions.
- We need to define the *level of anonymity* an *user* has when performing a *certain action*, given the attacker's *view of the system*.

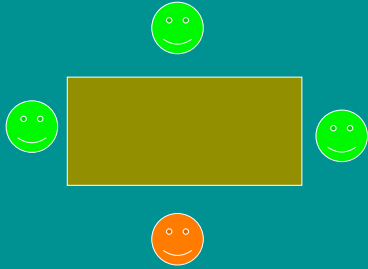
## Outline of Talk

- The theory of anonymity.
- Designs for anonymity.
- Real Anonymous Systems.

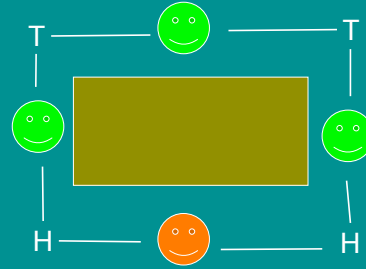
## The Dining Cryptographers Protocol



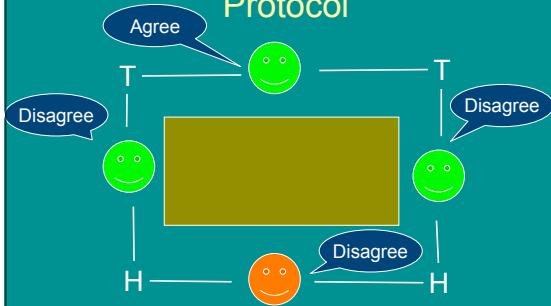
## The Dining Cryptographers Protocol



## The Dining Cryptographers Protocol

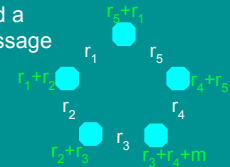


## The Dining Cryptographers Protocol



## Dining Cryptographers

- Nodes form a ring
- Each adjacent pair picks a random number
- Each node broadcasts the sum (xor) of the adjacent numbers
- The user who wants to send a message also adds the message
- The total sum (xor) is:
 
$$r_1 + r_2 + r_2 + r_3 + r_3 + r_4 + r_4 + r_5 + r_5 + r_1 + m = m$$

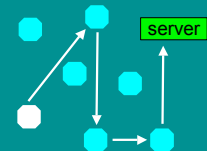


## Dinning Cryptographers

- It's impossible to tell who added  $m$ .
- Beyond suspicion even to a global attacker.
- Very inefficient: everyone must send the same amount of data as the real sender.

## Crowds

- A crowd is a group of  $n$  nodes
- The *initiator* selects randomly a node (called *forwarder*) and forwards the request to it
- A forwarder:
  - With prob.  $1-p_f$ , selects randomly a new node and forwards the request to him
  - With prob.  $p_f$ , sends the request to the server

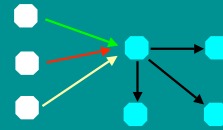


## Crowds

- The sender is beyond suspicion to the server.
- Some of the nodes could be corrupted.
- The initiator could forward the message to a corrupted node.
- The sender has probable innocence to other nodes.

## MIXes

- MIXes are proxies that forward messages between them
- A user contacts a MIX to send a message
- The MIX waits until it has received a number of messages, then forwards them in different order



## MIXes

- It is difficult to trace the route of each message.
- Provides beyond suspicion S-R unlinkability even to a global attacker.
- Messages have to be delayed (can be solved with dummy traffic).
- More complicated when sending series of packets

## Onion Routing

- Messages are routed through a number of nodes called Core Onion Routers (COR)
- The initiator selects the whole route and encrypts the message with all keys in reverse order
- Each node unwraps a layer (onion) and forwards the message to the next one



## Onion Routing

- Each node only learns the next one in the path
- Can be used together with MIXing.
- End-users can run their own COR
  - Better anonymity
- or use an existing one
  - More efficient
  - User's identity is revealed to the COR

## Mutli-casting

- Broadcast the message to the whole network.
- Provides beyond suspicion for the receiver.
- No anonymity for the sender.
- Multicasting is a good technique for broadcasting messages.
- but very inefficient to send just one message.

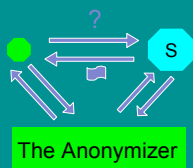
## Spooferd UDP

- IP packets on the Internet contain the IP address of the sender.
- This address is not used by routers, only by higher-level protocols such as TCP.
- UDP does not use this address.
- A random address can be used instead to provide sender anonymity.
- Method prohibited by many ISPs.

## Outline of Talk

- The theory of anonymity.
- Designs for anonymity.
- Real Anonymous Systems.

## The Anonymizer



An Internet connection reveals your IP number.

The Anonymizer promise "Anonymity"

Connection made via The Anonymizer.

The Server see only the Anonymizer.

## The Anonymizer

The sender is **Beyond Suspicion** to the server.

The server knows The Anonymizer is being used.

If there is enough other traffic, you are **Probably Innocence** to a global observer.

The global observer knows you are using the "The Anonymizer"

There is no anonymity to the "The Anonymizer"

## The Anonymizer

- From the small print:
- ... we disclose personal information only in the good faith belief that we are required to do so by law, or that doing so is **reasonably necessary** ...
- ... Note to European Customers: The information you provide us **will** be transferred outside the European Economic Area

## Tor

- Tor is an anonymous transport layer.
- It does not implement a file-sharing but file-sharing software can be run on top of it.
- Tor implements onion routing without MIXes.
- Its possible that a program run on top of Tor will reveal its IP address.

## Tor does not provide security

- In 2007 Dan Egerstad set up his own Tor exit node and monitored all the traffic.

## Tor does not provide security

- In 2007 Dan Egerstad set up his own Tor exit node and monitored all the traffic.
- He found unencrypted e-mail traffic including passwords from
  - foreign ministry of Iran,
  - Kazakh and Indian embassies in the U.S.
  - Russian embassy in Sweden.

## Tor does not provide security

- In 2007 Dan Egerstad set up his own Tor exit node and monitored all the traffic.
- He found unencrypted e-mail traffic including passwords from
  - foreign ministry of Iran,
  - Kazakh and Indian embassies in the U.S.
  - Russian embassy in Sweden.
- He got arrested for this.

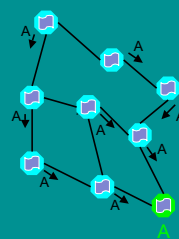
## Mixminion

- Mixminion is an anonymous e-mail system.
- It consists of a network of nodes that MIX messages before forwarding them as e-mail.
- It uses onion routing and mixes traffic.

## Anonymous File Sharing MUTE

- MUTE removes the IP address from the file exchange.
- Peers only know the IP address of their direct neighbours.
- Peers choose random "pseudo ID".
- Files are not sent directly between peers. Instead files are sent via a number of peers.
- MUTE uses a version of the "Ants" ad-hoc routing protocol.

## MUTE: Search

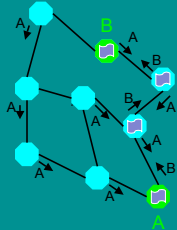


The search takes place as before, but this time the message uses its pseudo ID as the "from ID".

Each peer builds a routing table by records the ID and the connection.

A probabilistic time-to-live counter limits the search.

## MUTE: Reply



If B wants to reply it sends a message to A's pseudo ID.

This message is routed using the ad-hoc routing table.

The route to B is also recorded

## Anonymity Provided by MUTE

- MUTE makes it hard to link the IP address of a peer with its pseudo ID.
- Peers only know the ID address's of their direct neighbours.
- The network should provide enough cover to let a neighbour deny using a particular ID.
- If an attacker can completely surround a peer it loses anonymity.

## Freenet

- Freenet is an "anonymous publishing system".
- Anyone can publish by sending to a node.
- Search method similar to MUTE but only use a single search message.
- Based on "Distributed Hash Table".

## Outline of Talk

- The theory of anonymity.
- Designs for anonymity.
- Real Anonymous Systems.

Questions?