Tor Hidden Services

Roger Dingledine
Free Haven Project
Electronic Frontier Foundation

http://tor.eff.org/

31 July 2005

Talk Outline

- Tor overview
- Circuit-building in Tor
- Hidden services in Tor
- Demo
- Anonymity issues with hidden services

Who Needs Hidden Services?

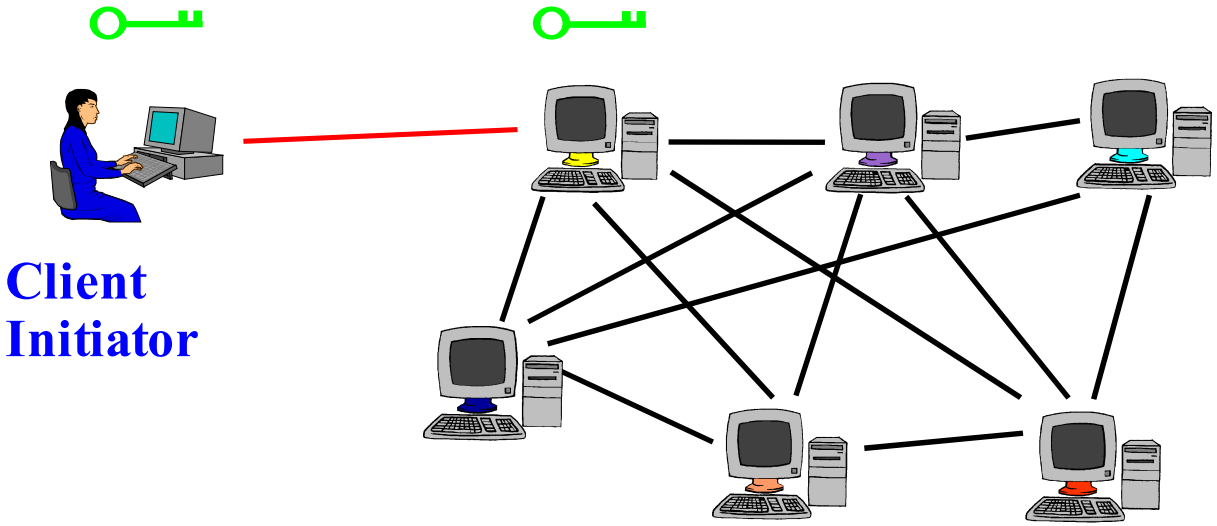
- Journalists, Dissidents, Whistleblowers (Indymedia, bloggers, Iran, Tibet)
- Censorship resistant publishers
- People who don't have public IP addresses
- Corporations
  - Google wants to test out new services without saying it's from Google
- Governments
  - Public announcement servers that can't be taken down by attackers

Tor overview

- 250 servers around the world. 50.000 users?
- Funded by EFF and United States Navy.
- Picked as the anonymizing layer for the EU PRIME project.
- Listed by PC World Magazine in the top 100 products of 2005.

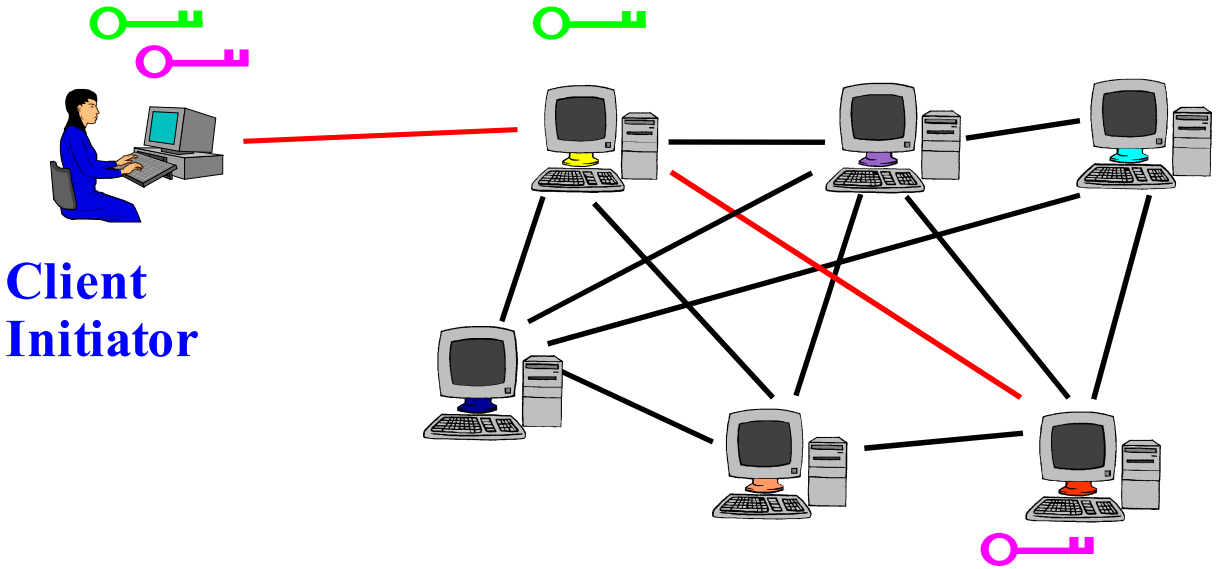
Tor Circuit Setup

- Client Proxy establishes session key + circuit w/ Onion Router 1



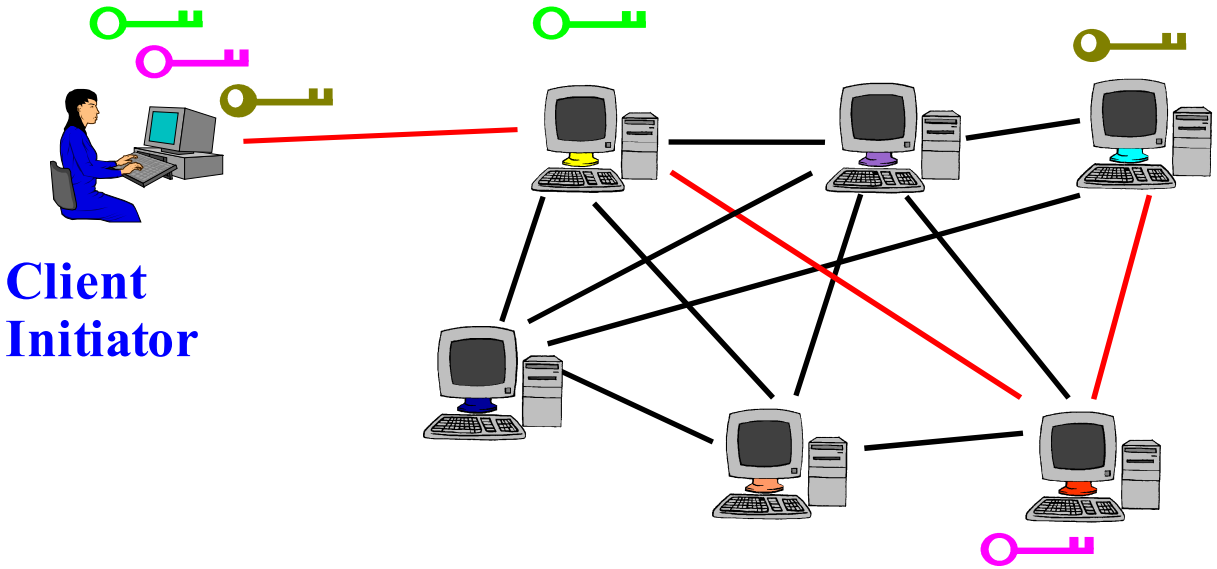
Tor Circuit Setup

- Client Proxy establishes session key + circuit w/ Onion Router 1
- Proxy tunnels through that circuit to extend to Onion Router 2



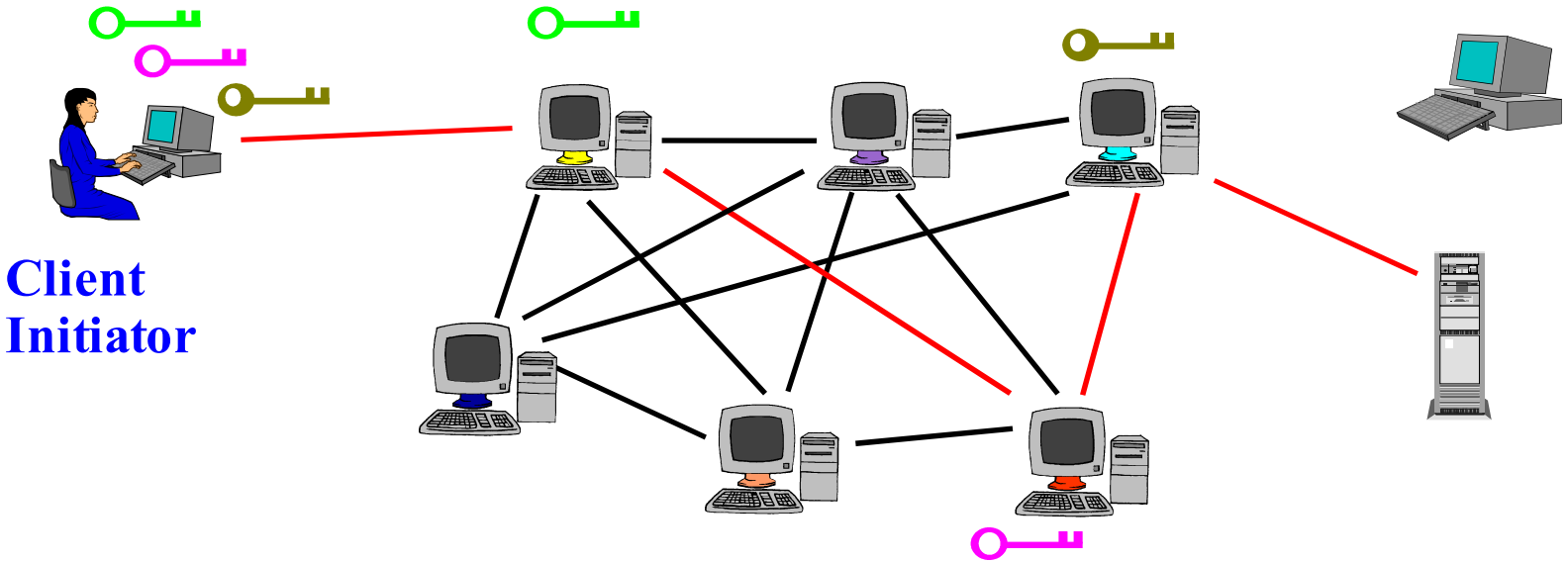
Tor Circuit Setup

- Client Proxy establishes session key + circuit w/ Onion Router 1
- Proxy tunnels through that circuit to extend to Onion Router 2
- Etc



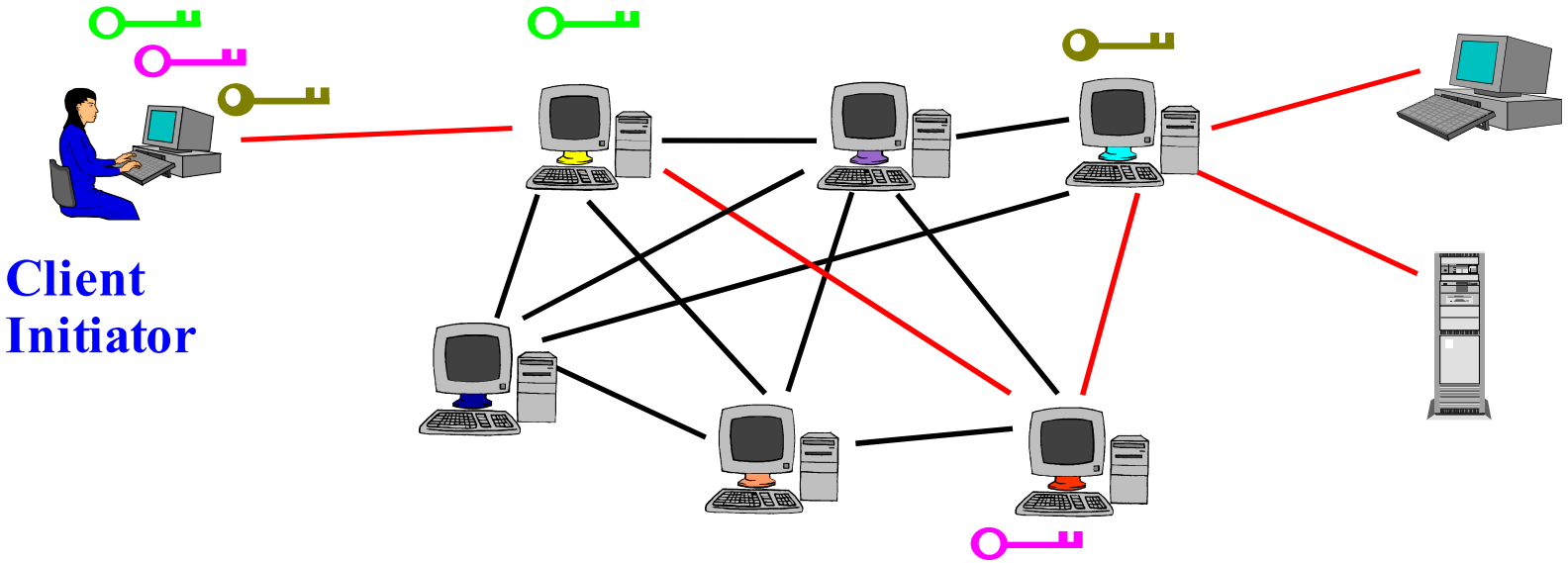
Tor Circuit Usage

- Client Proxy establishes session key + circuit w/ Onion Router 1
- Proxy tunnels through that circuit to extend to Onion Router 2
- Etc
- Client applications connect and communicate over Tor circuit



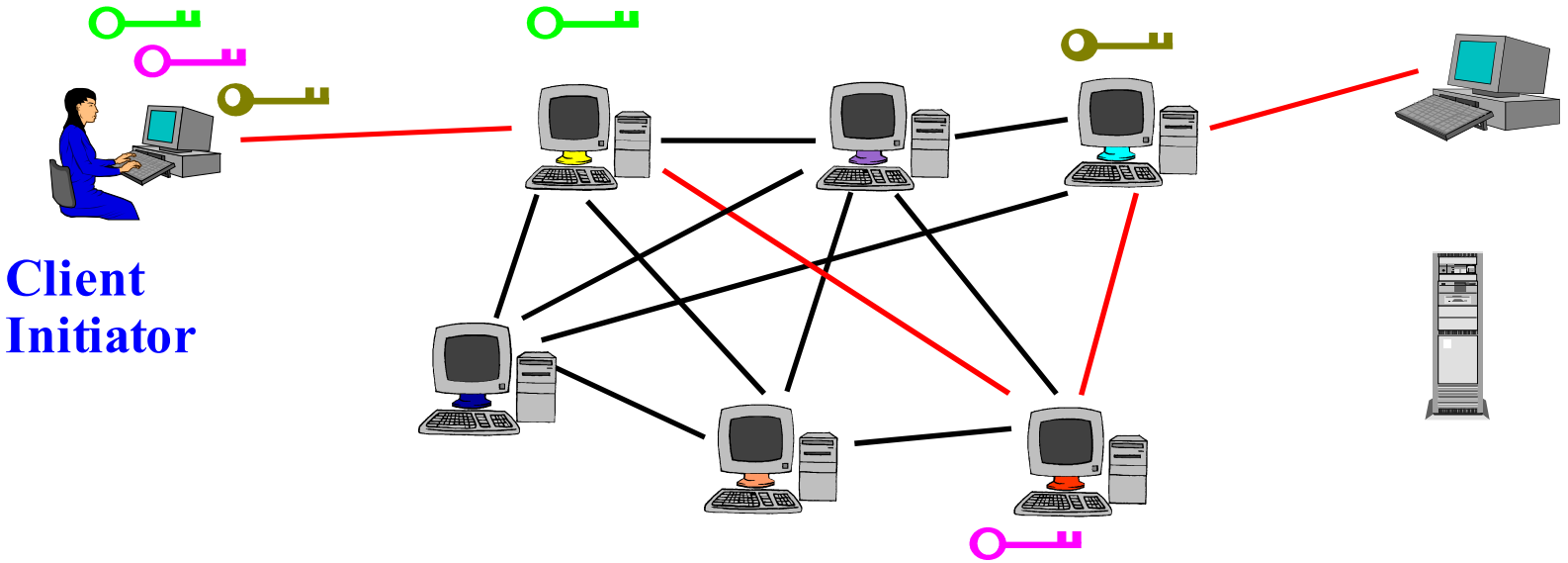
Tor Circuit Usage

- Client Proxy establishes session key + circuit w/ Onion Router 1
- Proxy tunnels through that circuit to extend to Onion Router 2
- Etc
- Client applications connect and communicate over Tor circuit



Tor Circuit Usage

- Client Proxy establishes session key + circuit w/ Onion Router 1
- Proxy tunnels through that circuit to extend to Onion Router 2
- Etc
- Client applications connect and communicate over Tor circuit



Where do I go to connect to the network?

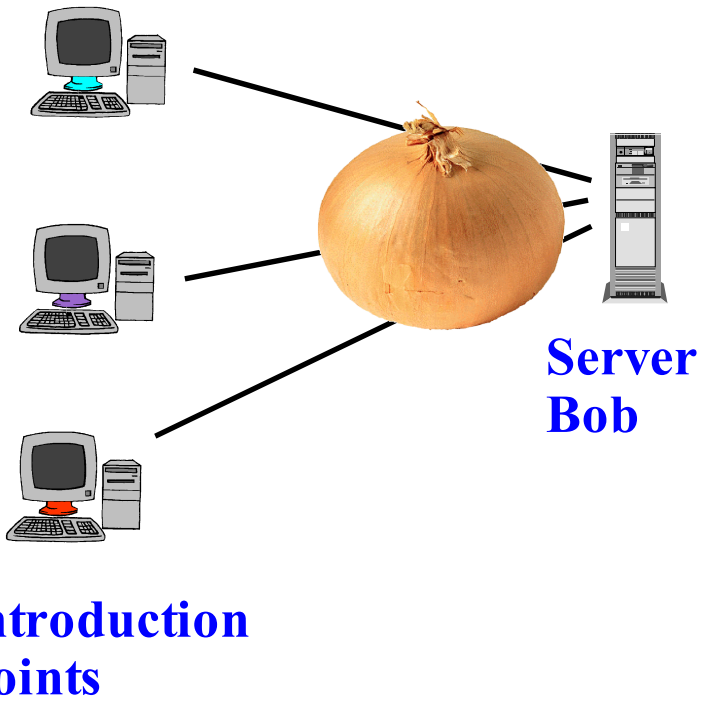
- Directory Servers
  - Maintain list of which onion routers are up, their locations, current keys, exit policies, etc.
  - Directory server keys ship with the code
  - These directories are cached and served by other servers, to reduce bottlenecks

Location Hidden Servers

- Alice can connect to Bob's server without knowing where it is or possibly who he is
- Can provide servers that
  - Are accessible from anywhere
  - Resist censorship
  - Require minimal redundancy for resilience in denial of service (DoS) attack
  - Can survive to provide selected service even during full blown distributed DoS attack
  - Resistant to physical attack (you can't find them)
- How is this possible?

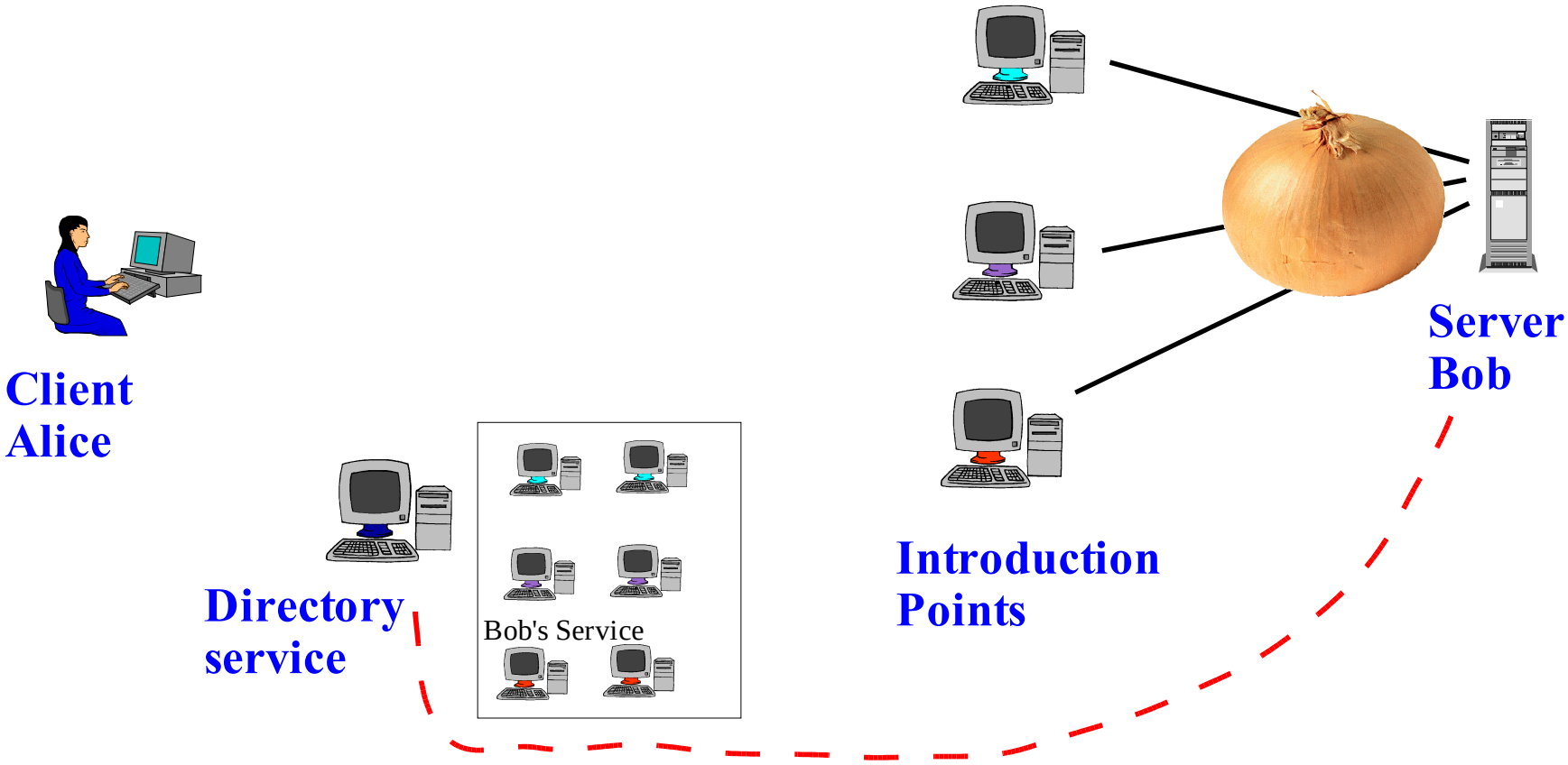
Location Hidden Servers

1. Server Bob creates onion routes to Introduction Points (IP)



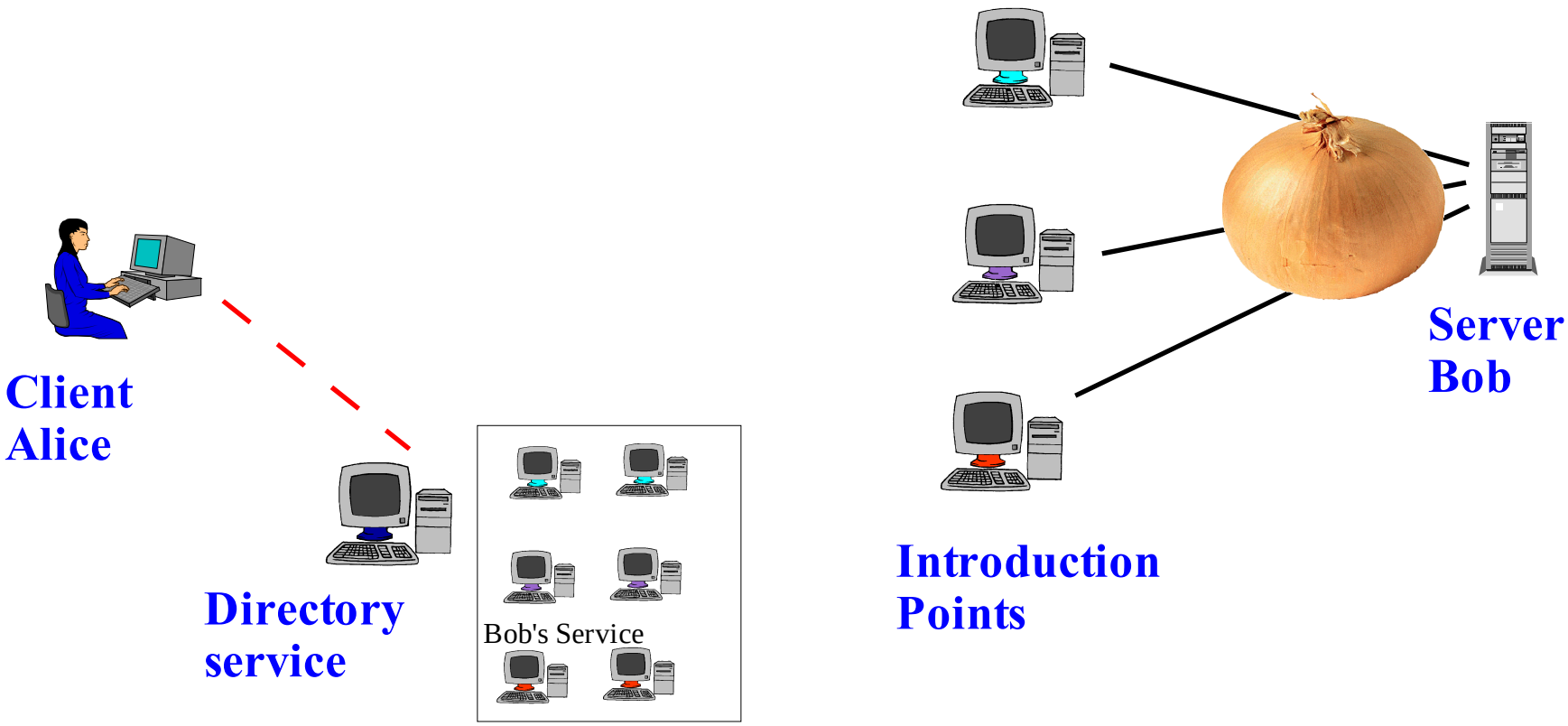
Location Hidden Servers

1. Server Bob creates onion routes to Introduction Points (IP)
2. Bob gets Service Descriptor incl. Intro Pt. addresses to Alice
   - In this example gives them to Service Lookup Server



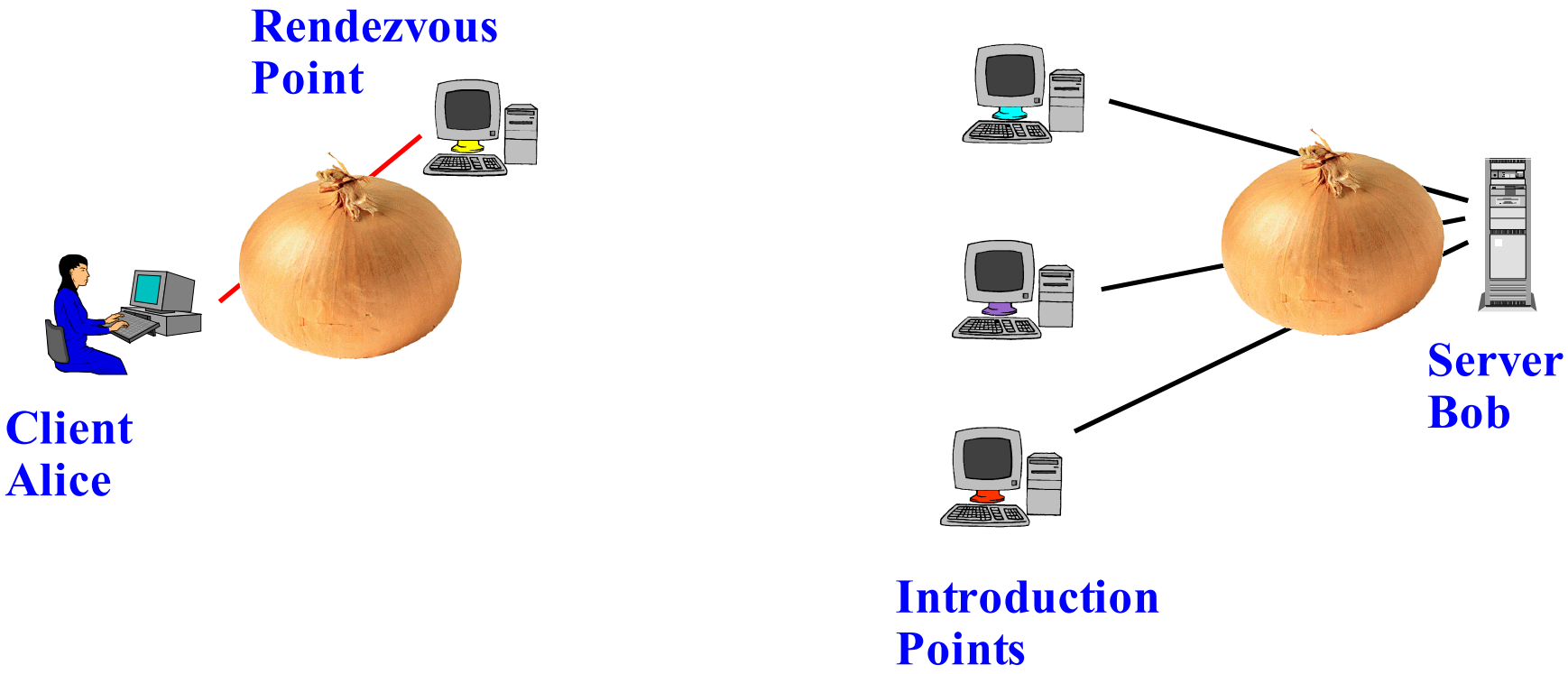
Location Hidden Servers

2'. Alice obtains Service Descriptor (including Intro Pt. address) at Lookup Server



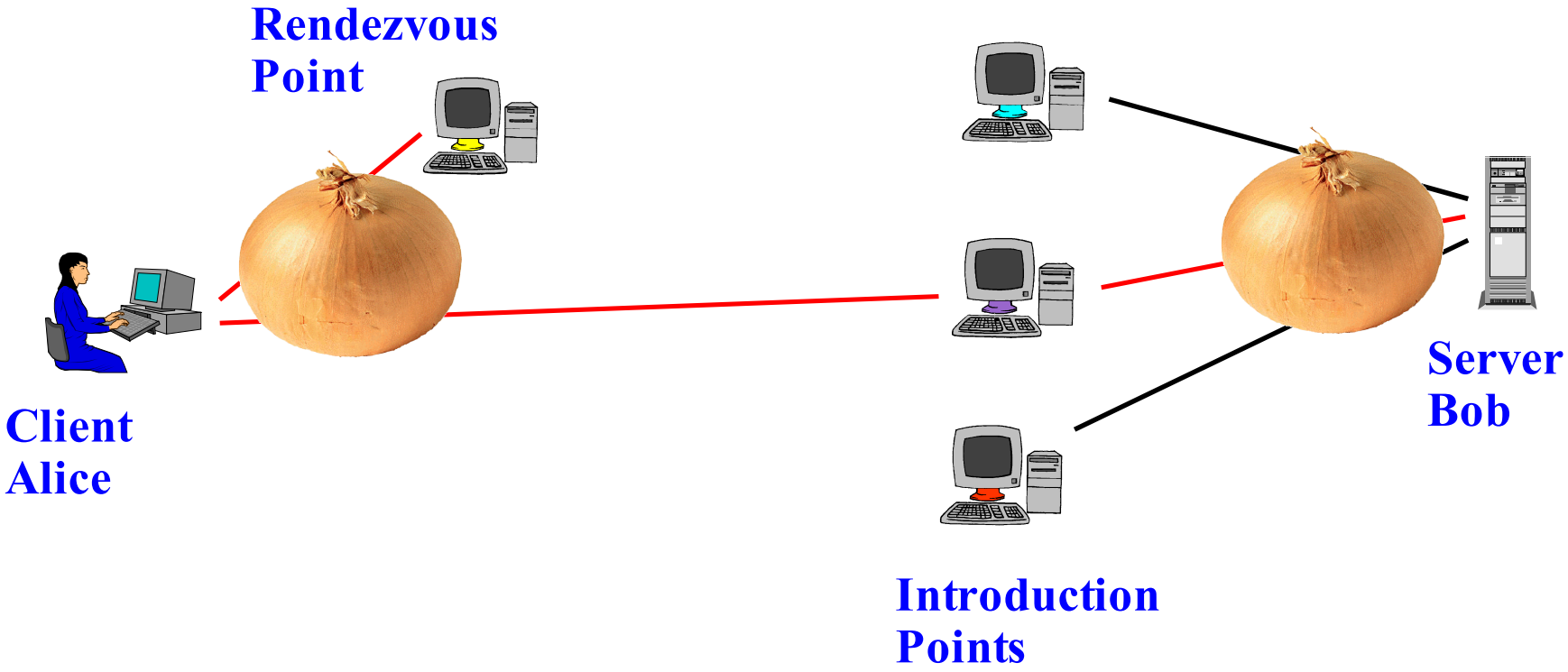
Location Hidden Servers

3. Client Alice creates onion route to Rendezvous Point (RP)



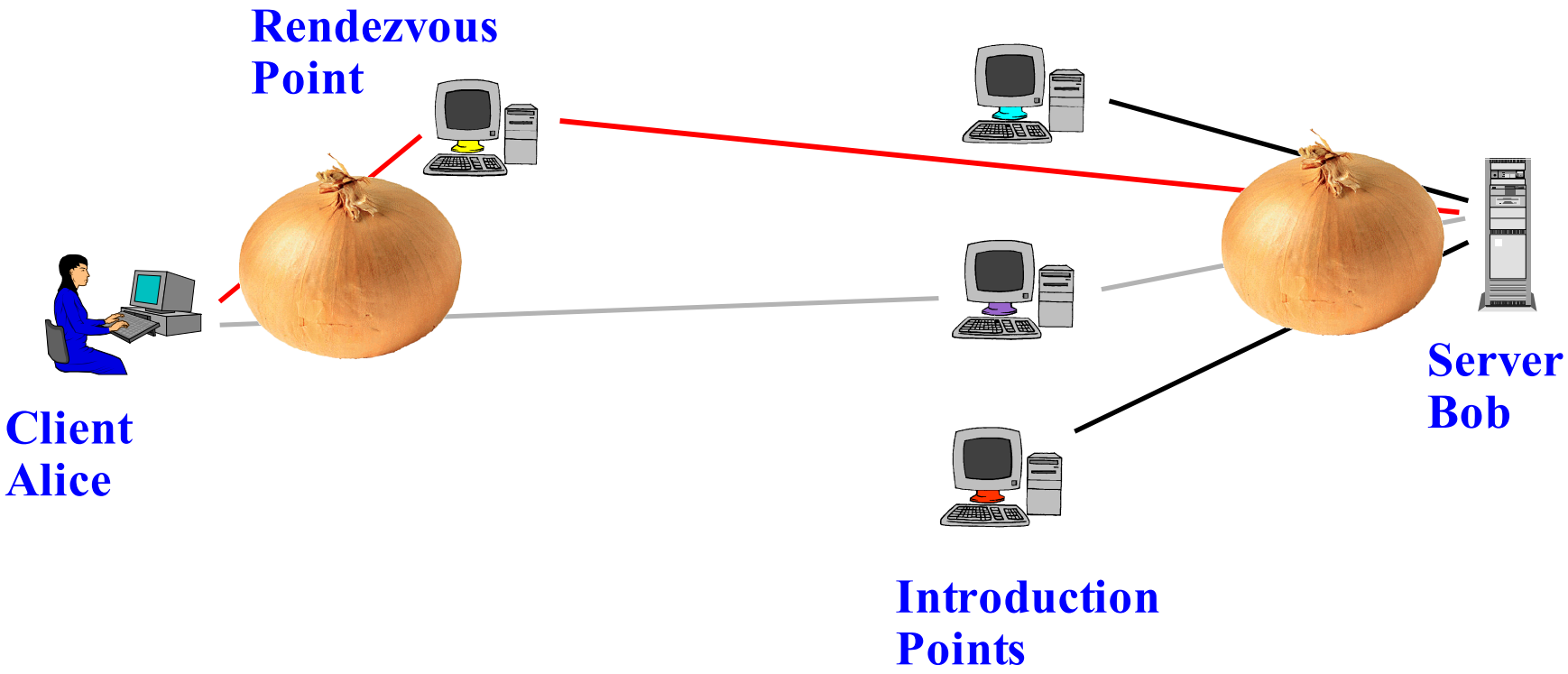
Location Hidden Servers

3. Client Alice creates onion route to Rendezvous Point (RP)
4. Alice sends RP addr. and any authorization through IP to Bob



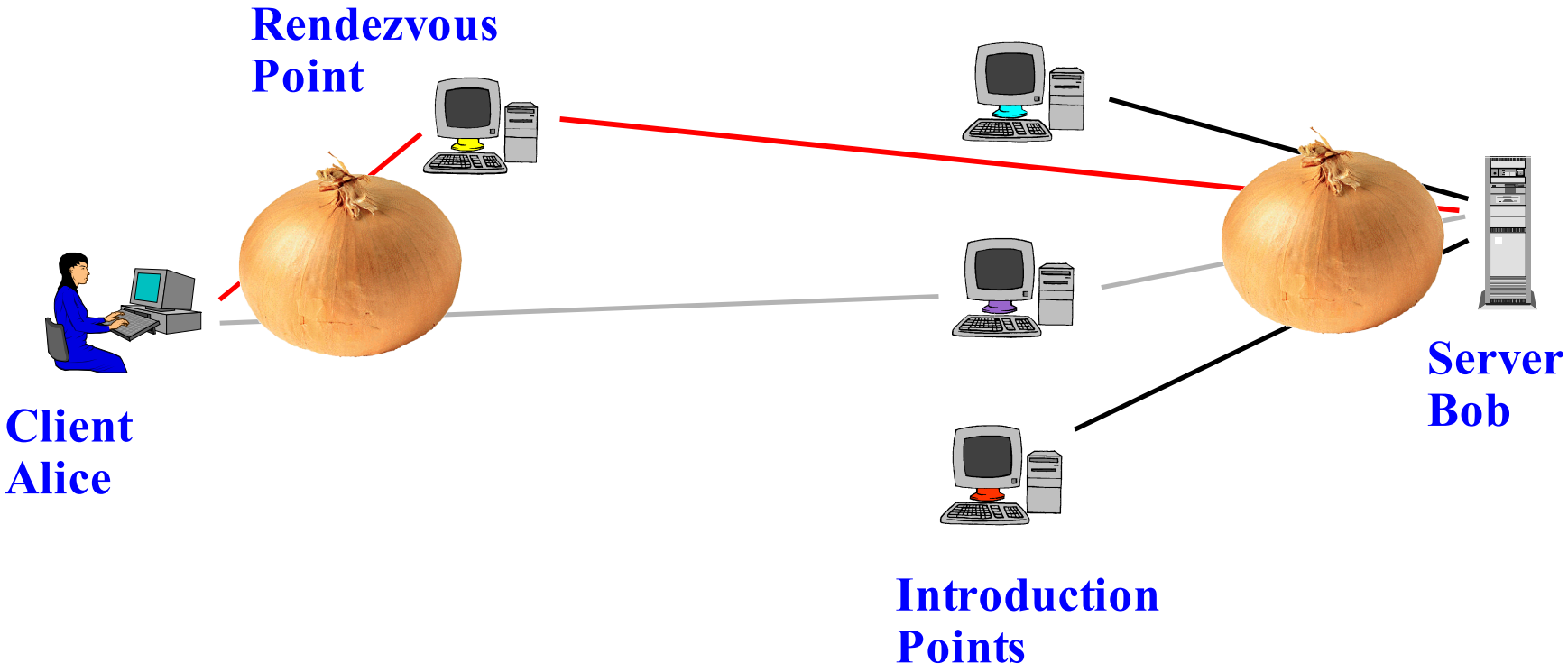
Location Hidden Servers

5. If Bob chooses to talk to Alice, connects to Rendezvous Point



Location Hidden Servers

5. If Bob chooses to talk to Alice, connects to Rendezvous Point
6. Rendezvous point mates the circuits from Alice and Bob



Demo

- First make a hidden-service that redirects to google
- Then install a local web server, and redirect to that.

Many webservers are tricky

- Need to bind to localhost only (or firewall the port)
- Turn off virtual hosts/names/etc
- Version strings, server-status often allowed from localhost
- Turn off PHP, etc
- Careful offering public websites + hidden services from the same Apache

Non-anonymity uses

- Run a service without an IP address
  - From deep inside the corporate lan!
- Authenticated servers you can run from anywhere and they look the same
  - On a USB drive?

Hidden IRC

- IRC servers as hidden services
- OFTC gateway as hidden service
- Decentralized Jabber servers where every user runs a hidden service Jabber?

Other notes

- Might want to back up your private_key file.
- If your computer isn't online all the time, that leaks information.
- More anonymity problems than normal Tor, since the adversary can force you to receive traffic.