CS 259

Protocols for Anonymity

Vitaly Shmatikov

Overview

- Basic concepts of anonymity
  - Chaum's MIX
  - Dining cryptographers
  - Knowledge-based definitions of anonymity
- Probabilistic anonymity
  - Onion Routing
  - Crowds
- Introduction to probabilistic model checking
  - Using a probabilistic model checker to analyze Crowds

Applications of Anonymity

- Privacy
  - Hide online transactions, Web browsing, etc. from intrusive governments, corporations and archivists
- Digital cash — **Good topic for a project**
  - Electronic currency with properties of paper money
- Anonymous electronic voting — **Good topic for a project**
- Censorship-resistant publishing
- Untraceable electronic mail
- Crypto-anarchy
  - “Some people say `anarchy won't work’. That's not an argument against anarchy; that's an argument against work.” – Bob Black

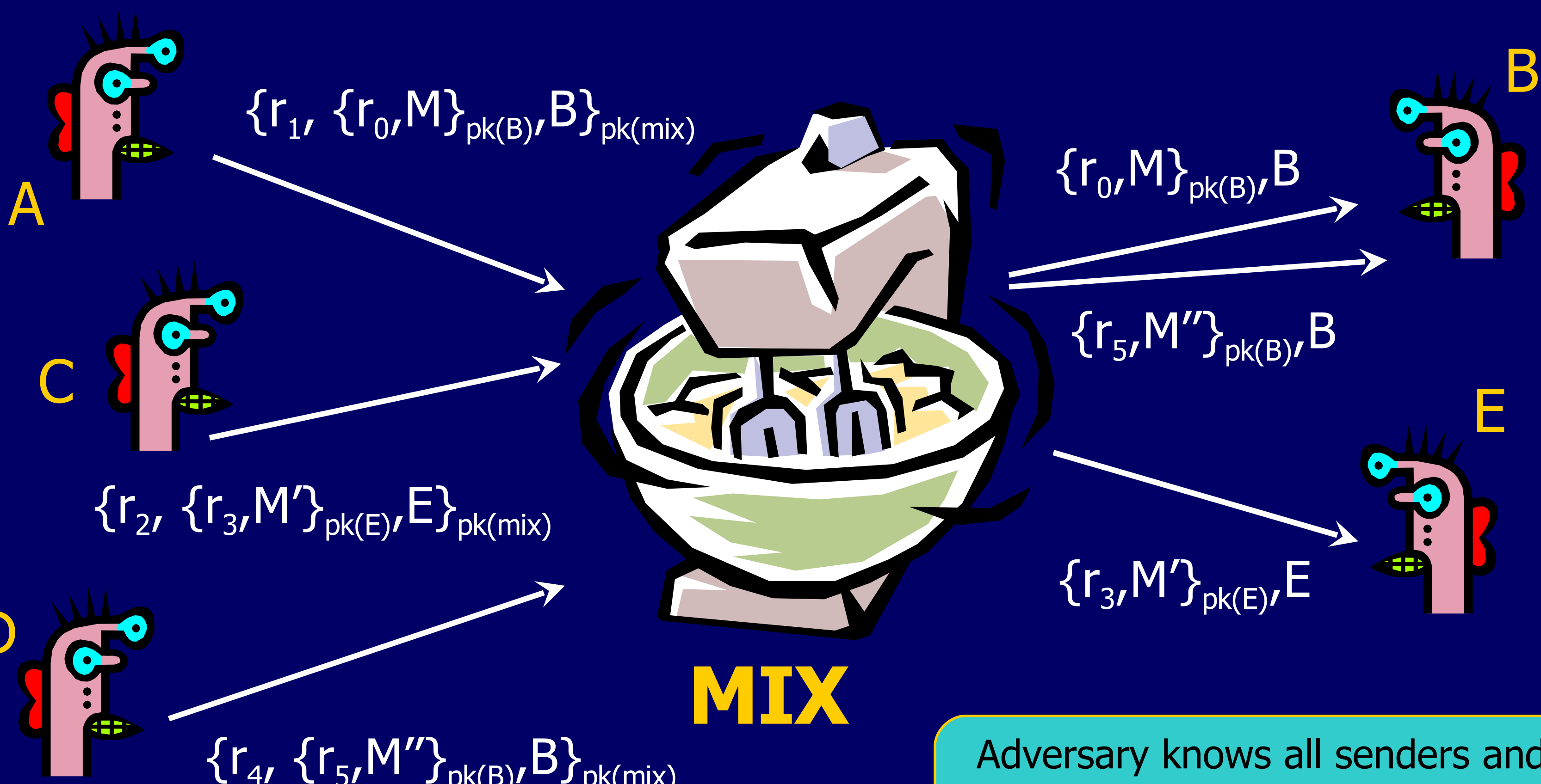
Chaum's MIX

- Early proposal for anonymous email
  - David Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms". Communications of the ACM, February 1981.

Before spam, people thought anonymous email was a good idea

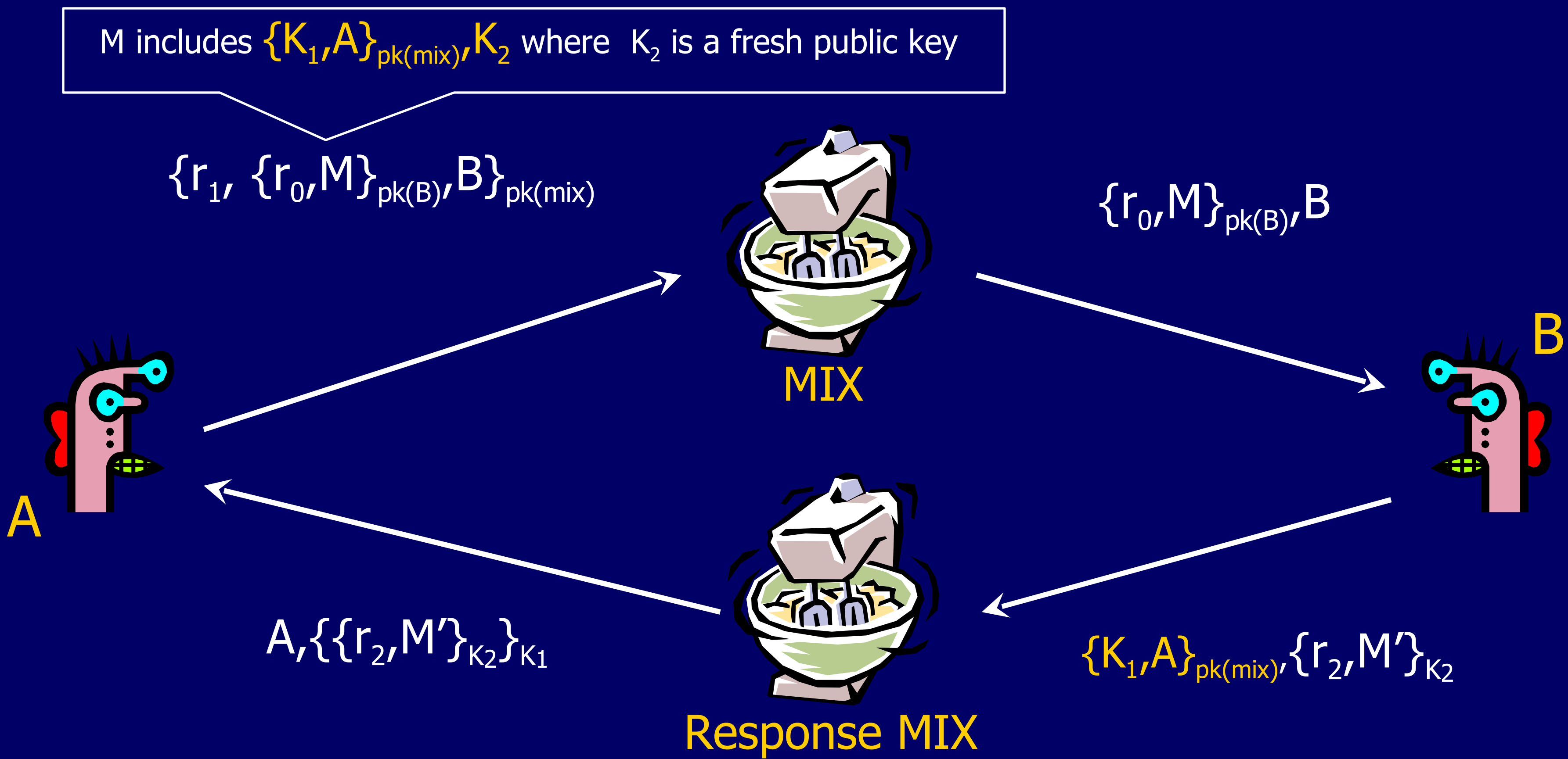
- Public key crypto + trusted re-mailer (MIX)
  - Untrusted communication medium
  - Public keys used as persistent pseudonyms
- Modern anonymity systems use MIX as the basic building block

Basic MIX Design



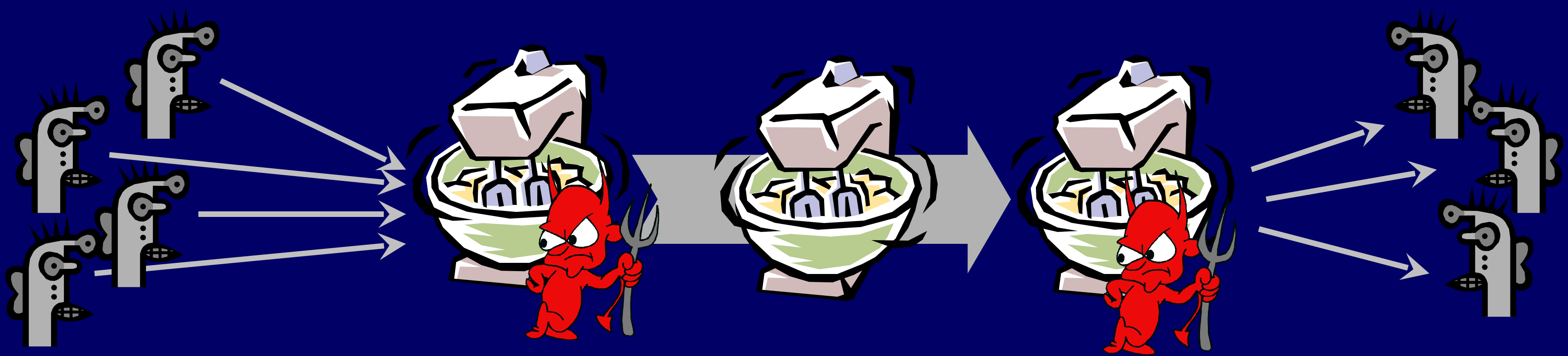
Adversary knows all senders and all receivers, but cannot link a sent message with a received message

Anonymous Return Addresses



Secrecy without authentication
(good for an online confession service)

MIX Cascade



- Messages are sent through a sequence of MIXes
- Some of the mixes may be controlled by adversary, but even a single good mix guarantees anonymity
- Need traffic padding and buffering to prevent timing correlation attacks

Dining Cryptographers

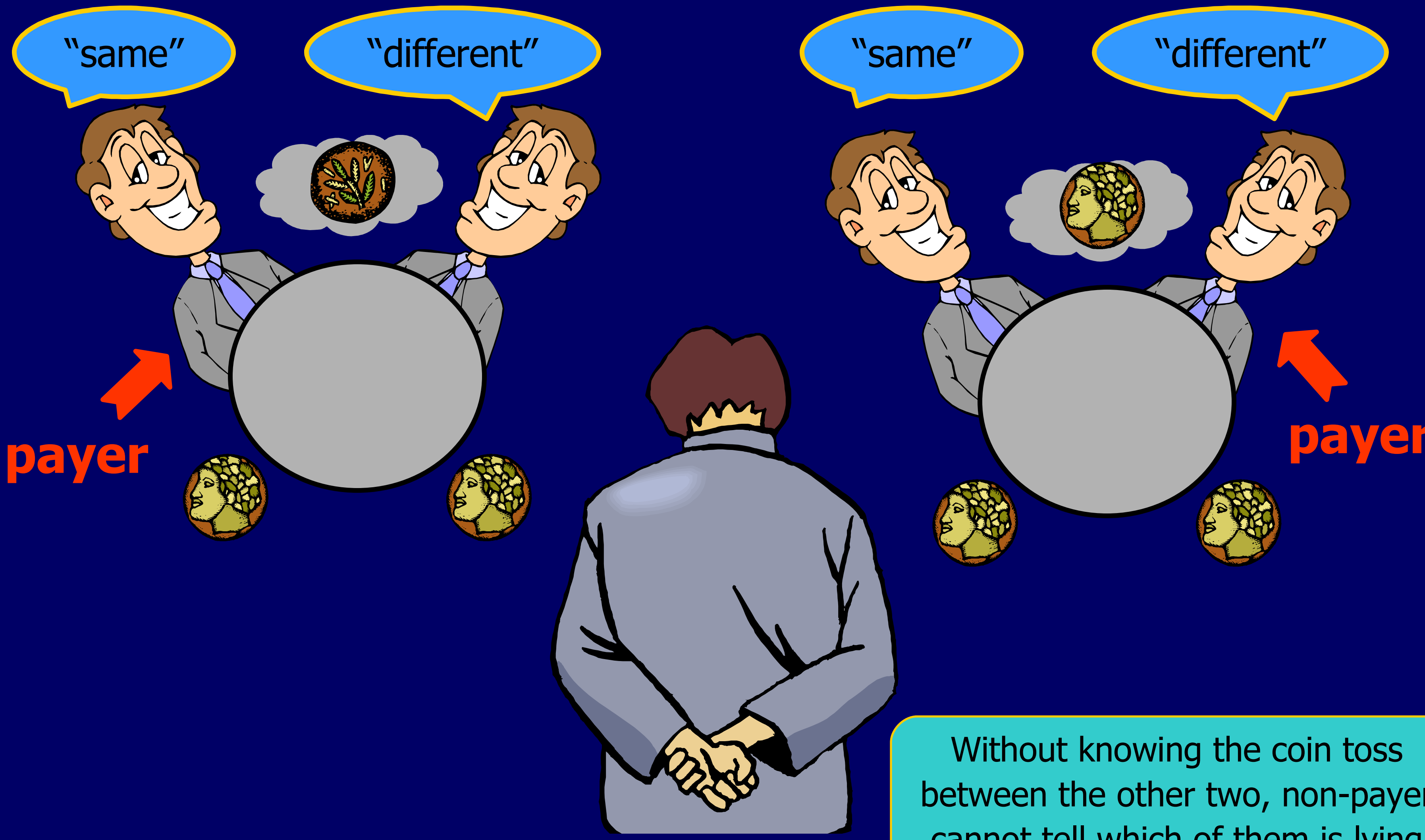
- Clever idea how to make a message public in a perfectly untraceable manner
  - David Chaum. "The dining cryptographers problem: unconditional sender and recipient untraceability." Journal of Cryptology, 1988.
- Guarantees information-theoretic anonymity for message senders
  - This is an unusually strong form of security: defeats adversary who has <u>unlimited</u> computational power
- Impractical, requires huge amount of randomness
  - In group of size N, need N random bits to send 1 bit

Three-Person DC Protocol

Three cryptographers are having dinner.
Either NSA is paying for the dinner, or
one of them is paying, but wishes to remain anonymous.

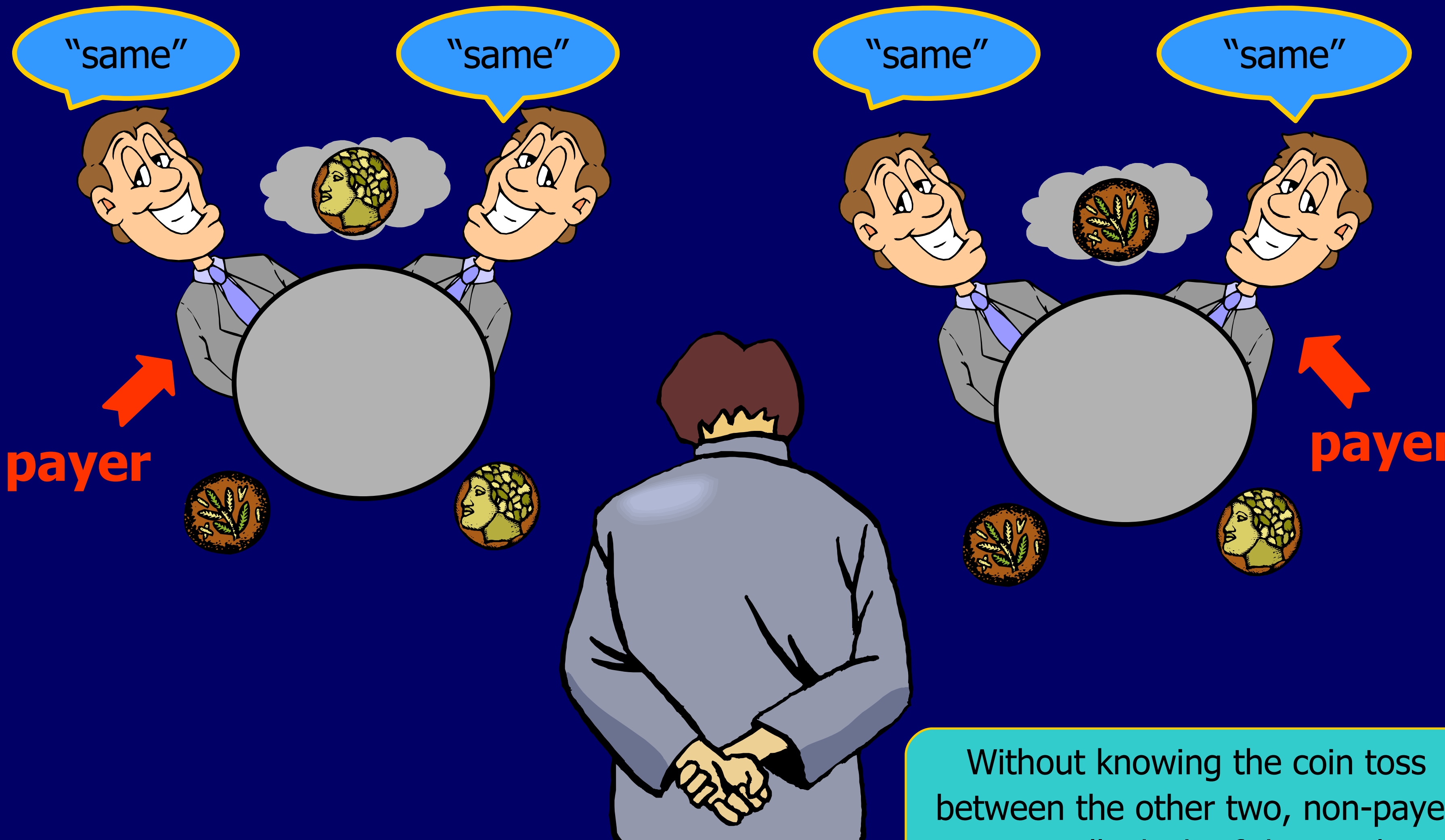
5. Each diner flips a coin and shows it to his left neighbor.
   - Every diner will see two coins: his own and his right neighbor's.
6. Each diner announces whether the two coins are the same. If he is the payer, he lies (says the opposite).
7. Odd number of "same" $\Rightarrow$ NSA is paying;
   even number of "same" $\Rightarrow$ one of them is paying
   - But a non-payer cannot tell which of the other two is paying!

Non-Payer's View: Same Coins



Without knowing the coin toss between the other two, non-payer cannot tell which of them is lying

Non-Payer’s View: Different Coins



Without knowing the coin toss between the other two, non-payer cannot tell which of them is lying

Superposed Sending

- This idea generalizes to any group of size N
- For each bit of the message, every user generates 1 random bit and sends it to 1 neighbor
  - Every user learns 2 bits (his own and his neighbor's)
- Each user announces (own bit XOR neighbor's bit)
- Sender announces (own bit XOR neighbor's bit XOR message bit)
- XOR of all announcements = message bit
  - Every randomly generated bit occurs in this sum twice (and is canceled by XOR), message bit occurs once

DC-Based Anonymity is Impractical

- Requires secure pairwise channels between group members
  - Otherwise, random bits cannot be shared
- Requires massive communication overhead and large amounts of randomness
- DC-net (a group of dining cryptographers) is robust even if some members cooperate
  - Guarantees perfect anonymity for the other members
- A great protocol to analyze
  - Difficult to reason about each member’s knowledge

What is Anonymity?

FBI intercepted three emails and learned that …

- Two of the emails came from the same account
- Emails are not in English
- The recipients are Bob386@hotmail.com, Dick Tracy and Osama Bin Laden, but it’s not known who received which email
- Emails were routed via Anonymizer.com

Wrong question: has “anonymity” been violated?

Right question: what does FBI actually know?

Definitions of Anonymity

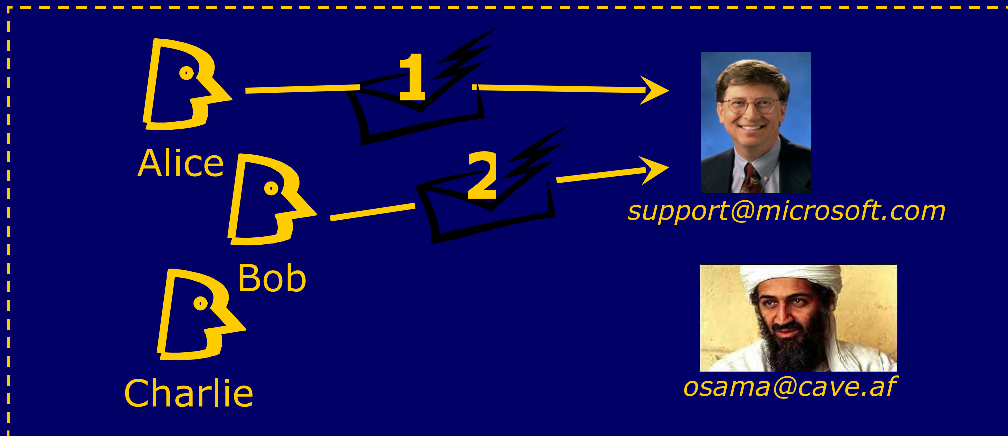
- “Anonymity is the state of being not identifiable within a set of subjects.”
  - There is no such thing as absolute anonymity
- Unlinkability of action and identity
  - E.g., sender and his email are no more related within the system than they are related in a-priori knowledge
- Unobservability
  - Any item of interest (message, event, action) is indistinguishable from any other item of interest
- “Anonymity is bullshit” - Joan Feigenbaum

Anonymity and Knowledge

- Anonymity deals with hiding information
  - User’s identity is hidden
  - Relationship between users is hidden
  - User cannot be identified within a set of suspects
- Natural way to express anonymity is to state what the adversary should not know
  - Good application for logic of knowledge
  - Not supported by conventional formalisms for security (process calculi, I/O automata, …)
- To determine whether anonymity holds, need some representation of knowledge

k-Anonymity

What actually happened



What adversary knows

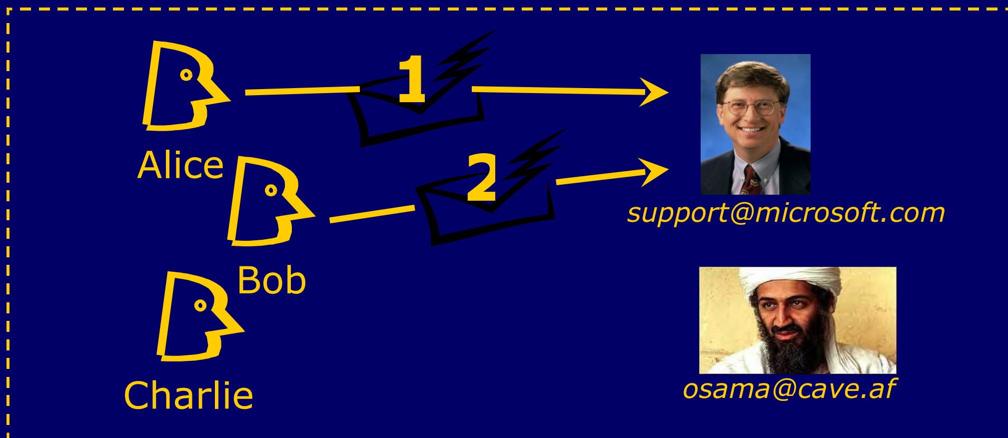
Sender suspects(1) = Alice or Charlie

Sender suspects(2) = Bob or Charlie

2-anonymity for senders:
2 plausible senders for each message

Absolute Anonymity

What actually happened



What attacker knows

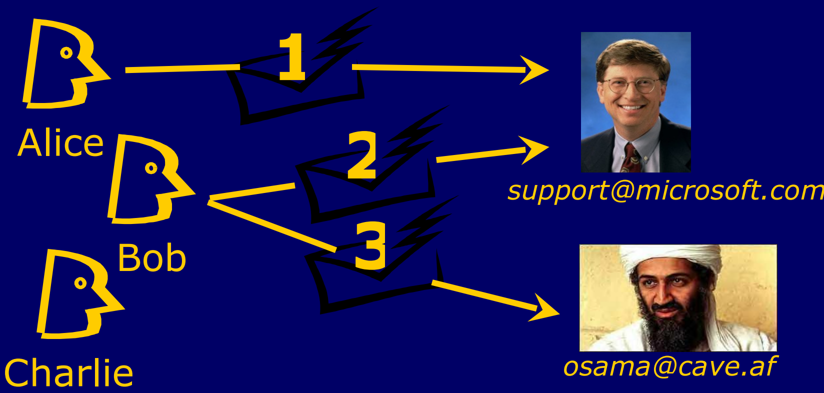
Sender suspects(1) = Alice, Bob or Charlie

Sender suspects(2) = Alice, Bob or Charlie

***absolute sender anonymity**:*
every agent is a plausible sender for every message

Identities Are Not Enough

What actually happened



What attacker knows

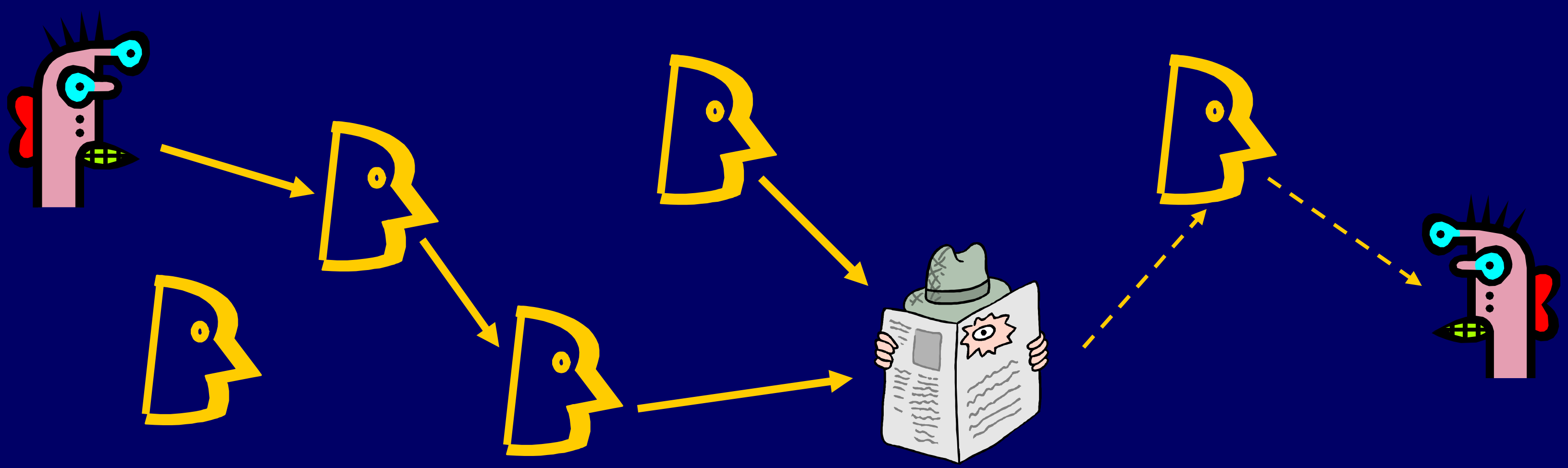
Sender suspects(1) = Alice, Bob or Charlie

Sender suspects(2) = Alice, Bob or Charlie

Sender(2) = Sender(3)

We need to be able to express this knowledge

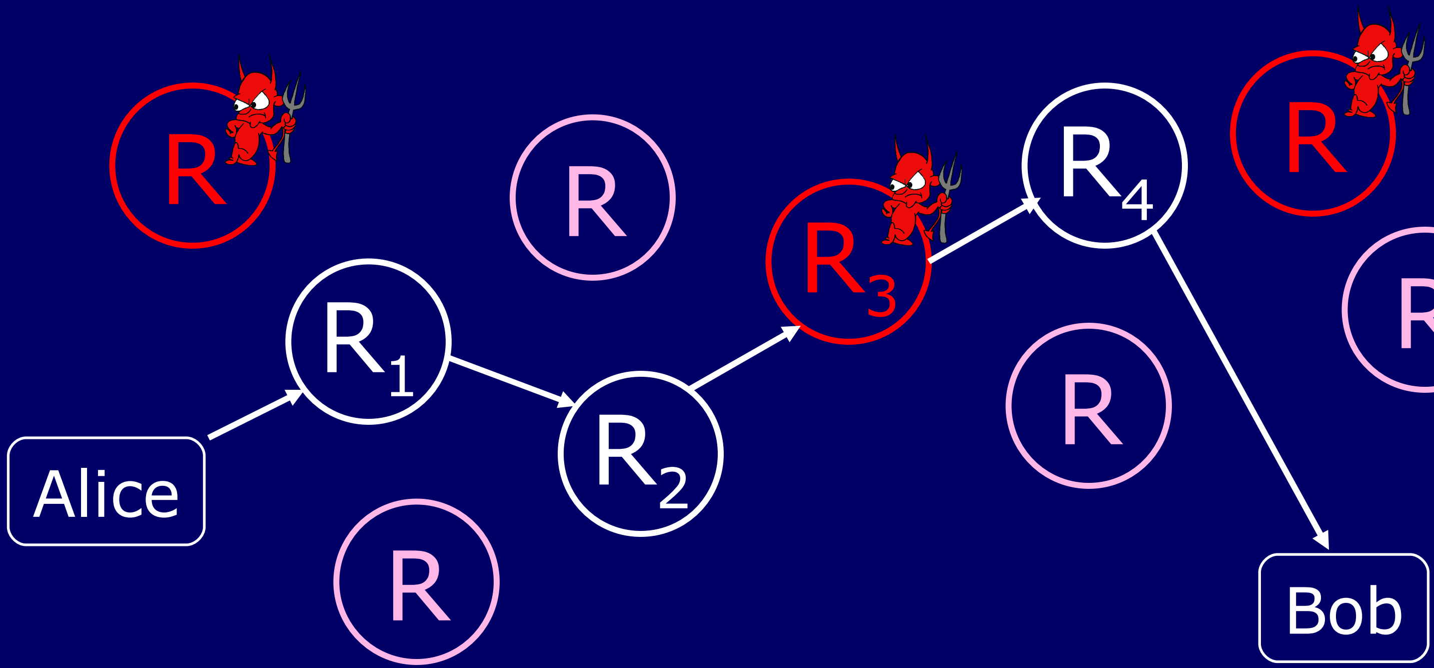
Anonymity via Random Routing



- Hide message source by routing it randomly
  - Popular technique: Crowds, Freenet, Onion Routing
- Routers don't know for sure if the apparent source of a message is the true sender or another router
  - Only secure against <u>local</u> attackers!

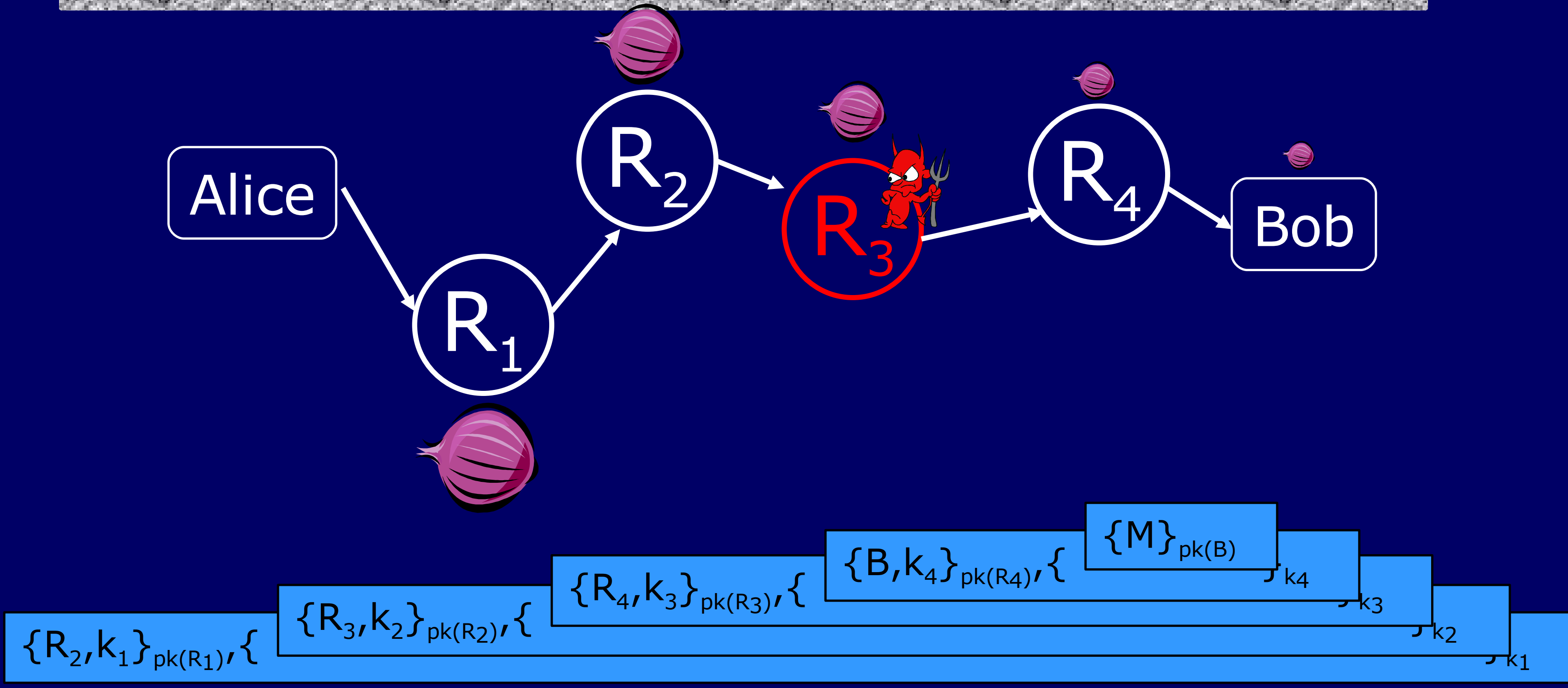
Onion Routing

[Reed, Syverson, Goldschlag '97]



- Sender chooses a random sequence of routers
  - Some routers are honest, some hostile
  - Sender controls the length of the path
  - Similar to a MIX cascade
- Goal: hostile routers shouldn't learn that Alice is talking to Bob

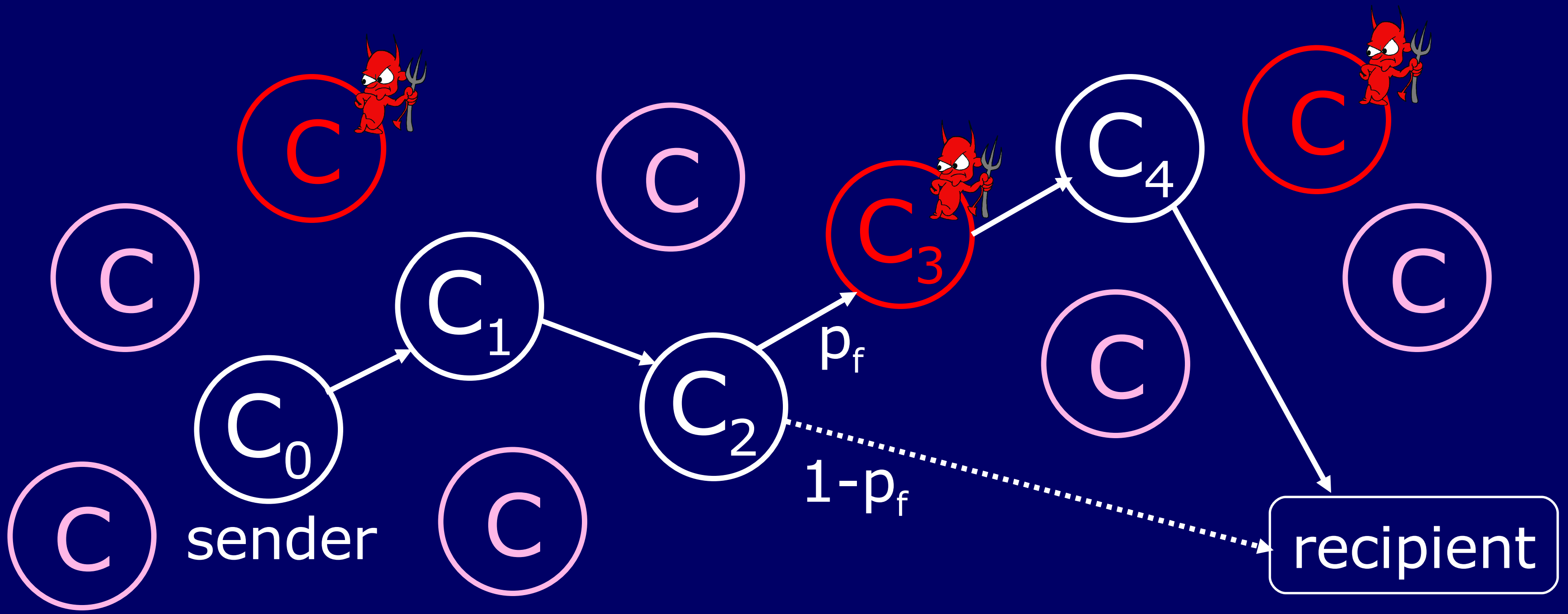
The Onion



- Routing info for each link encrypted with router's public key
- Each router learns only the identity of the next router

Crowds System

[Reiter,Rubin '98]



- Routers form a random path when establishing connection
  - In onion routing, random path is chosen in advance by sender
- After receiving a message, honest router flips a biased coin
  - With probability P_f randomly selects next router and forwards msg
  - With probability $1-P_f$ sends directly to the recipient

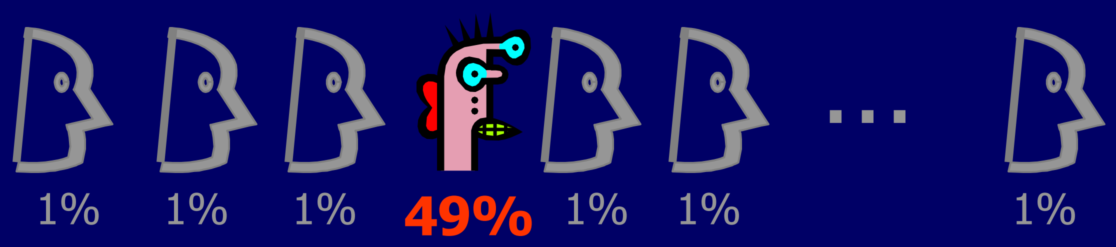
Probabilistic Notions of Anonymity

- Beyond suspicion
  - The observed source of the message is no more likely to be the true sender than anybody else
- Probable innocence
  - Probability that the observed source of the message is the true sender is less than 50%
- Possible innocence
  - Non-trivial probability that the observed source of the message is <u>not</u> the true sender

> Guaranteed by Crowds if there are sufficiently many honest routers: $N_{good}+N_{bad} \geq p_f/(p_f-0.5) \bullet (N_{bad}+1)$

A Couple of Issues

- Is probable innocence enough?



Maybe Ok for “plausible deniability”

- Multiple-paths vulnerability
  - Can attacker relate multiple paths from same sender?
    - E.g., browsing the same website at the same time of day
  - Each new path gives attacker a new observation
  - Can’t keep paths static since members join and leave

Anonymity Bibliography

- Free Haven project (anonymous distributed data storage) has an excellent anonymity bibliography
  - http://www.freehaven.net/anonbib/
- Many anonymity systems in various stages of deployment
  - Mixminion
    - http://www.mixminion.net
  - Mixmaster
    - http://mixmaster.sourceforge.net
  - Anonymizer
    - http://www.anonymizer.com
  - Zero-Knowledge Systems
    - http://www.zeroknowledge.com
- Cypherpunks
  - http://www.csua.berkeley.edu/cypherpunks/Home.html
  - Assorted rants on crypto-anarchy