Anonymity, Onion Routing, and Tor

Presented by: ?

(preserving my anonymity)

Online Anonymity: What?

- A concept where the identities of communicators are kept secret
- Anonymous online activity
  - Performing online communications
  - No one can (logically) infer who is talking to whom on a public network
- Online privacy requires more than just encrypting/decrypting data..

  ..we must also protect the identities!

Traffic Analysis

- Online message = Data + Headers
- Headers include:
  - Sender (IP address)
  - Receiver (IP address)
  - Size of data
  - Time sent
  - Etc..
- Even if the data is encrypted, revealing information can still be found in the headers
- Traffic analysis is the process of inferring who is talking to whom on a public network

Online Anonymity: Why?

- Public knowledge of the source and destination of internet traffic *could* be detrimental
  - Information could be used against the users by adversaries
- Adversaries:
  - Greedy corporations
    - Withhold health insurance to those who search for information about life-threatening diseases
    - Price discrimination based on location
  - Sneaky Advertisers
    - User profiling
    - Profile sharing (or even identity theft)
  - Censorship-happy governments
    - Penalties for accessing sensitive information
  - Militant organizations
    - Communicating with "the enemy" may be life-threatening

Online Anonymity: How?

- Anonymous proxies
  - Weaknesses:
    - Trusting an unknown proxy provider
    - Man in the middle attacks
- Onion Routing / Tor
  - Weakness:
    - To be discussed

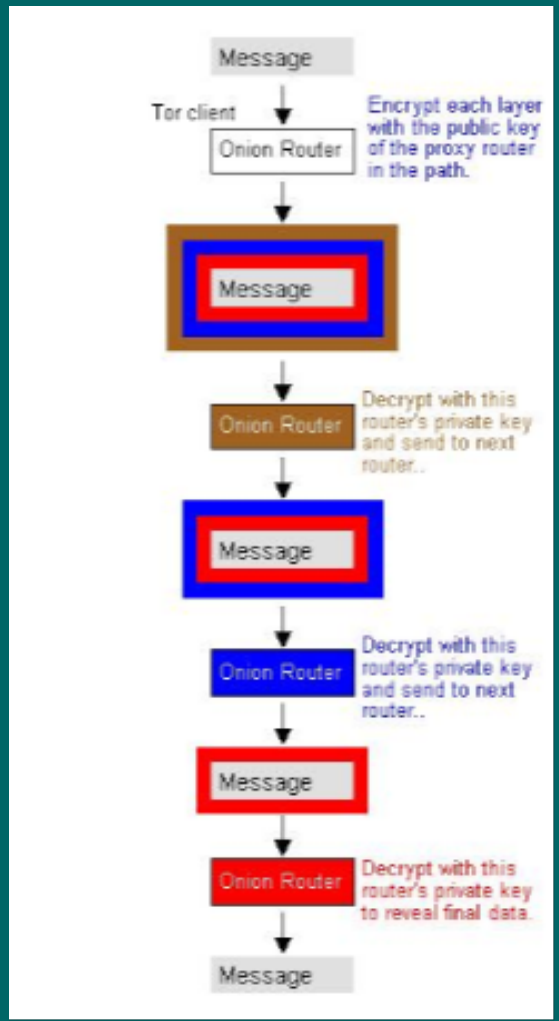
Onion Routing: What?

- Provides anonymous connections through "onion routers"
- Overview of process:
  1. Sender creates a message
  2. Message gets encrypted several times (the onion)
  3. Message gets sent through several onion routers
  4. Each onion router decrypts (peels a layer off of the onion) and passes it on to the next
  5. Finally, receiver receives a completely decrypted message

Onion Routing: How?

- Onion Router
  - A node in the onion routing network
- Onion Proxy
  - Gets message from sender, constructs onion, and determines onion route
- Exit Node
  - The last onion router sends the message to the receiver
- Onion
  - A layered, encrypted data object
  - Decryption results in:
    - Smaller onion (or message)
    - Location for the next router
    - Encryption key for return trip

Onion Routing: How?



Onion Routing: Why?

- Ensures anonymity
  - Each onion knows only its predecessor and successor
  - If a node is compromised, anonymity is not compromised
- Protects against traffic analysis
  - Traffic analysis only reveals the exit node and the receiver

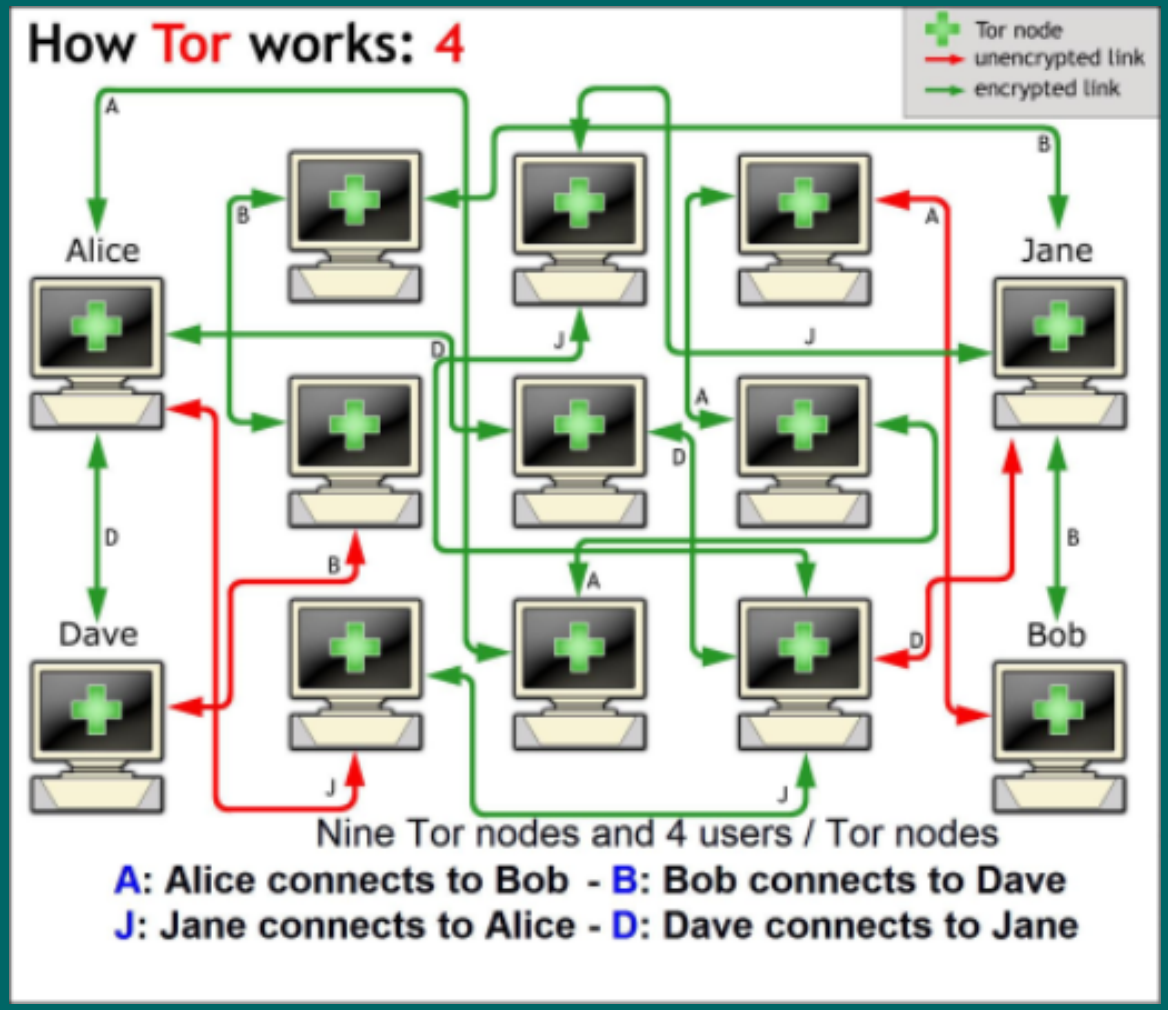
Onion Routing: Why not?

- Only a proof of concept was ever built
- Does not prevent against timing analysis
  - Analyze the time between when certain nodes send messages and other nodes receive
- Exit node sniffing (eavesdropping)
  - Traffic to/from exit node is not encrypted

Tor: What?

- The Onion Router
  - A second-generation onion router
- A routing service that is:
  - Free
  - More secure
  - Scalable
  - Widely-adopted

Tor: How?


How Tor works: 4
Tor node
unencrypted link
encrypted link
Alice
Jane
Dave
Bob
A
B
D
J
Nine Tor nodes and 4 users / Tor nodes
A: Alice connects to Bob - B: Bob connects to Dave
J: Jane connects to Alice - D: Dave connects to Jane

Tor: Why?

- Major improvements over first-generation
  - Perfect forward secrecy
    - Previous and subsequent messages can't be compromised even if an encryption key is compromised
  - Works with more than just browser-based traffic
    - SOCKS compliant
    - IM clients
  - DNS servers
    - DNS lookups go through the onion routing network instead of bypassing it
  - Many more features...

Tor: Extra features

- Works with Linux/OS X/Windows
- Firefox plug-in
- Pidgin plug-in
- Easy GUI

Questions