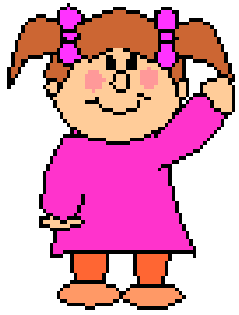
ENEE 459-C
Computer Security

Security protocols (continued)

UNIVERSITY OF
MARYLAND

Key Agreement: Diffie-Hellman Protocol

Key agreement protocol, both A and B contribute to the key
Setup: p prime and g generator of Z_p^*, p and g public.



$g^a \bmod p$

$g^b \bmod p$

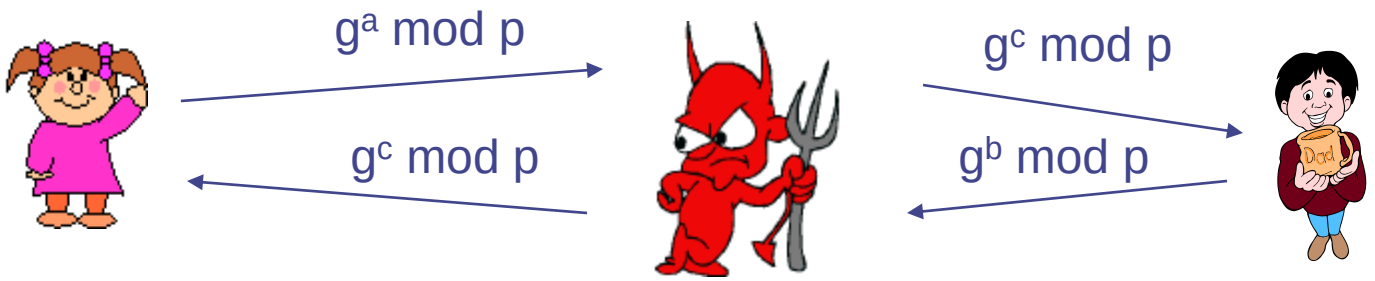
Pick random, secret a
Compute and send $g^a \bmod p$

$K = (g^b \bmod p)^a = g^{ab} \bmod p$

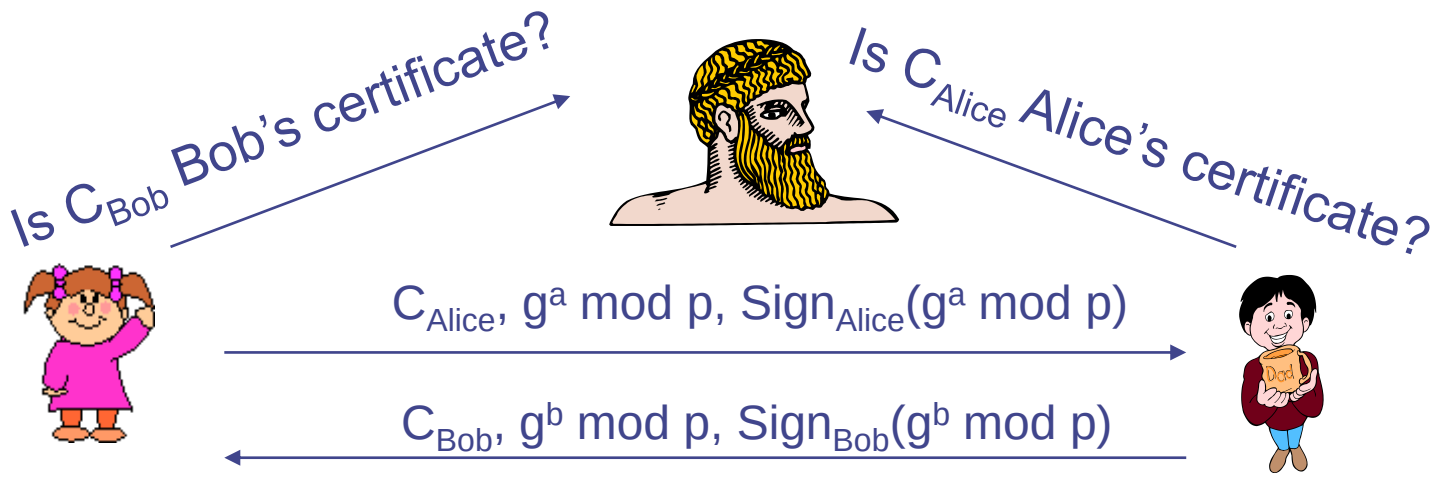
Pick random, secret b
Compute and send $g^b \bmod p$

$K = (g^a \bmod p)^b = g^{ab} \bmod p$

Authenticated Diffie-Hellman



Alice computes g^{ac} mod p and Bob computes g^{bc} mod p !!!



Zero Knowledge

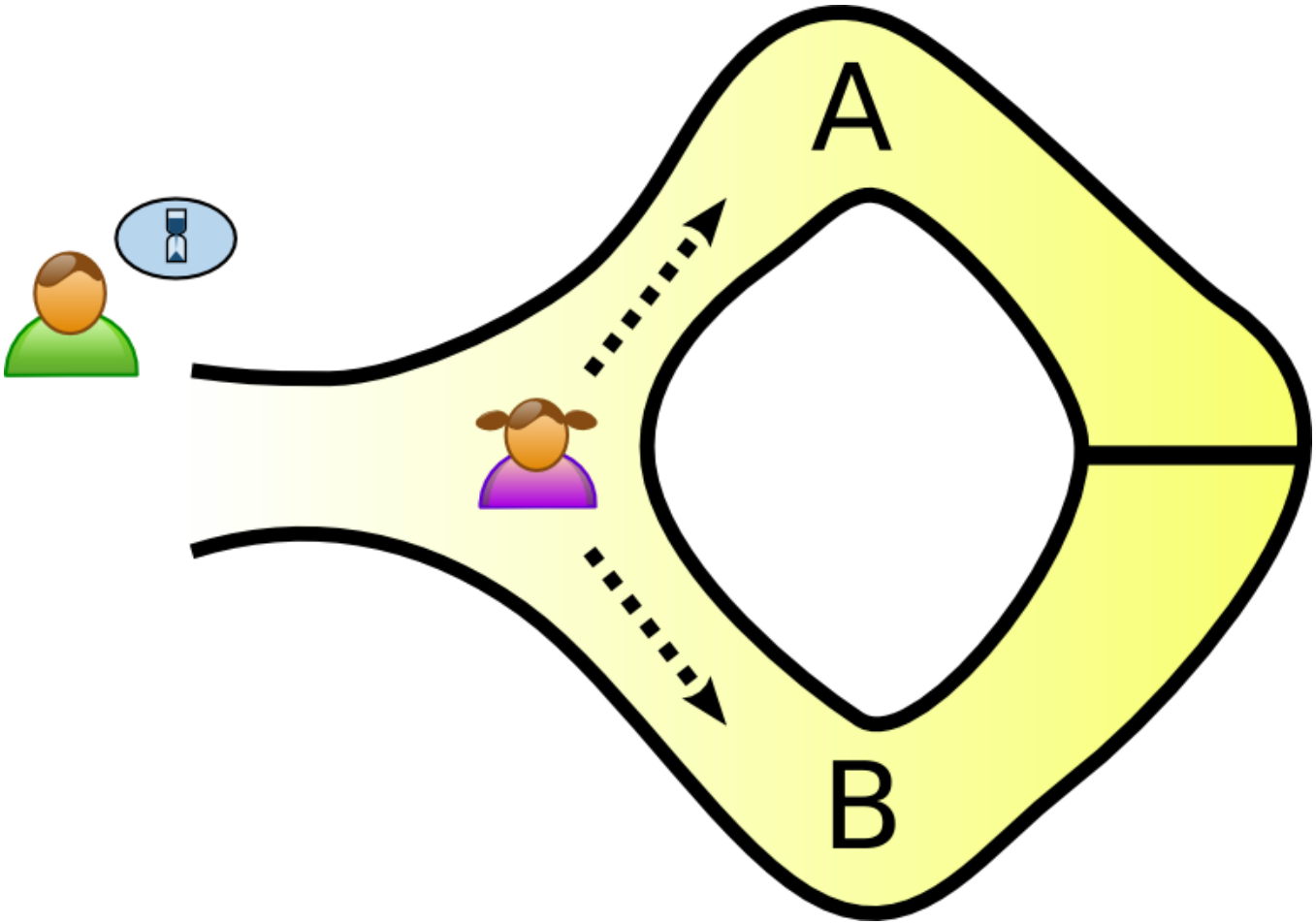
Proofs of Knowledge

- An interactive proof system involves a prover and a verifier
- The prover proves he knows something to the verifier
- Zero-knowledge proof of knowledge (ZKPK): The prover convinces verifier that he knows a secret without revealing the secret
  - Application: Log into a server without revealing your password

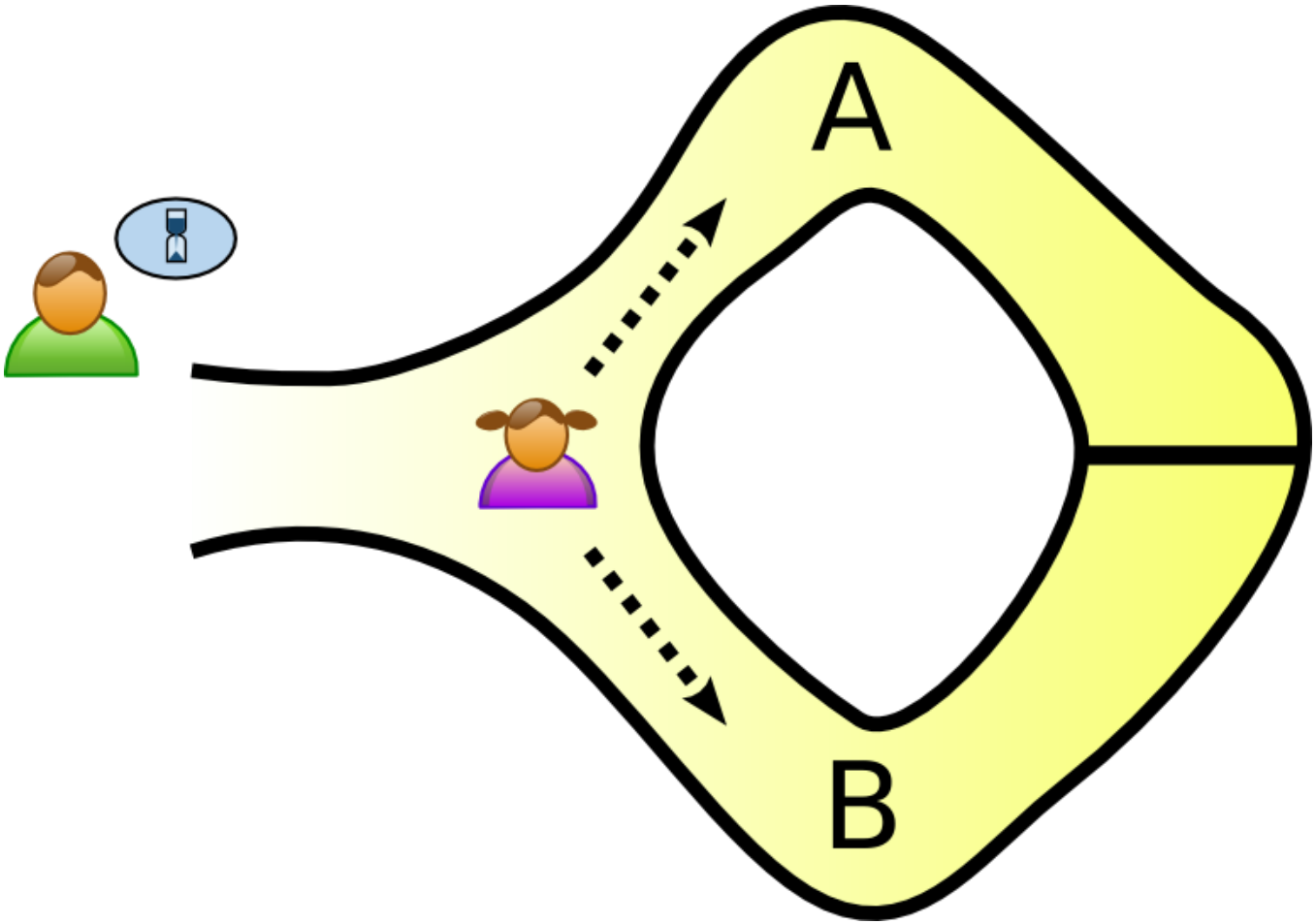
Properties of ZKPK

- Completeness
  - If both prover and verifier are honest, protocol succeeds with overwhelming probability
- Soundness
  - No one who does not know the secret can convince the verifier (except with very small probability)
    - Intuition: the protocol should not enable prover to prove a false statement
- Zero knowledge
  - The proof does not leak any information

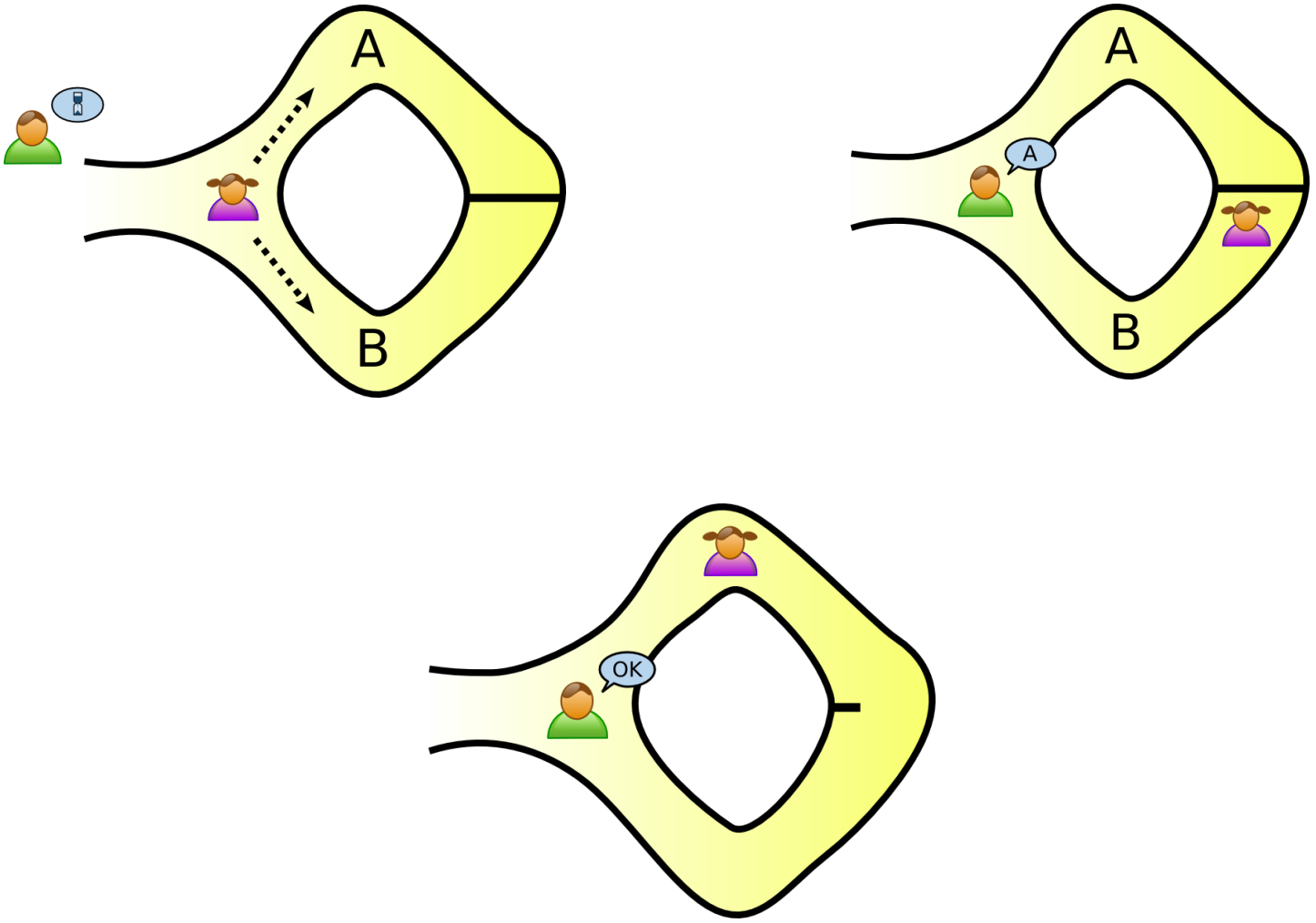
An example (wikipedia)



An example (wikipedia)

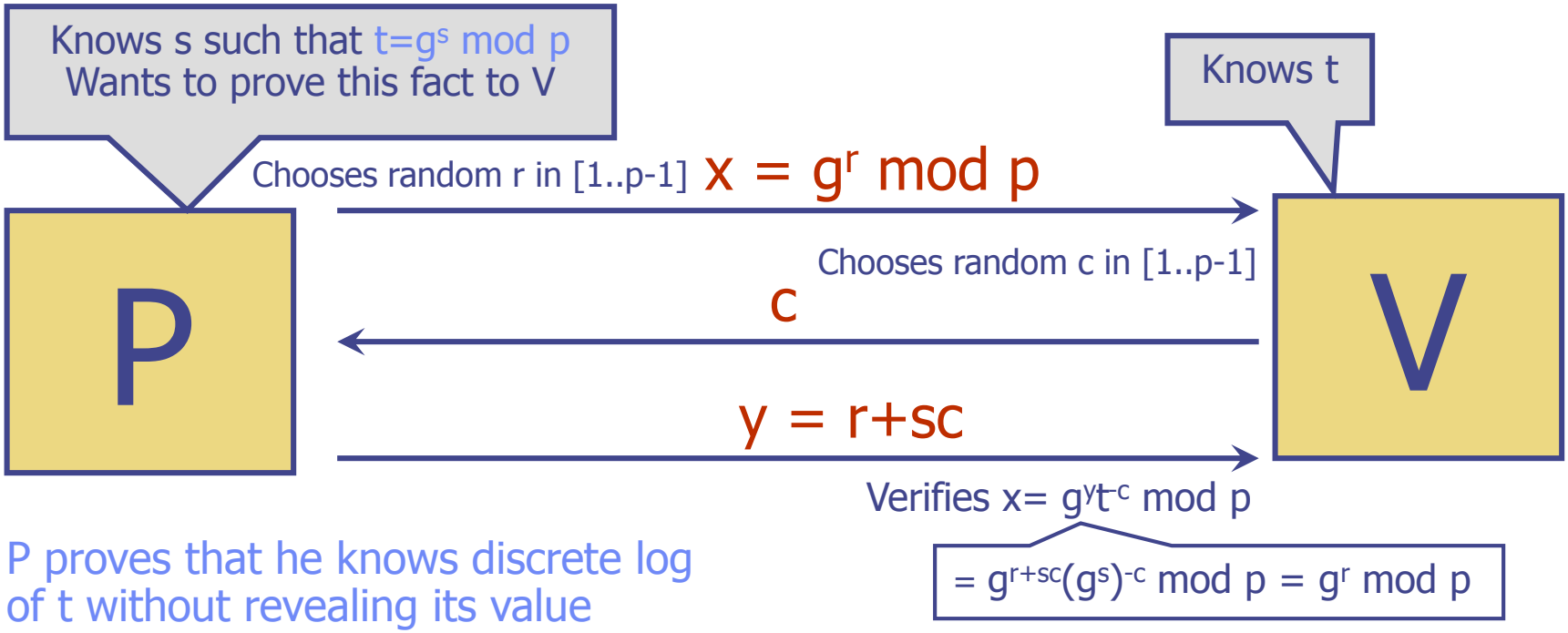


An example (wikipedia)



Schnorr's Id Protocol

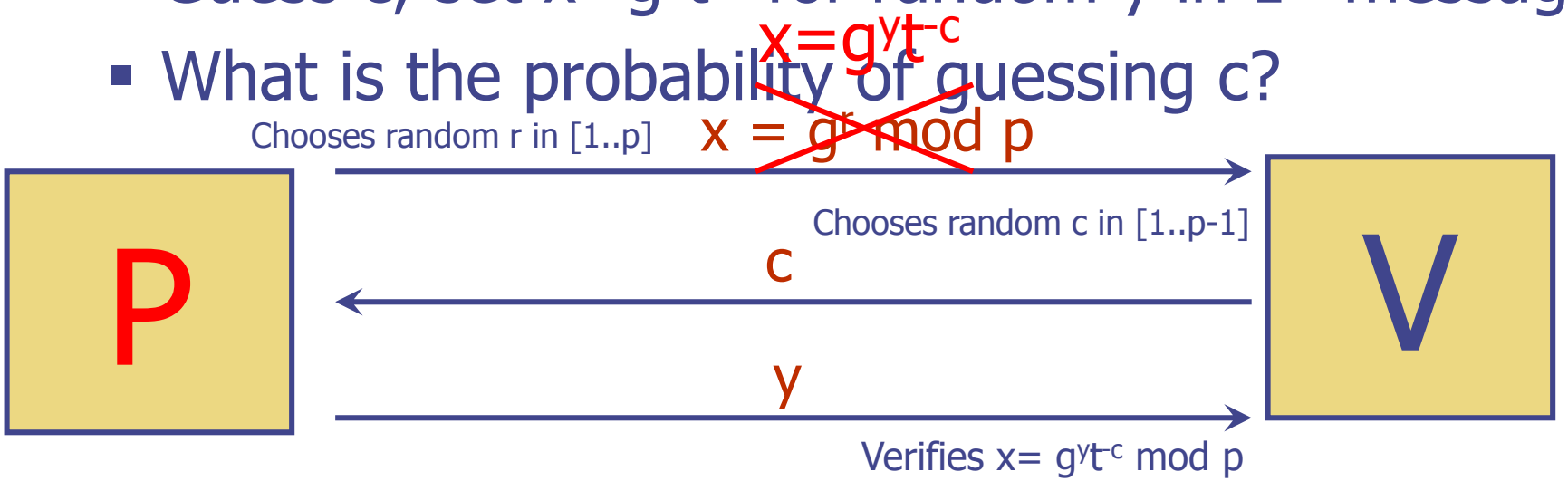
- System parameters
  - Prime p
  - g is a generator of Z_p^*



P proves that he knows discrete log of t without revealing its value

Cheating Prover

- Prover can cheat if he can guess c in advance
  - Guess c, set $x=g^y t^{-c}$ for random y in 1st message
  - What is the probability of guessing c?



P proves that he "knows" discrete log of t even though he does not know s

Anonymous Communication

Privacy on Public Networks

- Internet is designed as a public network
  - Machines on your LAN may see your traffic, network routers see all traffic that passes through them
- Routing information is public
  - IP packet headers identify source and destination
  - Even a passive observer can easily figure out who is talking to whom
- Encryption does not hide identities
  - Encryption hides payload, but not routing information

What is Anonymity?

- Anonymity is the state of being not identifiable within a set of subjects
  - Hide your activities among others' similar activities
- Unlinkability of communicating identities
  - For example, sender and receiver are no more related after observing communication than they were before (difficult to achieve)

Applications of Anonymity

- Privacy
  - Hide online transactions, Web browsing, etc. from intrusive governments, marketers and archivists
- Anonymous electronic voting
- Censorship-resistant publishing

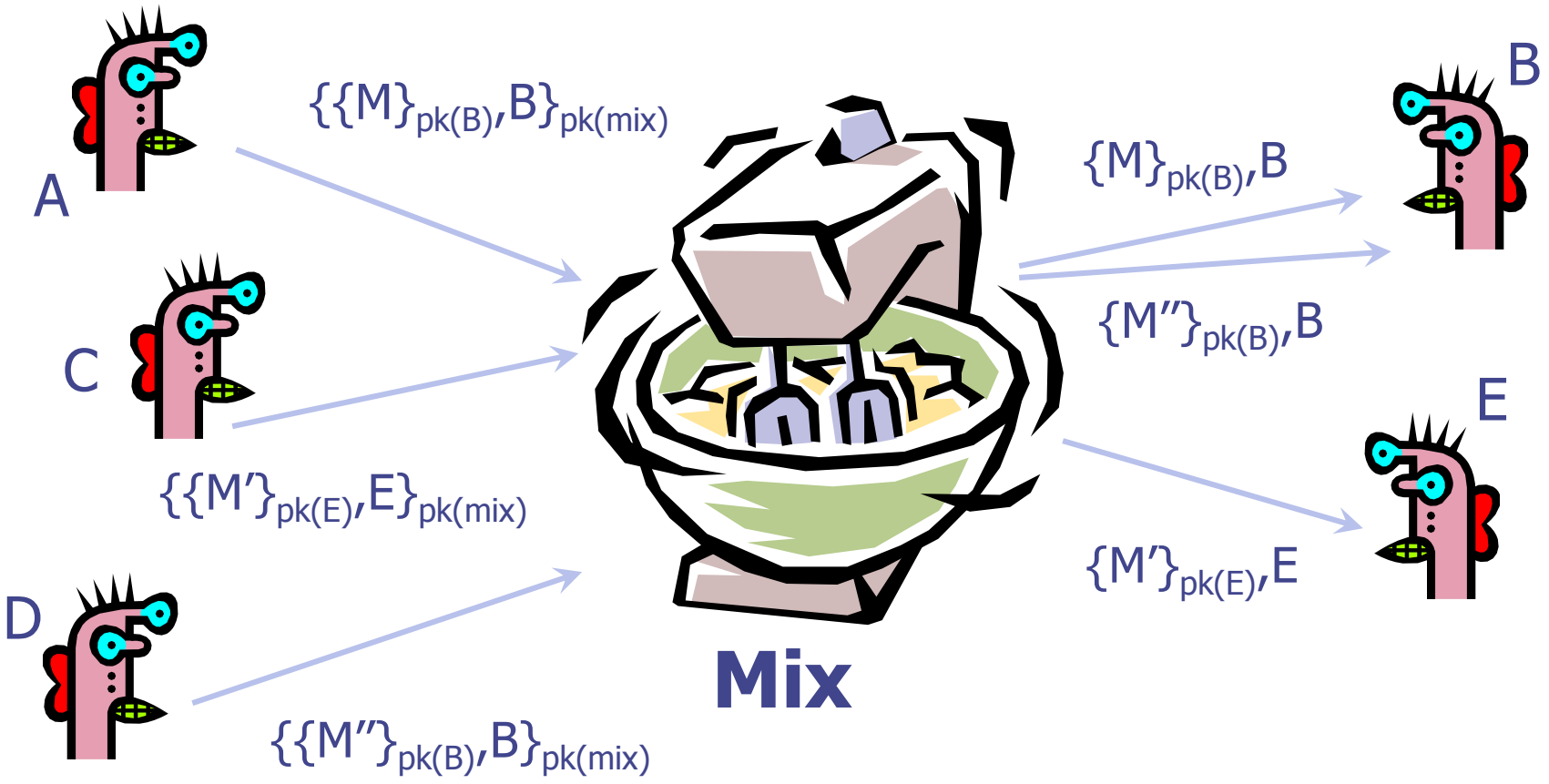
Attacks on Anonymity

- Passive traffic analysis
  - Infer from network traffic who is talking to whom
- Compromise of network nodes
  - Attacker may compromise some routers
  - It is not obvious which nodes have been compromised
    - Attacker may be passively logging traffic
  - Better not to trust any individual router
    - Assume that some fraction of routers is good, don't know which

Chaum's Mix

- Early proposal for anonymous email
  - David Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms". Communications of the ACM, February 1981.
- Public key crypto + trusted re-mailer (Mix)
- Modern anonymity systems use Mix as the basic building block

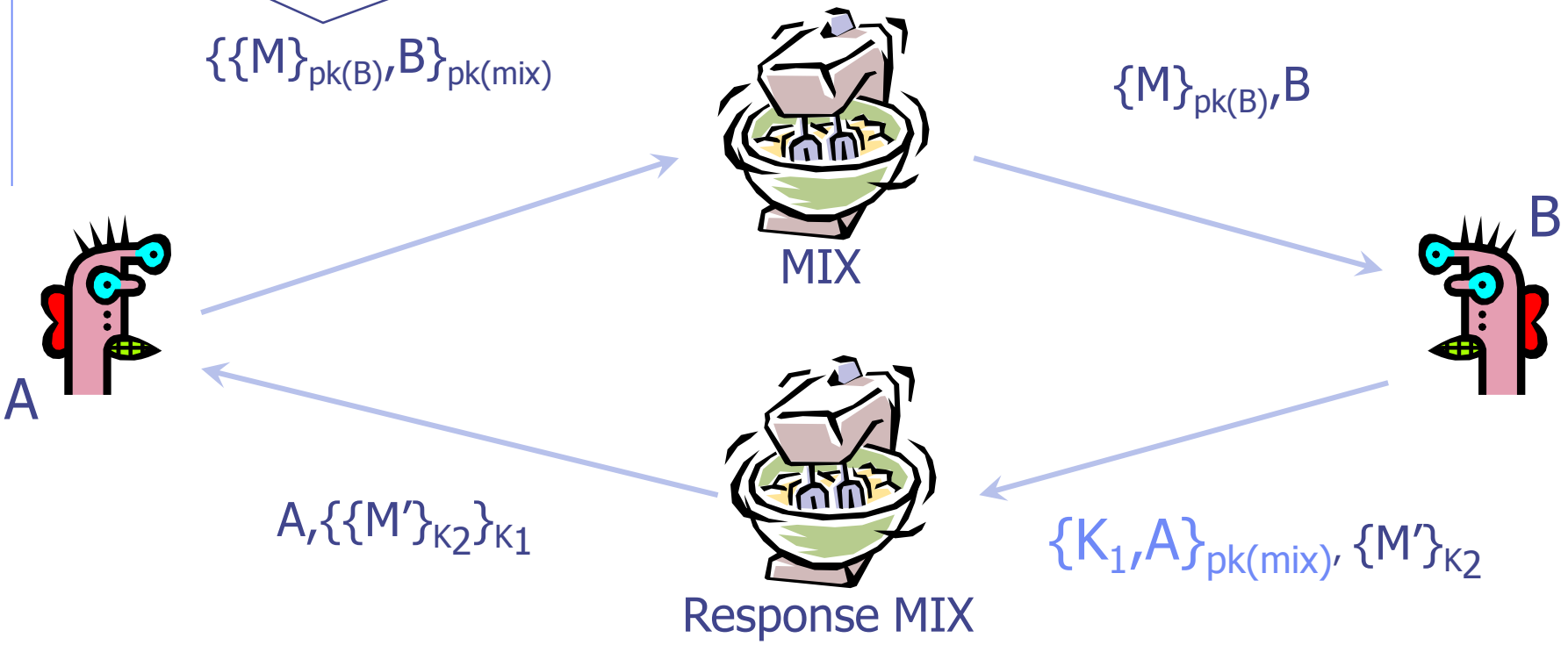
Basic Mix Design



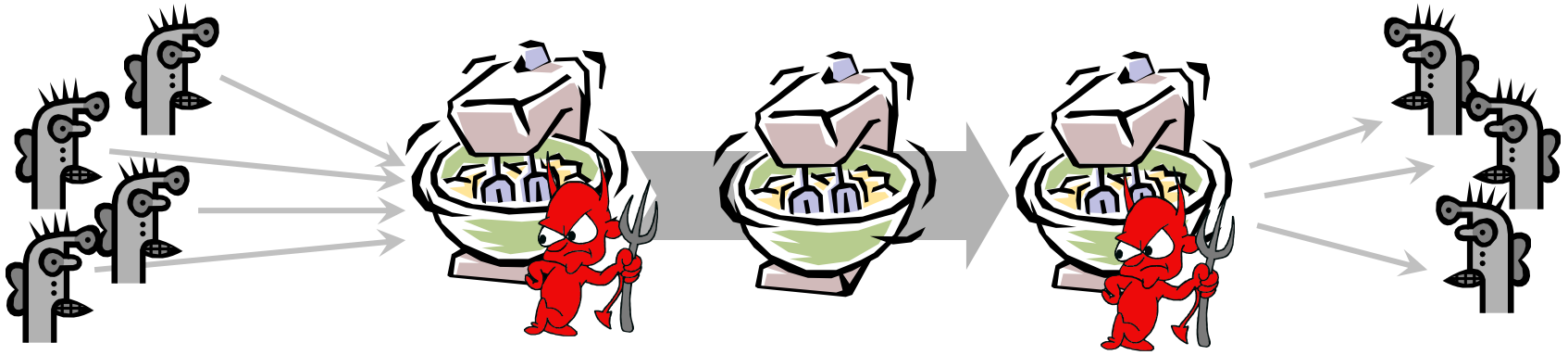
Anonymous Return Addresses

M includes $\{K_1,A\}_{pk(mix)}$, K_2 where K_2 is a fresh

public key

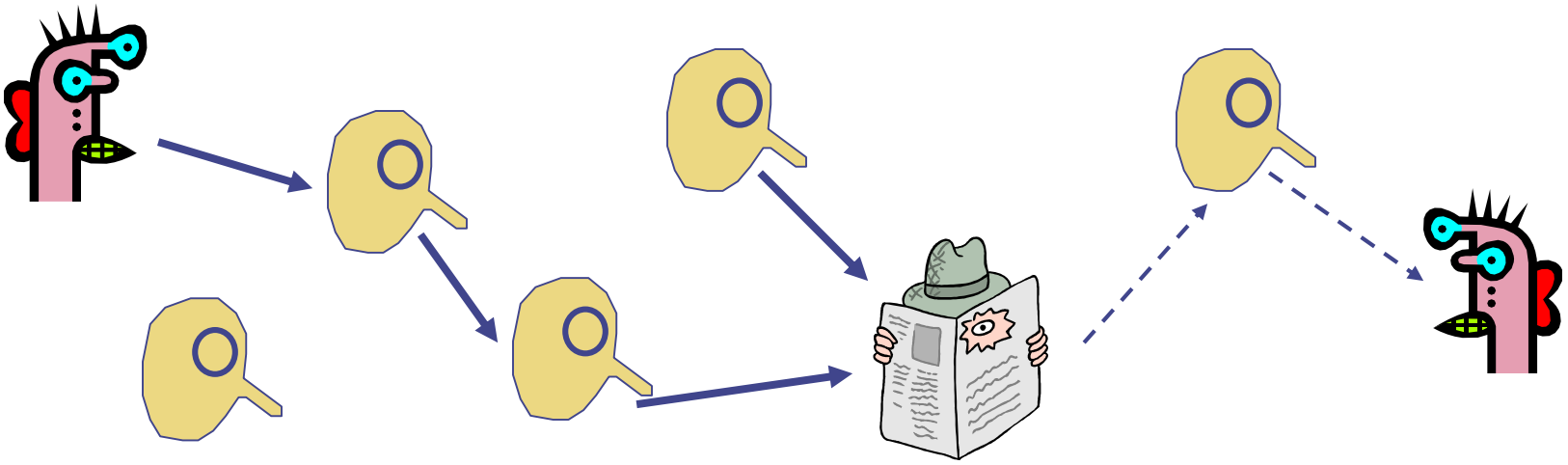


Use many mixers



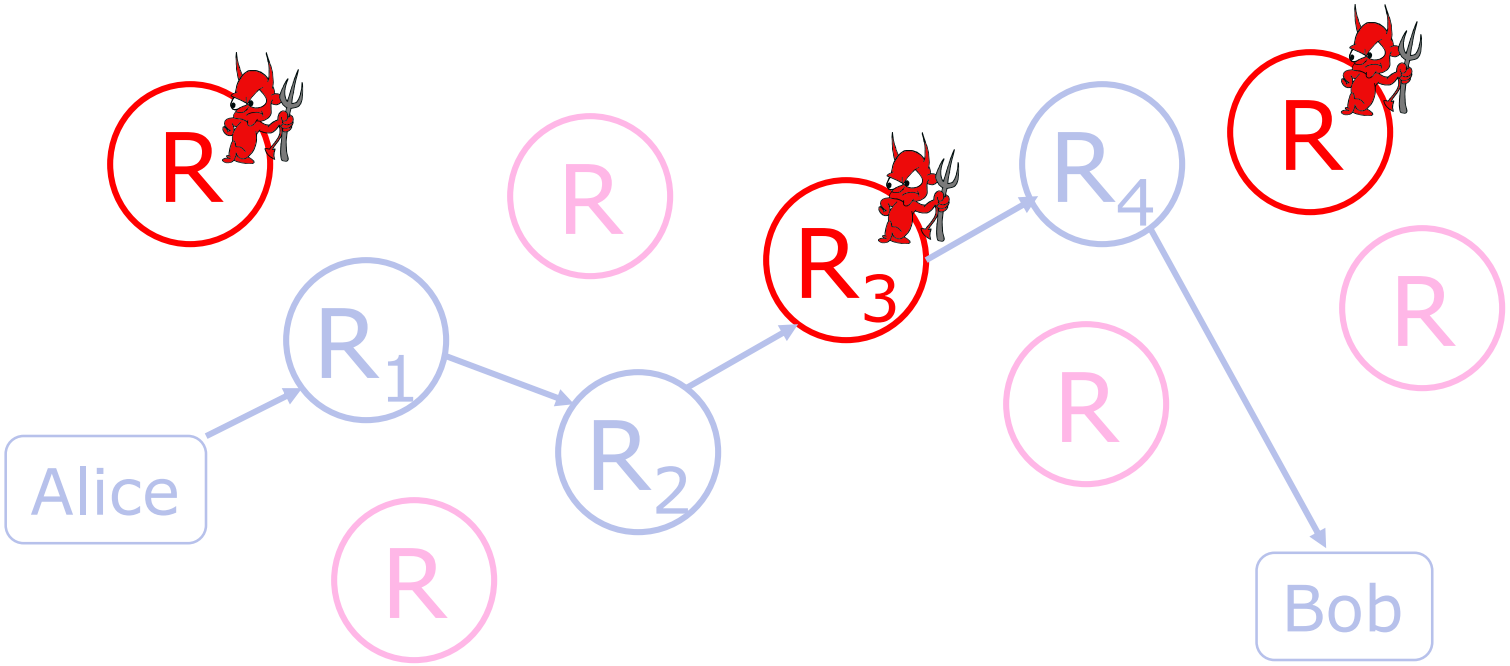
- Messages are sent through a sequence of mixes
  - Can also form an arbitrary network of mixes ("mixnet")
- Some of the mixes may be controlled by attacker, but even a single good mix guarantees anonymity

Idea: Randomized Routing



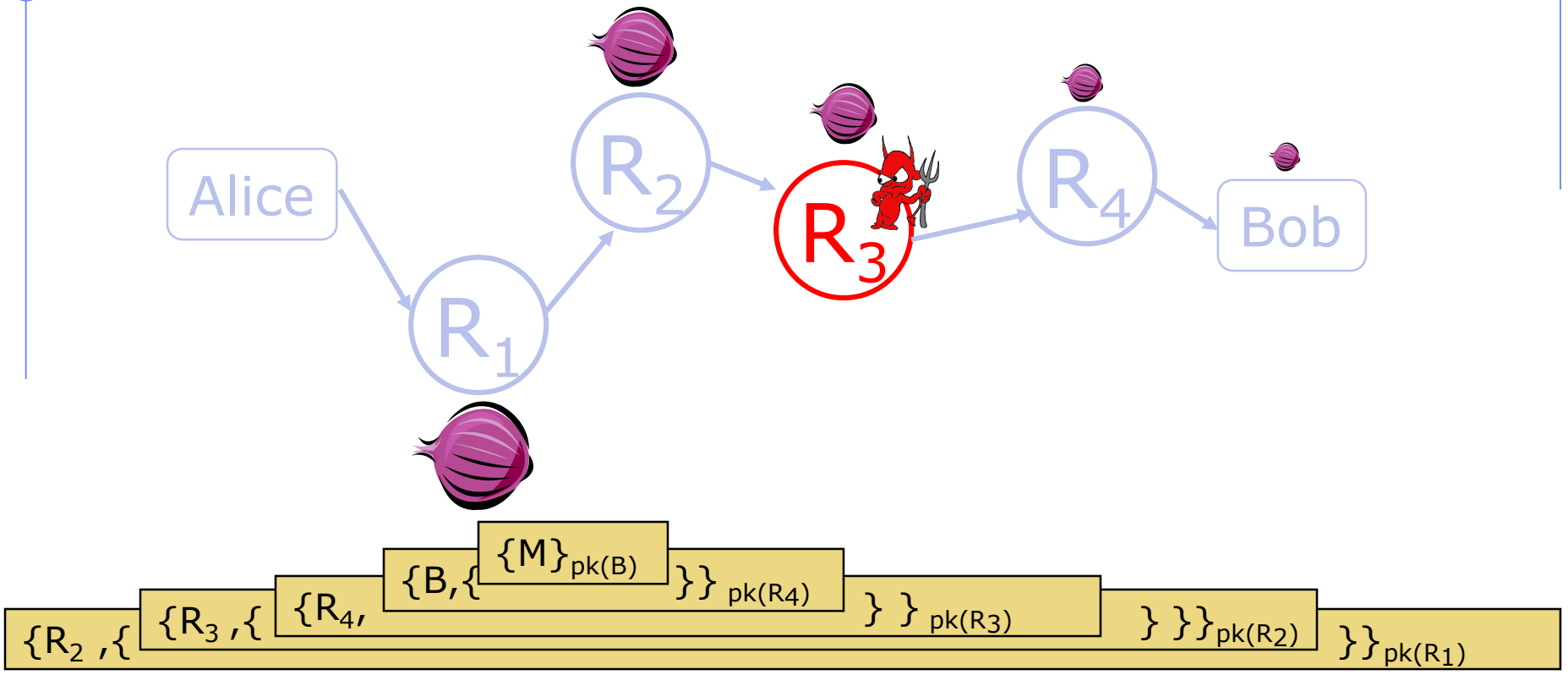
- Hide message source by routing it randomly
  - Popular technique: Crowds, Freenet, Onion routing
- Routers don't know for sure if the apparent source of a message is the true sender or not

Onion Routing [Reed, Syverson, Goldschlag '97]



- Sender chooses a random sequence of routers
  - Some routers are honest, some controlled by attacker
  - Sender controls the length of the path

Route Establishment



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

Disadvantages of Basic Mixnets/Onion Routing

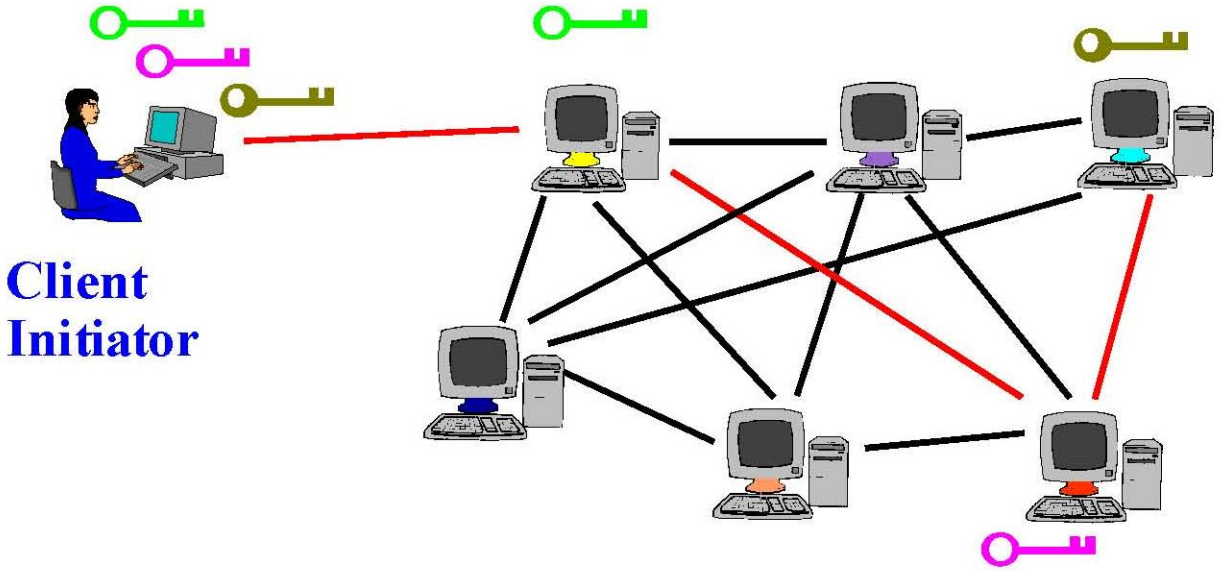
- Public-key encryption and decryption at each mix/router are computationally expensive
- Basic mixnets have high latency
  - Ok for email, not Ok for anonymous Web browsing
- Challenge: low-latency anonymity network

Tor

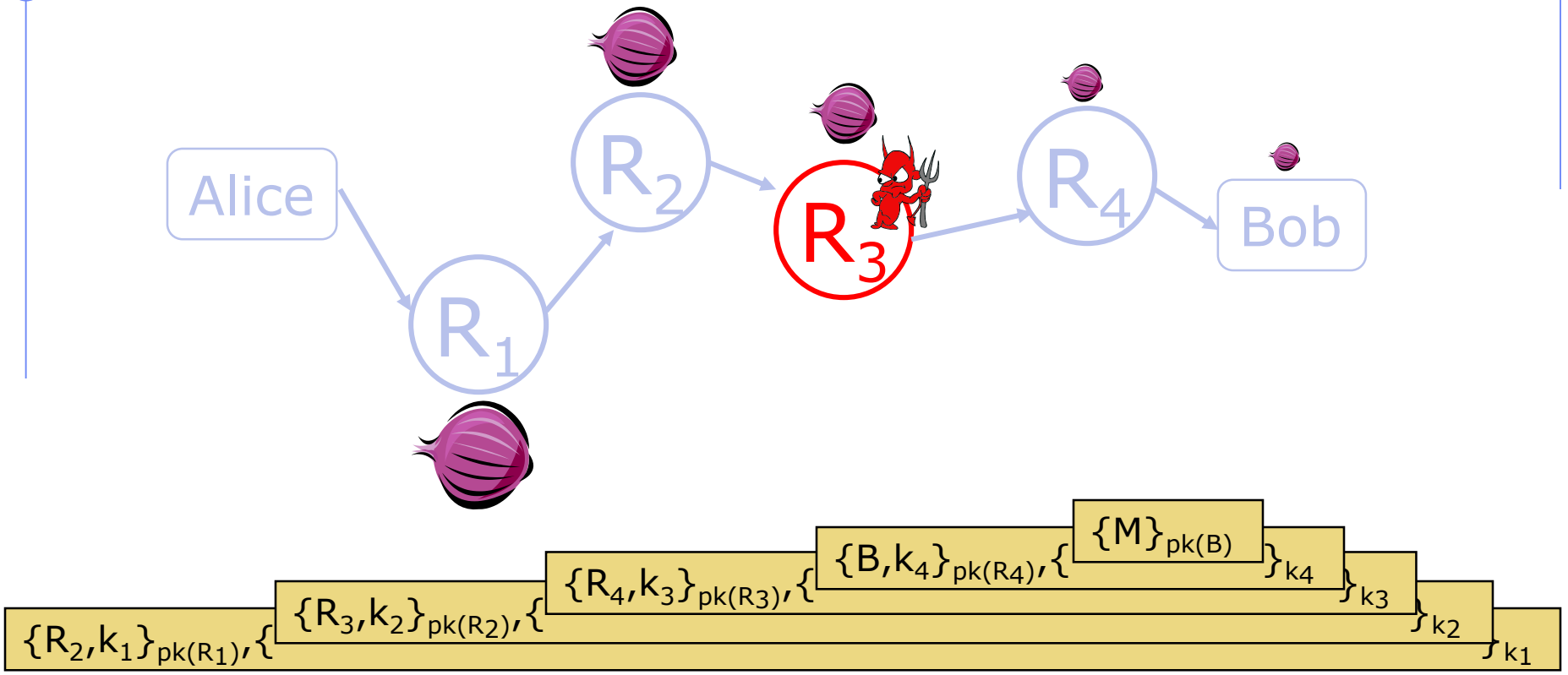
- Second-generation onion routing network
  - https://www.torproject.org/
  - Developed by Roger Dingledine, Nick Mathewson and Paul Syverson
  - Specifically designed for low-latency anonymous Internet communications
- Running since October 2003
- Thousands of users
- "Easy-to-use" client proxy
  - Freely available, can use it for anonymous browsing

Tor Circuit Setup

- Client proxy establishes symmetric session keys with onion routers



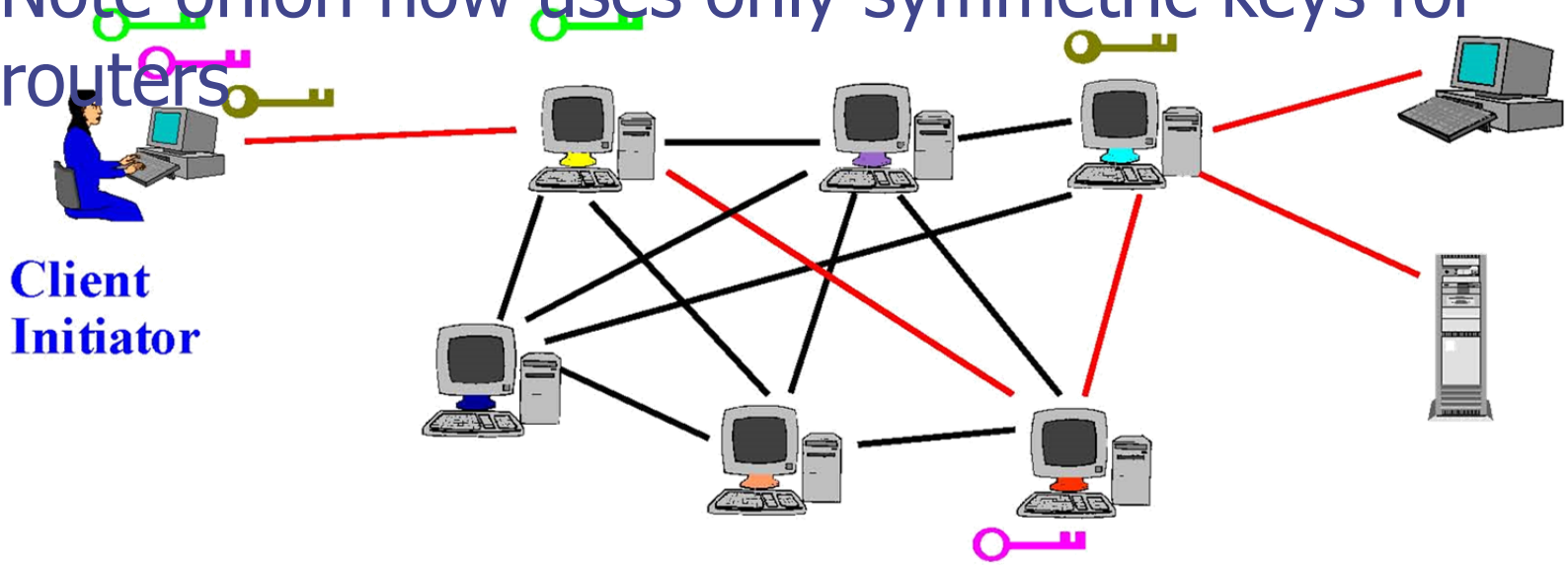
Tor Circuit Setup (details)



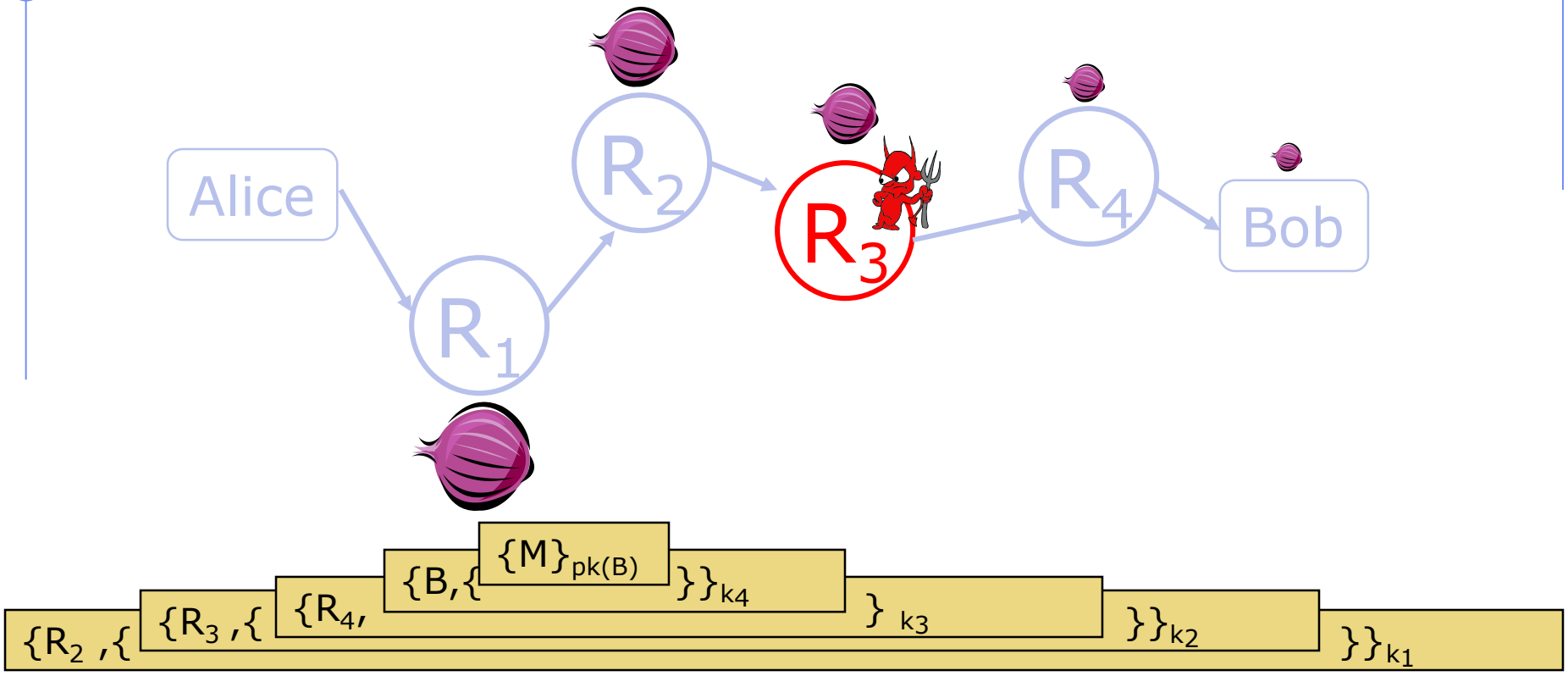
- Routing info for each link encrypted with router's public key
- Each router learns only the identity of the next router *and symmetric key with source*

Using a Tor Circuit

- Client applications connect and communicate over the established Tor circuit
  - Note onion now uses only symmetric keys for routers



Using a Tor Circuit(details)



Note onion now uses only symmetric keys for routers

Tor Management Issues

- Many applications can share one circuit
  - Multiple TCP streams over one anonymous connection
- Tor router doesn't need special privileges
  - Encourages people to set up their own routers
  - More participants = better anonymity for everyone
- Directory servers
  - Maintain lists of active onion routers, their locations, current public keys, etc.
  - Control how new routers join the network
    - "Sybil attack": attacker creates a large number of routers
  - Directory servers' keys ship with Tor code

Deployed Anonymity Systems

- Tor (http://tor.eff.org)
  - Overlay circuit-based anonymity network
  - Best for low-latency applications such as anonymous Web browsing
- Mixminion (http://www.mixminion.net)
  - Network of mixes
  - Best for high-latency applications such as anonymous email