Feb-13 Lecture

Problem

- The Internet is a very public place
  - surveillance occurs at a variety of vantage points
- Encryption provides only limited value
  - Packet headers reveal a great deal about users
  - Signature detection reveals even more
- One need: end-to-end anonymity
- One solution: a distributed, anonymous overlay network

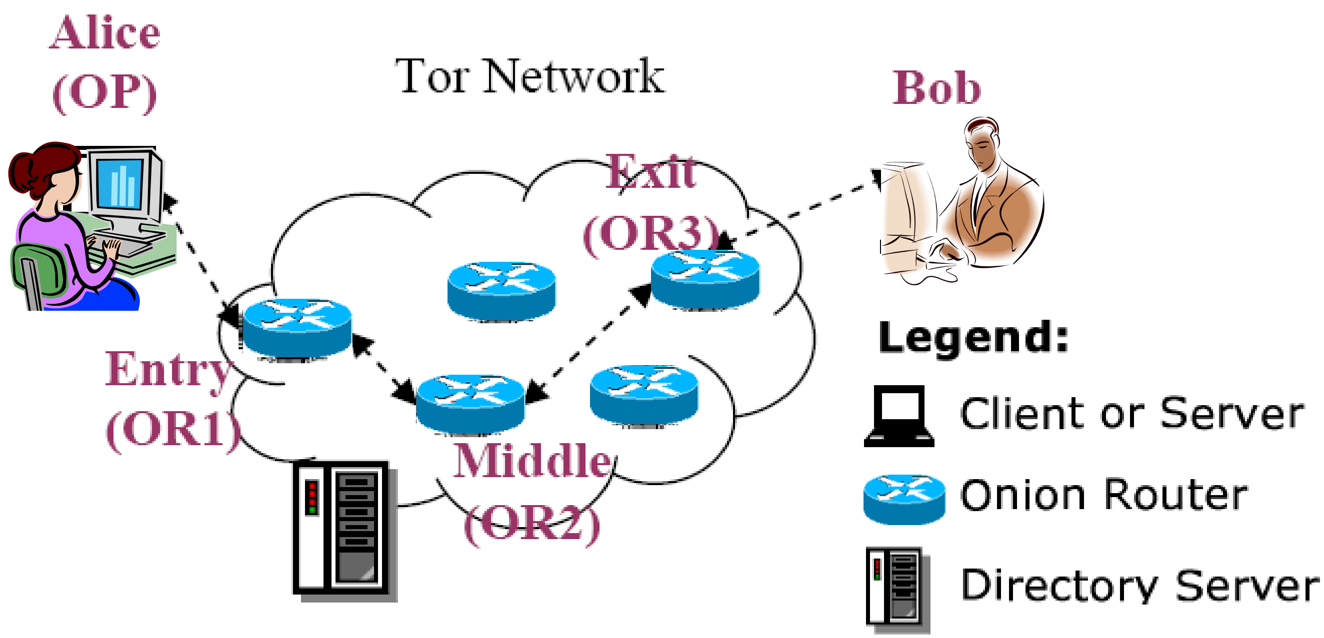
What is Tor

- Tor is just that: (1) distributed, (2) anonymous, (3) overlay network
- Individuals use Tor to keep websites from tracking them, or to connect to those Internet services blocked by their local Internet providers
- Tor's hidden services let users publish web sites and other services without needing to reveal the location of the site

Design

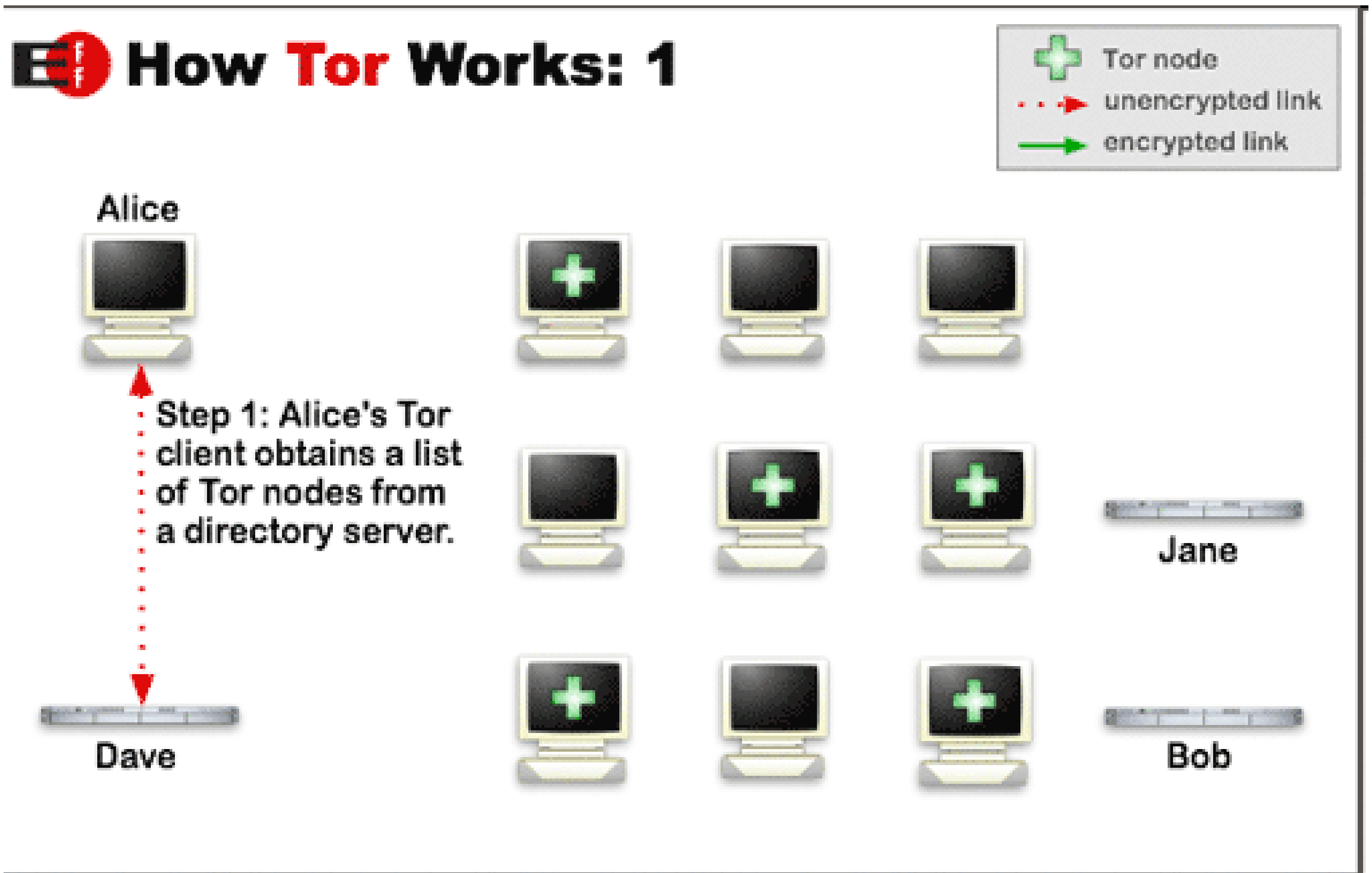
- Overlay network at the user level
- Onion Routers (OR) route traffic
- Onion Proxy (OP) fetches directories and creates virtual circuits on the network on behalf of users
- Uses TCP with TLS
- All data is sent in fixed size (bytes) cells

Components of Tor

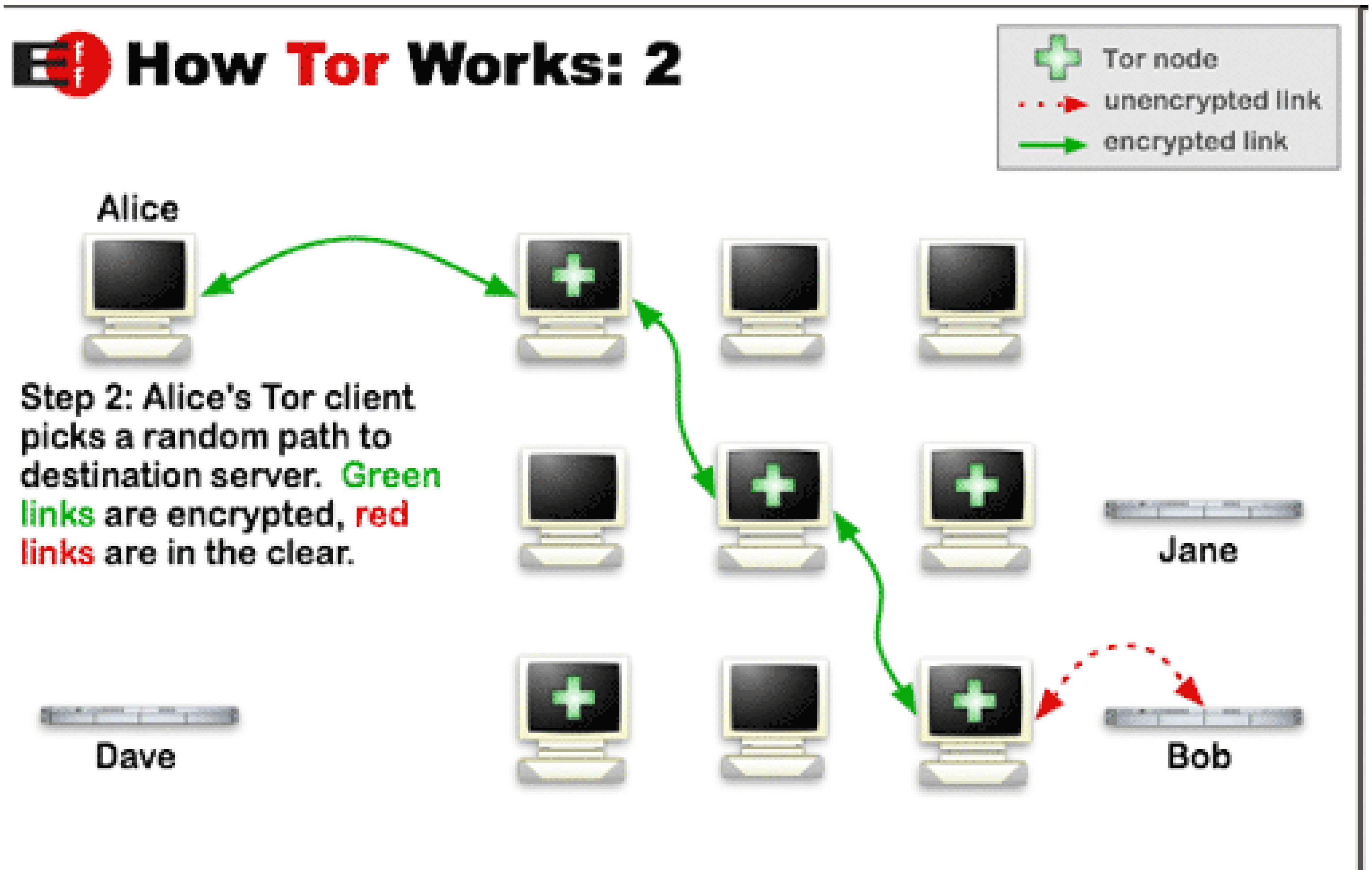


- **Client**: the user of the Tor network
- **Server**: the target TCP applications such as web servers
- **Tor (onion) router**: the special proxy relays the application data
- **Directory server**: servers holding Tor router information

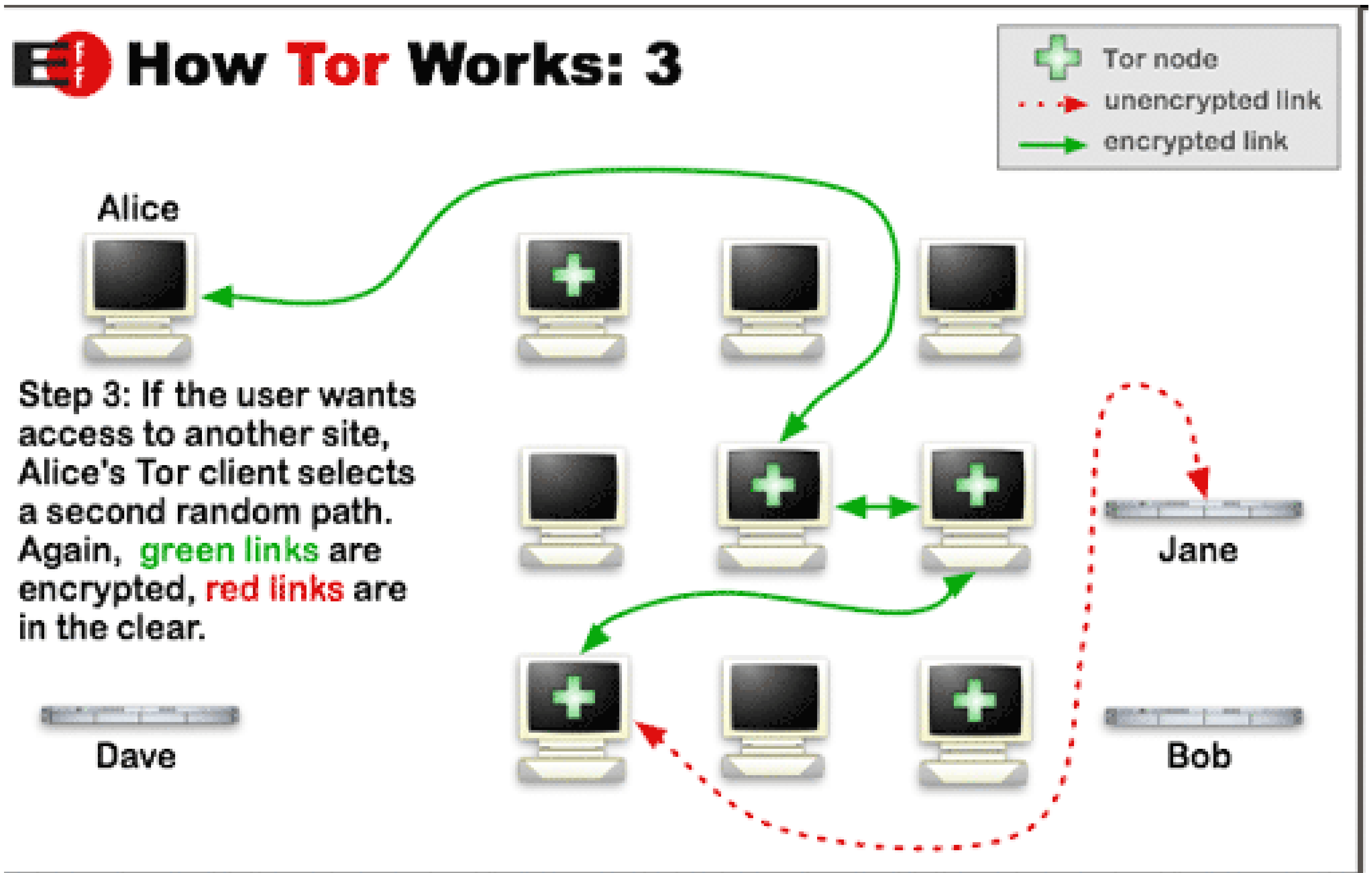
How does Tor work?



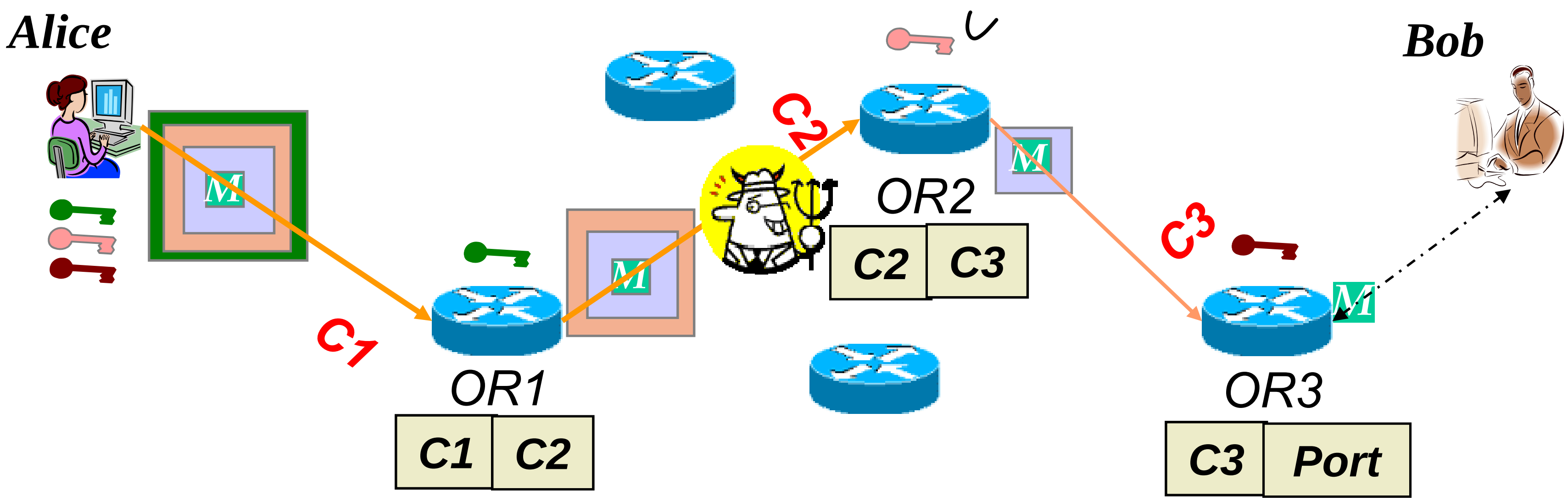
How does Tor work?



How does Tor work?



How Tor Works? -- Onion Routing



- A circuit is built incrementally hop by hop
- Onion-like encryption
  - Alice uses multiple AES keys (symmetric)
  - Messages are divided into equal sized cells
  - Each router knows only its predecessor and successor
  - Only the Exit router (OR3) can see the message, however it does not know where the message came from

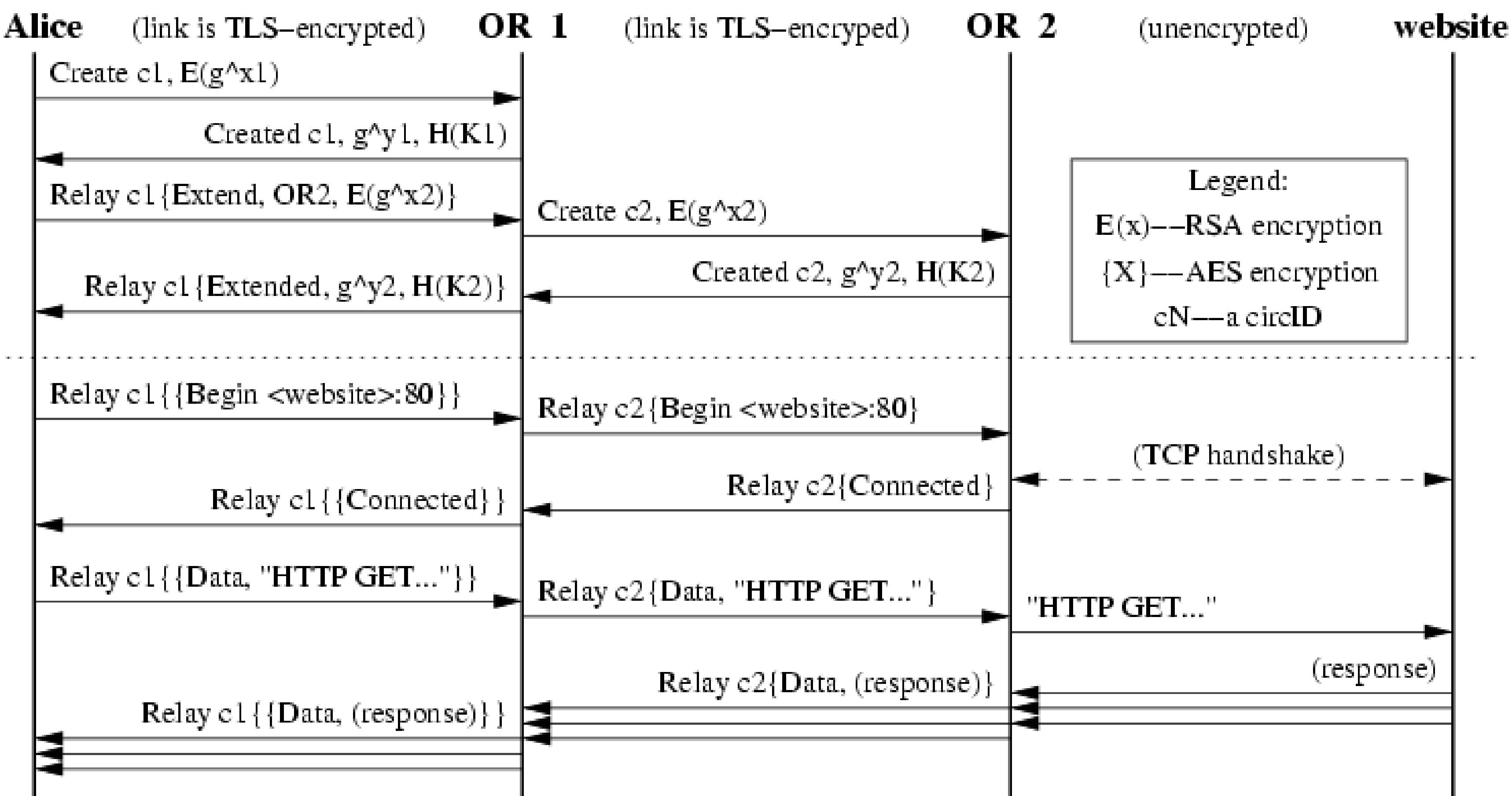
Cells

- All data is sent in fixed size (bytes) cells
- Control cell commands:
  - Padding, create, destroy
- Relay cell commands:
  - Begin, data, connected, teardown, ...

2	1	509 bytes
CircID	CMD	DATA

2	1	2	6	2	1	498
CircID	Relay	StreamID	Digest	Len	CMD	DATA

Commands in Use



- RSA is asymmetric (public key)
- AES is symmetric (shared key)

Creating the First Hop

- Symmetric key
- Diffie-Hellman
  - Sends ½
  - Encrypted with OR1 pub key

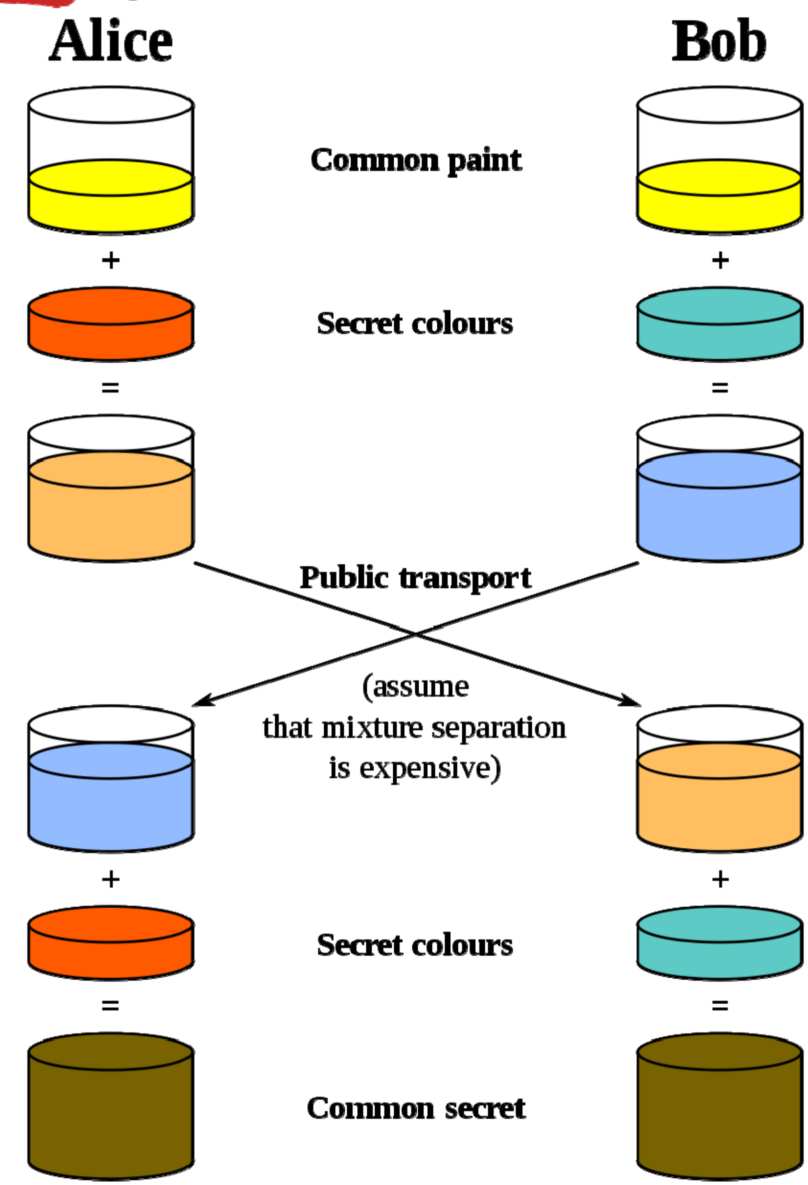
Constructing a circuit

A user's OP constructs circuits incrementally, negotiating a symmetric key with each OR on the circuit, one hop at a time. To begin creating a new circuit, the OP (call her Alice) sends a *create* cell to the first node in her chosen path (call him Bob). (She chooses a new circID C_{AB} not currently used on the connection from her to Bob.) The *create* cell's payload contains the first half of the Diffie-Hellman handshake (g^x), encrypted to the onion key of the OR (call him Bob). Bob responds with a *created* cell containing g^y along with a hash of the negotiated key $K = g^{xy}$.

- Receive second half back plus hash of key

Diffie-Hellman Keying

- g(x) is tan
- g(y) is blue
- Hash is of brown



Commands in Use

- Alice creates a circuit with the first OR
  - Negotiates a symmetric key
- Once the first leg is set up, Alice sends an extend message to the first OR
  - Specifies the second OR which the first OR will contact
  - First and second ORs use a second key
  - Alice knows about the second OR but the second OR doesn't know about Alice

Commands in Use

- Alice then sends a “begin” message to the exit router
  - The source IP address is the second OR
  - The destination IP address is the destination
- At the exit-point of the network, the egress OR has to send the message in the clear
  - Why?
- Does the web site know who is sending the request?

Summary of Who Knows What

- Alice pretty much knows everything
  - OR1, OR2, Bob, content
- OR1 knows Alice and OR2
  - Does not know what is being sent
- OR2 knows OR1 and Bob and contents
  - Does not know Alice
- Bob knows OR2 and contents
  - May or may not know it is Alice

Other Services

- Multiple TCP connections per “circuit”
- Congestion control (and bandwidth limiters)
- Exit policy descriptors
- Integrity checks

Non-Services

- No peer-to-peer model
  - In particular, the short-lived server aspect
- Some obscure attacks are still possible
  - If someone really wants to figure out what you're doing, they will
- No "protocol normalization"
  - Some protocols (e.g., HTTP) can still be the basis for signatures-based identification
- No "steganographic"
  - Does not hide who is connecting to the network

"Shining Light in Dark Places"

- Group created Tor router, then analyzed what they were able to see

- Basic Characteristics
  - A circuit was used for multiple connections but rotated over time

- Traffic collected in late 2007/early 2008

“Shining Light in Dark Places”

- More details on the circuit
  - The circuit typically consisted of three ORs
    - Entrance, middle, and exit
      - Entrance OR only OR that can see originator
      - Exit OR only UR that can see unencrypted traffic

- Set the exit policy
  - Set to “open exit policy” means more likely OR acts as exit router
  - Set to “exit traffic blocked” means more like OR acts as entrance (or middle) OR

"Shining Light in Dark Places"

Table 1. Exit traffic protocol distribution by number of TCP connections, size, and number of unique destination hosts.

Protocol	Connections	Bytes	Destinations
HTTP	12,160,437 (92.45%)	411 GB (57.97%)	173,701 (46.01%)
SSL	534,666 (4.06%)	11 GB (1.55%)	7,247 (1.91%)
BitTorrent	438,395 (3.33%)	285 GB (40.20%)	194,675 (51.58%)
Instant Messaging	10,506 (0.08%)	735 MB (0.10%)	880 (0.23%)
E-Mail	7,611 (0.06%)	291 MB (0.04%)	389 (0.10%)
FTP	1,338 (0.01%)	792 MB (0.11%)	395 (0.10%)
Telnet	1,045 (0.01%)	110 MB (0.02%)	162 (0.04%)
Total	13,154,115	709 GB	377,449

- Only 3.5% of HTTP connections were > 1MB

"Shining Light in Dark Places"

- Duh people, remember that the last hop is not secure/encrypted or anything
  - Lots of POP, IMAP, telnet, and FTP passwords
- (But this was 2007/2008, right?!?)
- Running an OR and observing traffic becomes a really good easy way to look for passwords
- How often does it happen?
  - How can you even tell?

"Shining Light in Dark Places"

- Tools like wireshark tend to try and do reverse DNS to replace an IP address with a host name in the display
  - Such DNS queries can be tracked
  - OR exit routers generating significant DNS reverse-lookup queries are likely snooping traffic

- Goes without saying, but lots of the traffic was malicious traffic
  - Researchers received a lot of complaints based on tracing the source IP addr of traffic they put in the network

“Shining Light in Dark Places”

Client Distribution		Router Distribution	
Country	*Total*	*Country*	*Total*
Germany	2,304	Germany	374
China	988	United States	326
United States	864	France	69
Italy	254	China	40
Turkey	221	Italy	36
United Kingdom	170	Netherlands	35
Japan	155	Sweden	35
France	150	Finland	25
Russia	146	Austria	24
Brazil	134	United Kingdom	24

Relative Tor Usage	
Country	*Ratio*
Germany	7.73
Turkey	2.47
Italy	1.37
Russia	0.89
China	0.84
France	0.77
United Kingdom	0.75
United States	0.62
Brazil	0.56
Japan	0.32

Exam

- Take home, open note, book, Internet
  - Made available at approximately 8:00am on 2/27 (Wed)
  - Due at 11:59pm on 2/28 (Thur)
  - To be done **individually**

- 8-10 multi-part essay questions

- Duration goal: the exam should take a approx. 4 hrs
  - The more you know the notes and papers, the less time it will take

- No class on Wednesday
  - I will be online to answer questions