Lesson 07-01: Network Security - Tor

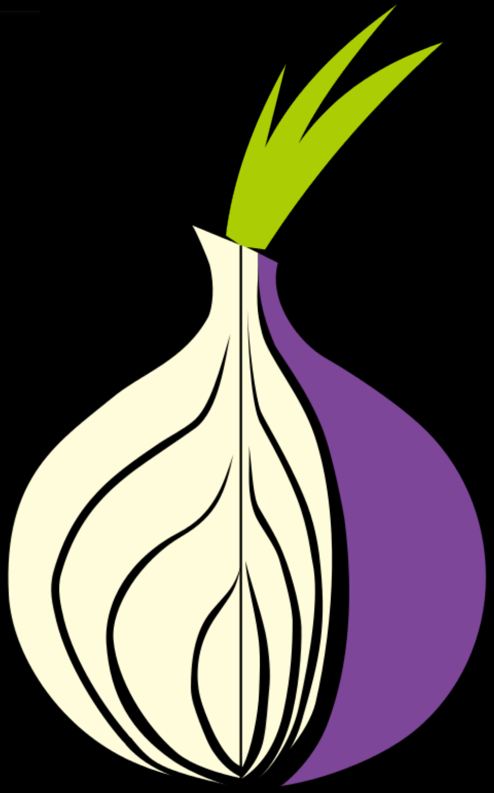
CS 356 Computer Networks

Mikyung Han

mhan@cs.utexas.edu

Tor: Enabling Anonymous Communication Over the Internet

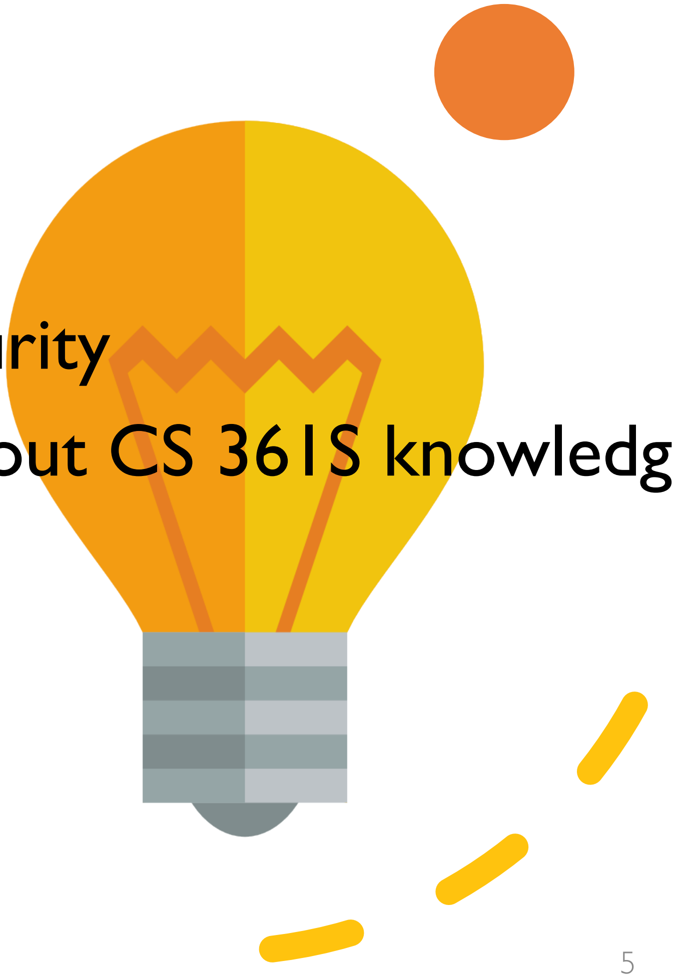
WE ARE
ANONYMOUS



WE ARE NOT SO ANONYMOUS

Primer Slides

- Quick review of terminologies/techniques in security
- Included for any CS student to follow easily without CS 361S knowledge



Why Tor?

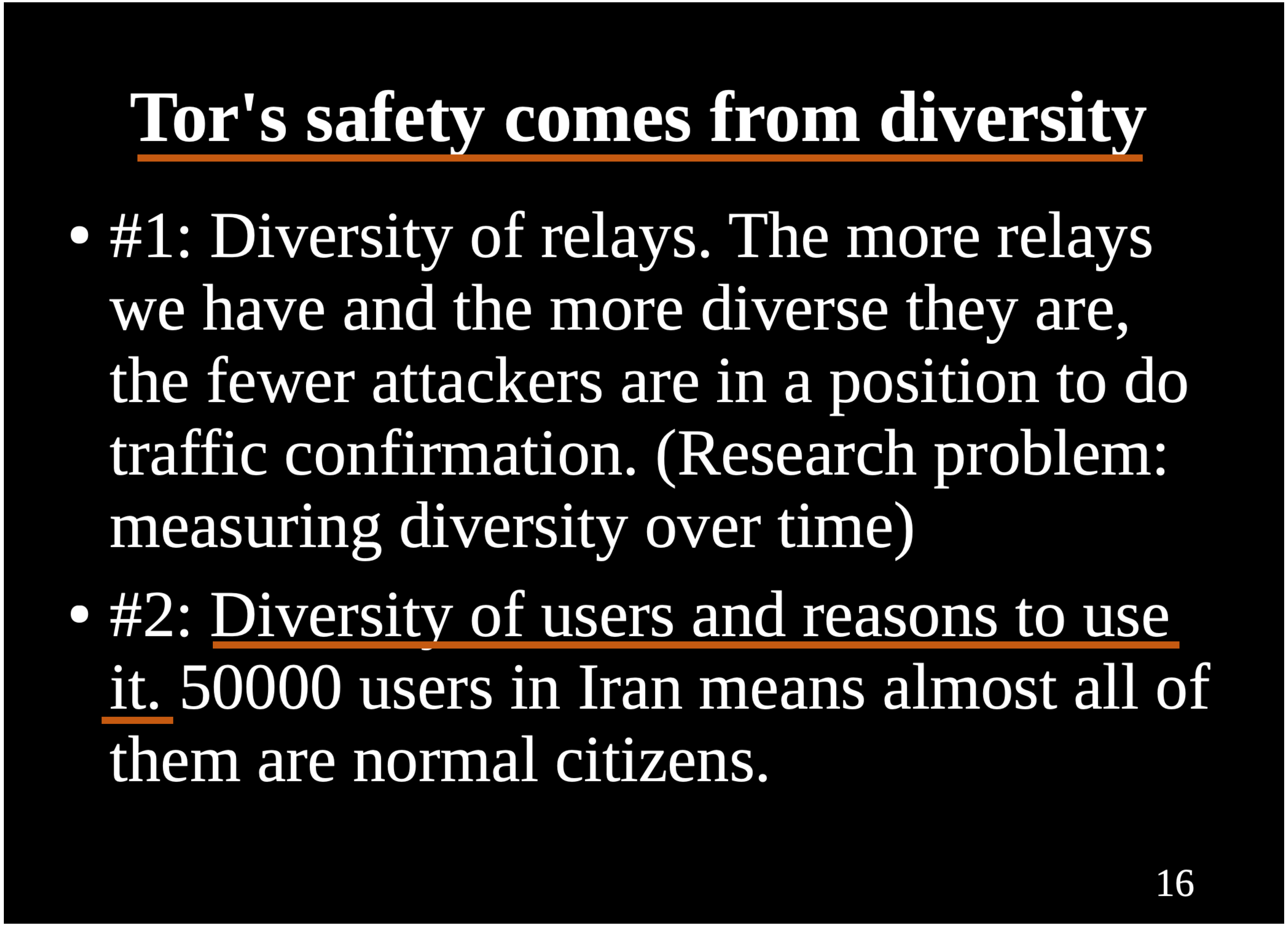
- Practical: It's a real network used by real users
- Popular: 7K relays, 200 Gbit/s of traffic, 2M+ daily users
- Philosophical: Freedom of speech is fundamental in democracy
- Publication: Active research being done on Tor
  A great topic for undergraduate research!

Privacy and security matters to all of us!

Who are these 2,000,000+ users?

Besides shoppers of underground market, Tor is used by

- Normal people
- Journalists
- Activists and whistleblowers
- Law enforcement officers
- Militaries
- Special support group
- etc



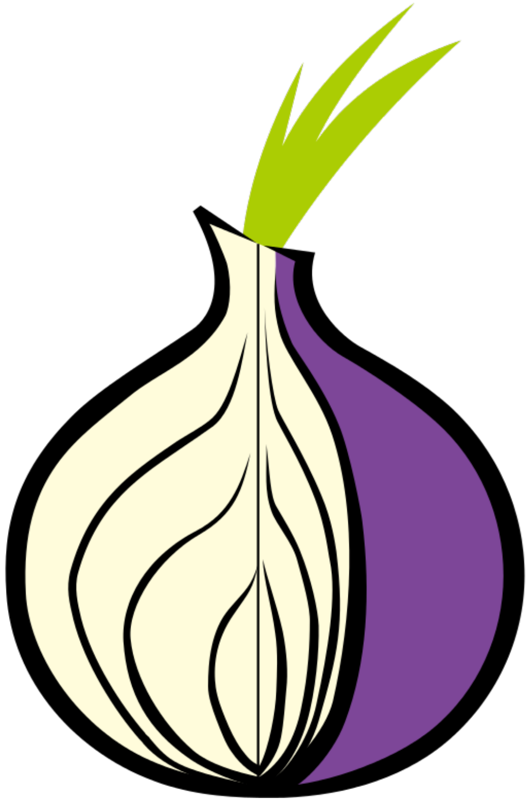
Tor's safety comes from diversity

- #1: Diversity of relays. The more relays we have and the more diverse they are, the fewer attackers are in a position to do traffic confirmation. (Research problem: measuring diversity over time)
- #2: Diversity of users and reasons to use it. 50000 users in Iran means almost all of them are normal citizens.

16

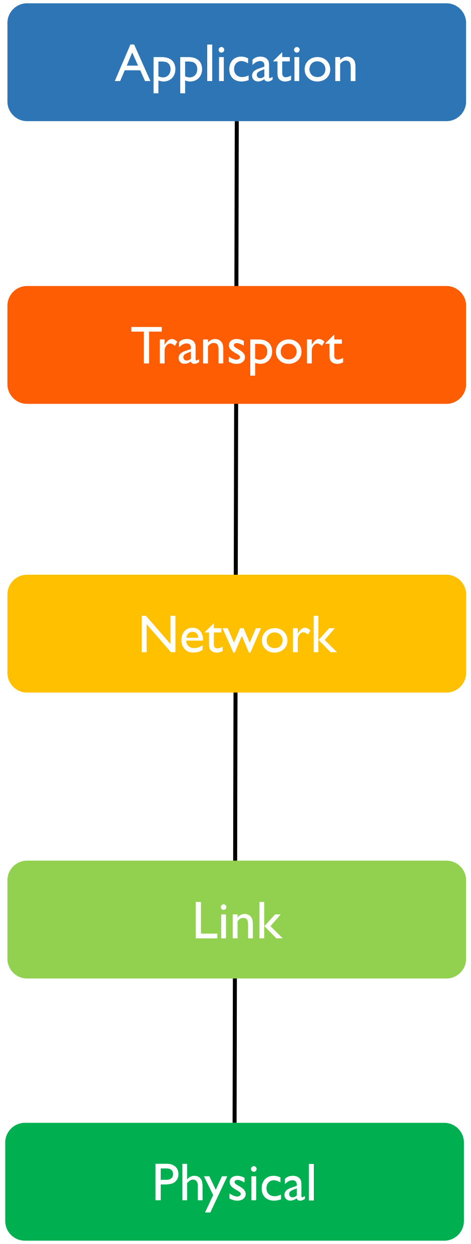
Roger Dingledine at DEFCON 27 (8. 2019)

So what is Tor?



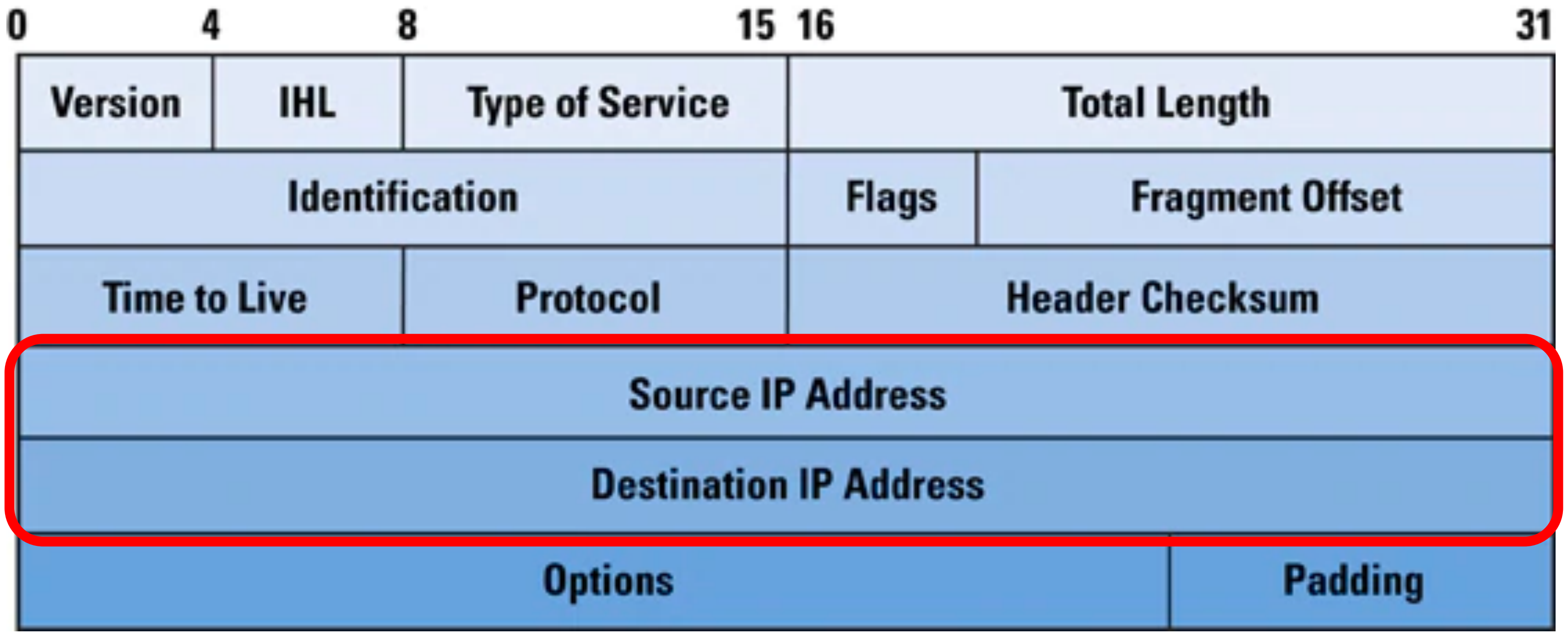
Outline

1. Intro
2. Network Primer
3. What is Tor
4. Security Primer
5. How Tor Works
6. Attacks and Censorship on Tor
7. Assignments
8. Teaching Philosophy



Example Protocols		Responsible for
FTP, HTTP, SMTP	Application	application specific needs
TCP, UDP	Transport	process to process data transfer
IP	Network	host to host data transfer across different networks
Ethernet, WiFi	Link	data transfer between physically adjacent nodes
802.3 PHY	Physical	bit-by-bit or symbol-by-symbol delivery

What do you see in the IP header?

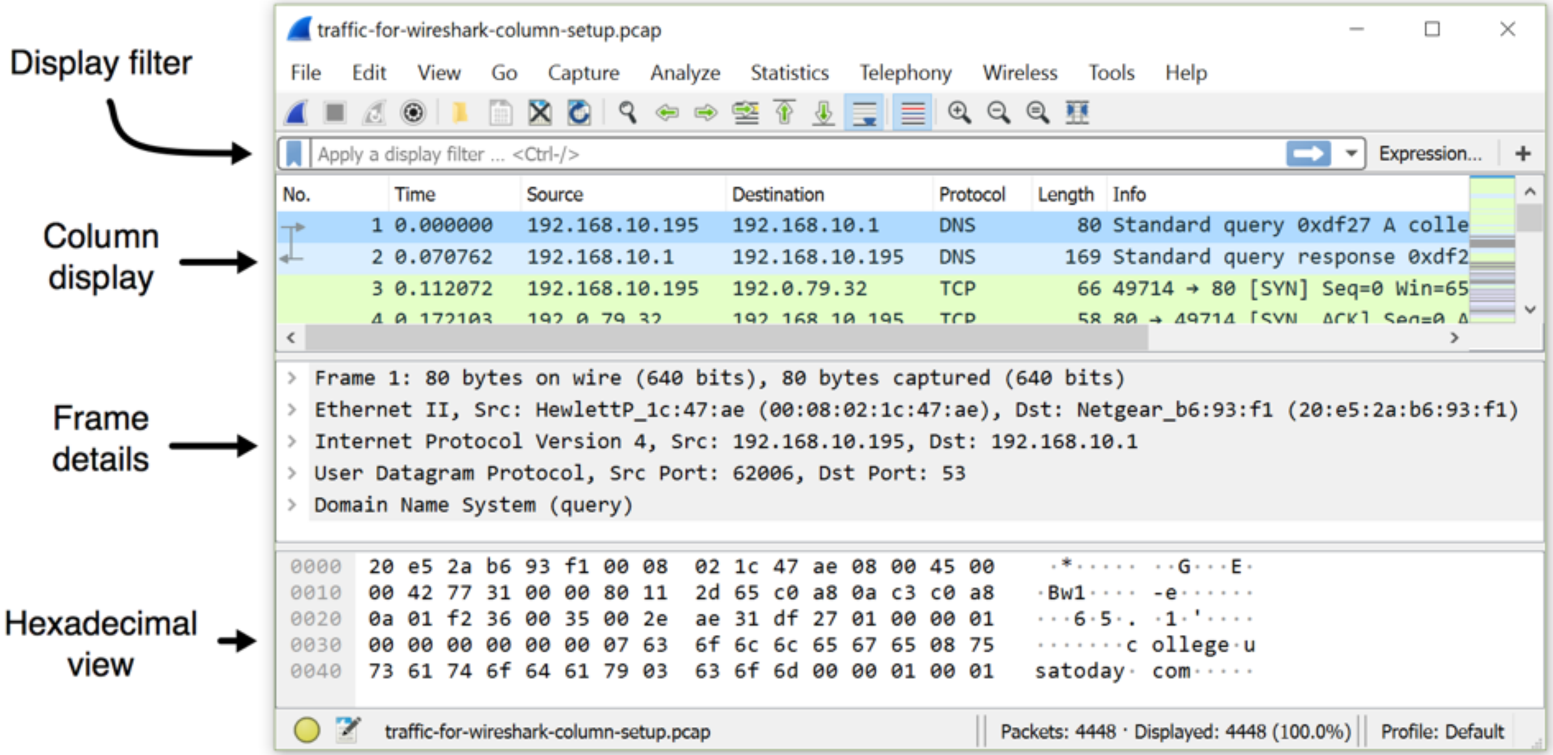


0	4	8 15	16		31
Version	IHL	Type of Service	Total Length		
Identification			Flags	Fragment Offset	
Time to Live		Protocol	Header Checksum		
Source IP Address					
Destination IP Address					
Options					Padding

This is a bad news if you want anonymity

WIRESHARK

- Free open-source packet analyzer
- https://www.wireshark.org/



Display filter

Column display

Frame details

Hexadecimal view

traffic-for-wireshark-column-setup.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
+

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.10.195	192.168.10.1	DNS	80	Standard query 0xdf27 A colle
2	0.070762	192.168.10.1	192.168.10.195	DNS	169	Standard query response 0xdf2
3	0.112072	192.168.10.195	192.0.79.32	TCP	66	49714 → 80 [SYN] Seq=0 Win=65
4	0.172103	192.0.79.32	192.168.10.195	TCP	58	80 → 49714 [SYN, ACK] Seq=0 A

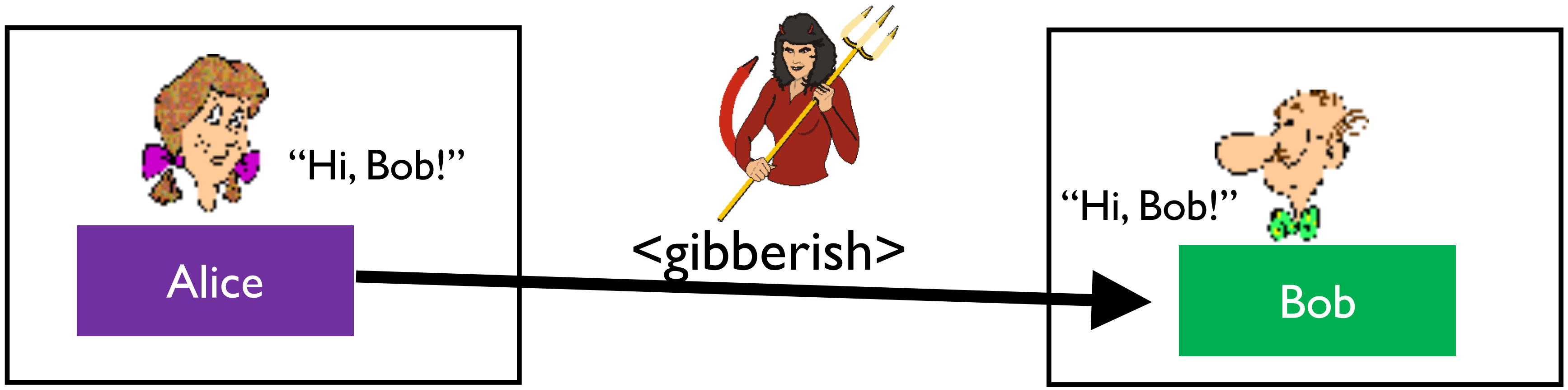
```
> Frame 1: 80 bytes on wire (640 bits), 80 bytes captured (640 bits)
> Ethernet II, Src: HewlettP_1c:47:ae (00:08:02:1c:47:ae), Dst: Netgear_b6:93:f1 (20:e5:2a:b6:93:f1)
> Internet Protocol Version 4, Src: 192.168.10.195, Dst: 192.168.10.1
> User Datagram Protocol, Src Port: 62006, Dst Port: 53
> Domain Name System (query)
```

```
0000  20 e5 2a b6 93 f1 00 08  02 1c 47 ae 08 00 45 00    ·*····· ··G···E·
0010  00 42 77 31 00 00 80 11  2d 65 c0 a8 0a c3 c0 a8   ·Bw1···· -e······
0020  0a 01 f2 36 00 35 00 2e  ae 31 df 27 01 00 00 01   ···6·5·. ·1·'····
0030  00 00 00 00 00 00 07 63  6f 6c 6c 65 67 65 08 75   ·······c ollege·u
0040  73 61 74 6f 64 61 79 03  63 6f 6d 00 00 01 00 01   satoday· com·····
```

traffic-for-wireshark-column-setup.pcap | Packets: 4448 · Displayed: 4448 (100.0%) | Profile: Default

What about encryption?

Encryption is NOT enough for anonymity: Encryption just protects content



"Hi, Bob!"

Alice

<gibberish>

"Hi, Bob!"

Bob

Even if the communication is encrypted

By observing packets, one can

- infer who is talking to whom at what time for how long
- infer physical locations
- use that to track behaviors and interests

Internet communication is NOT anonymous!

To provide anonymity and privacy, we need another layer in network stack

For anonymity

Need clever routing to skirt surveillance

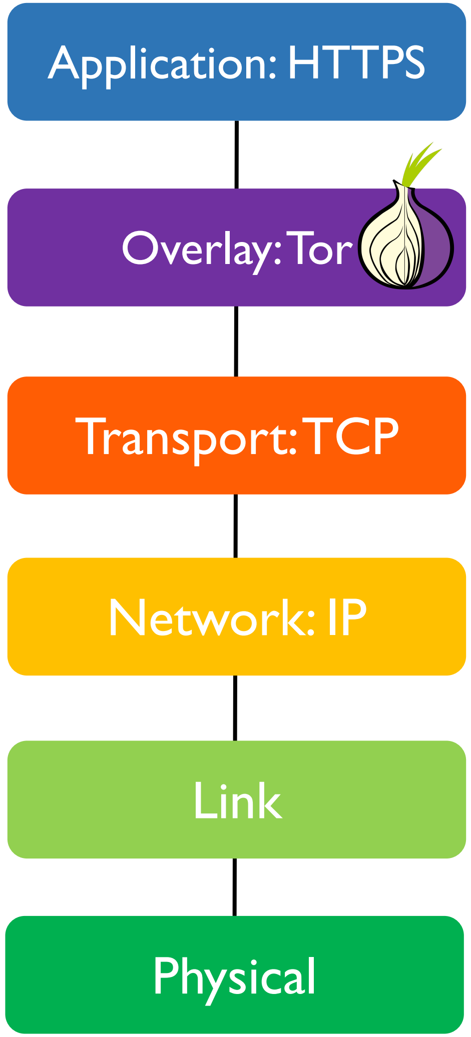
For privacy

Need encryption over each hop

Outline

1. Intro
2. Network Primer
3. What is Tor
4. Security Primer
5. How Tor Works
6. Attacks and Censorship on Tor
7. Assignments
8. Teaching Philosophy

Tor is an overlay network designed to provide anonymous communication



Application: HTTPS

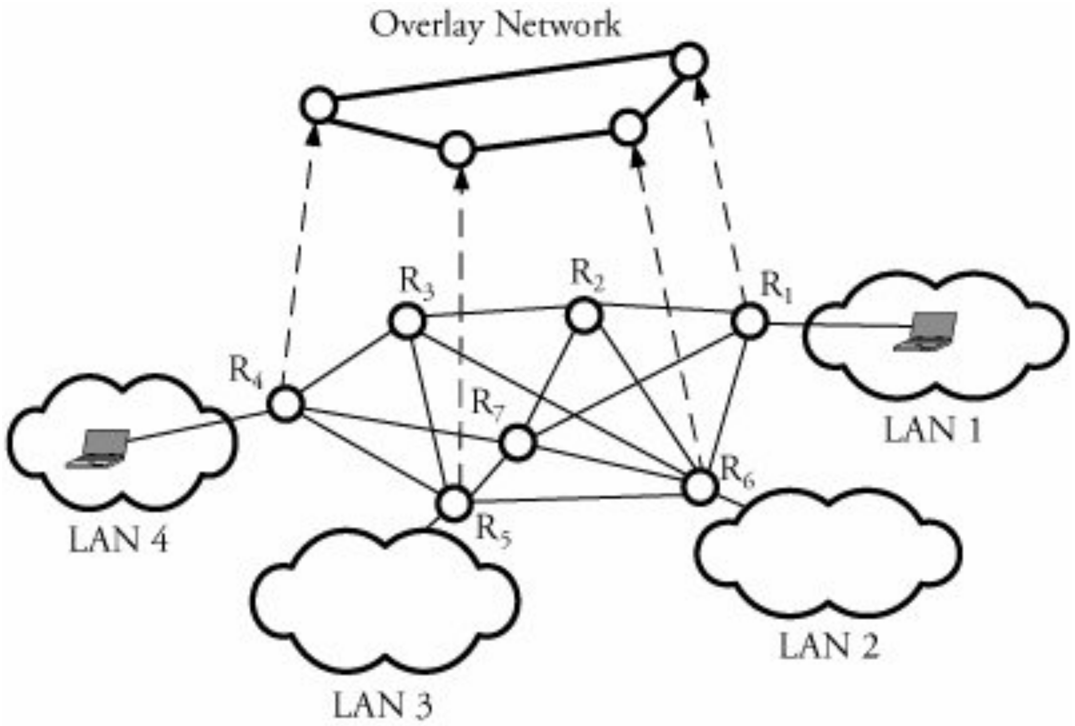
Overlay: Tor

Transport: TCP

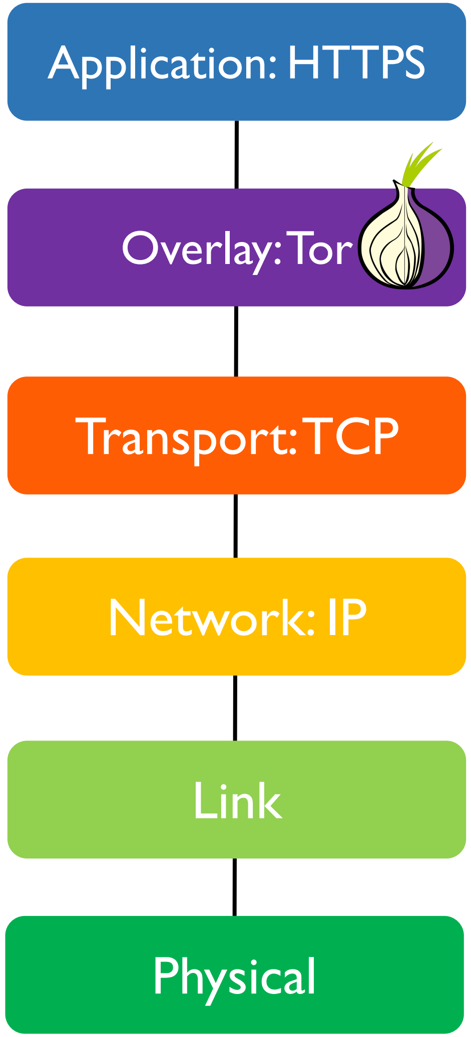
Network: IP

Link

Physical



In Tor's overlay network, each hop is a separate TCP connection



Application: HTTPS

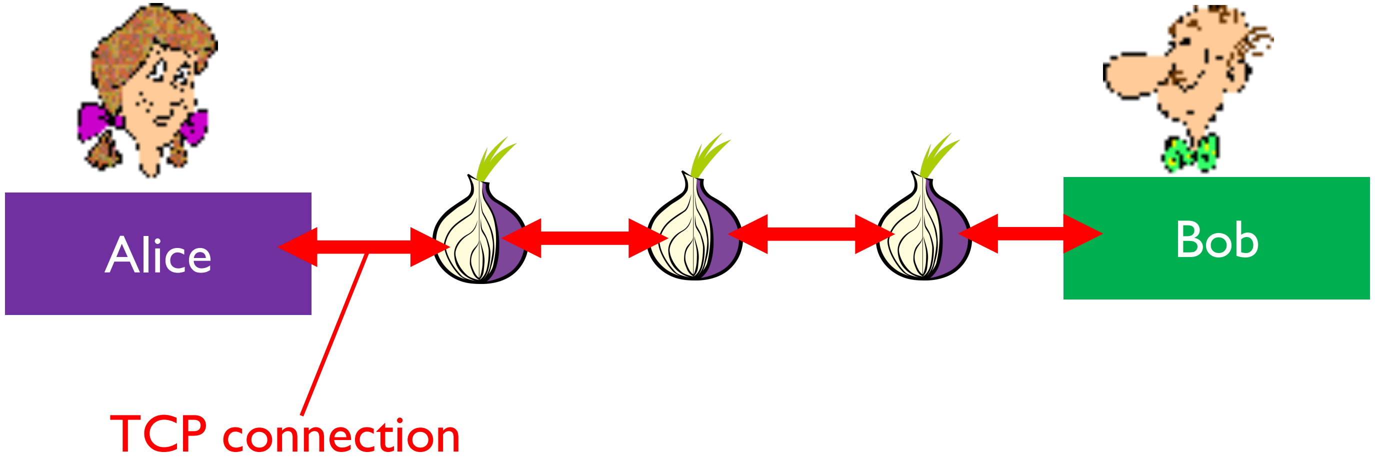
Overlay: Tor

Transport: TCP

Network: IP

Link

Physical

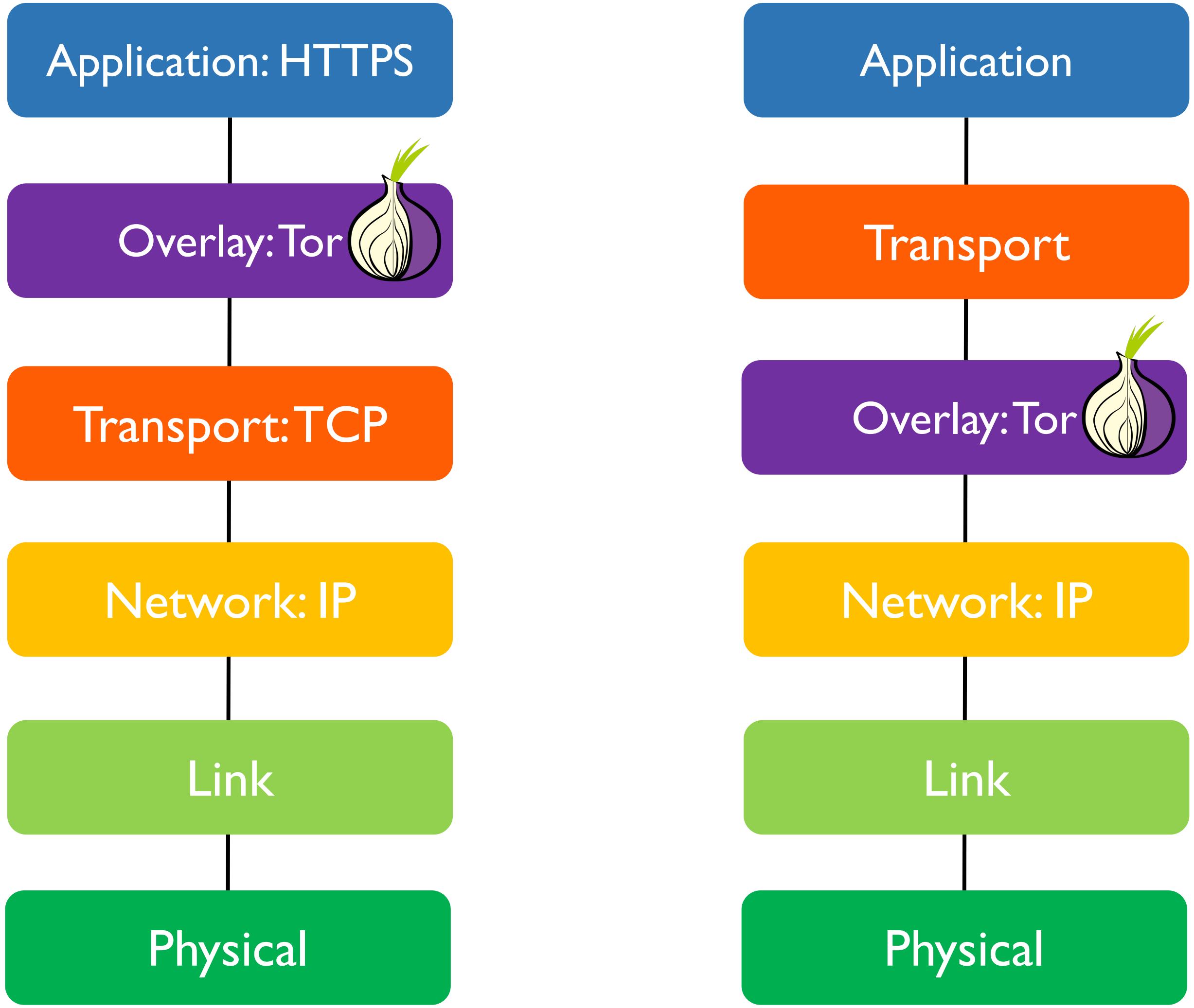


Alice

Bob

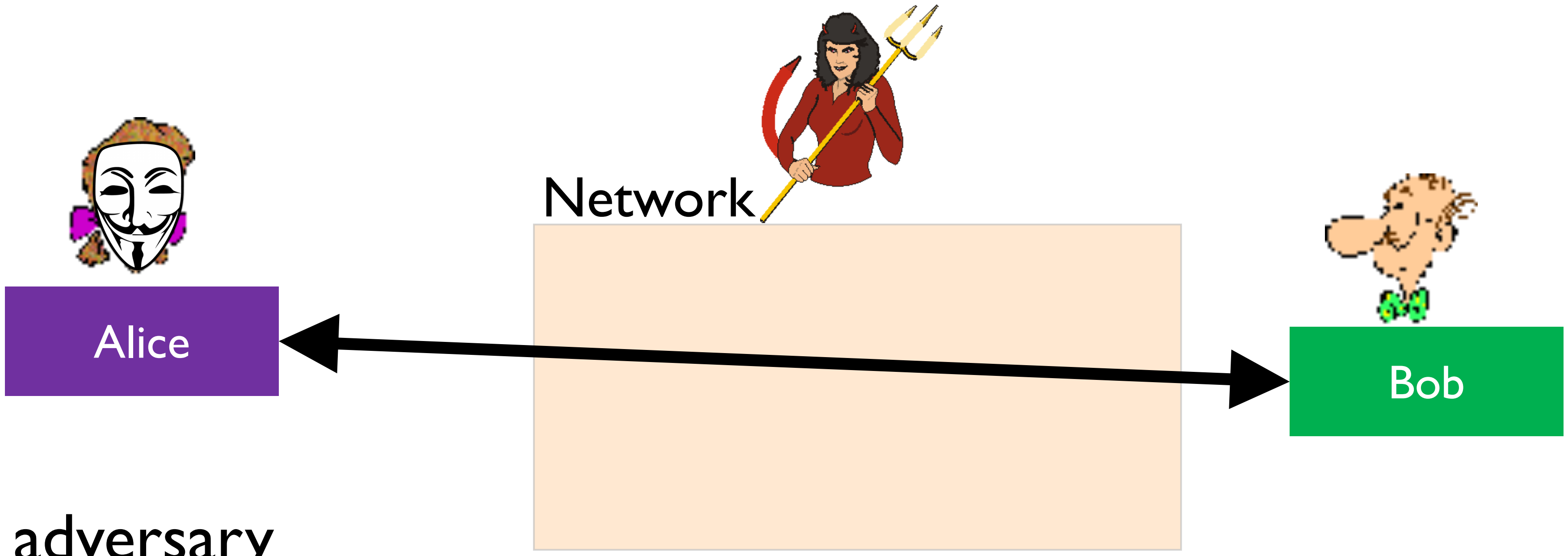
TCP connection

Tor’s design choice on overlay network



Tor is an overlay network designed to provide anonymous communication

Defining Anonymity



An adversary

- knows Alice is online
- knows there are some communication activities to Bob
- Does NOT know it is Alice that is talking with Bob

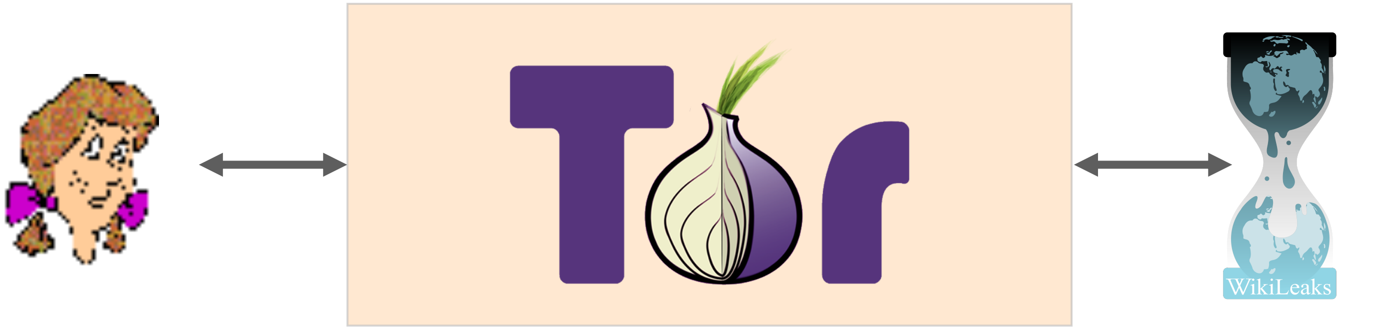
What about VPN?

- Doesn't VPN already provide anonymity?
- What is the difference between Tor and VPN?
- Pros and cons?

Group discussion questions

Answers will come as we learn more about Tor!

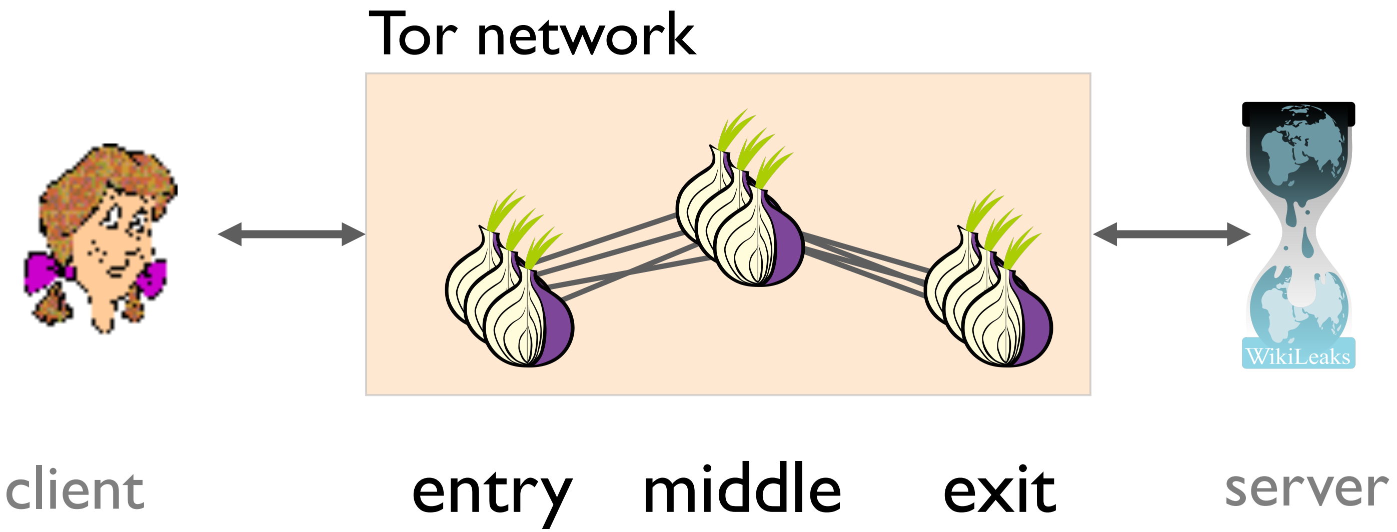
Tor aims to prevent adversaries from following packets from client to server



client

server

To do that,
Tor bounces traffic around a network of relays



Tor network

client

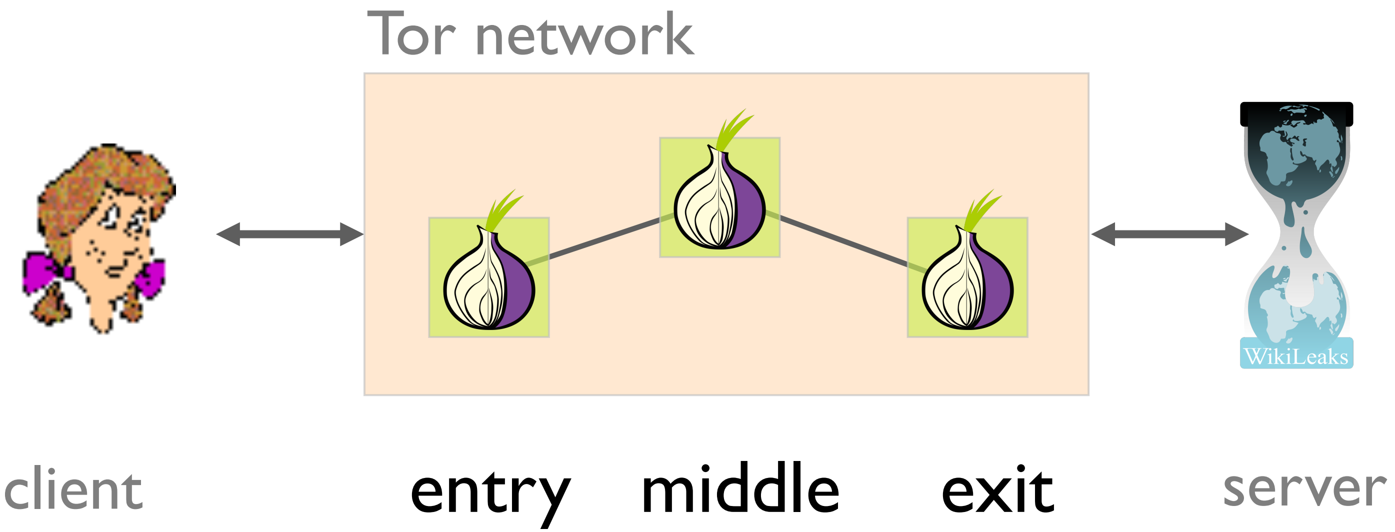
entry

middle

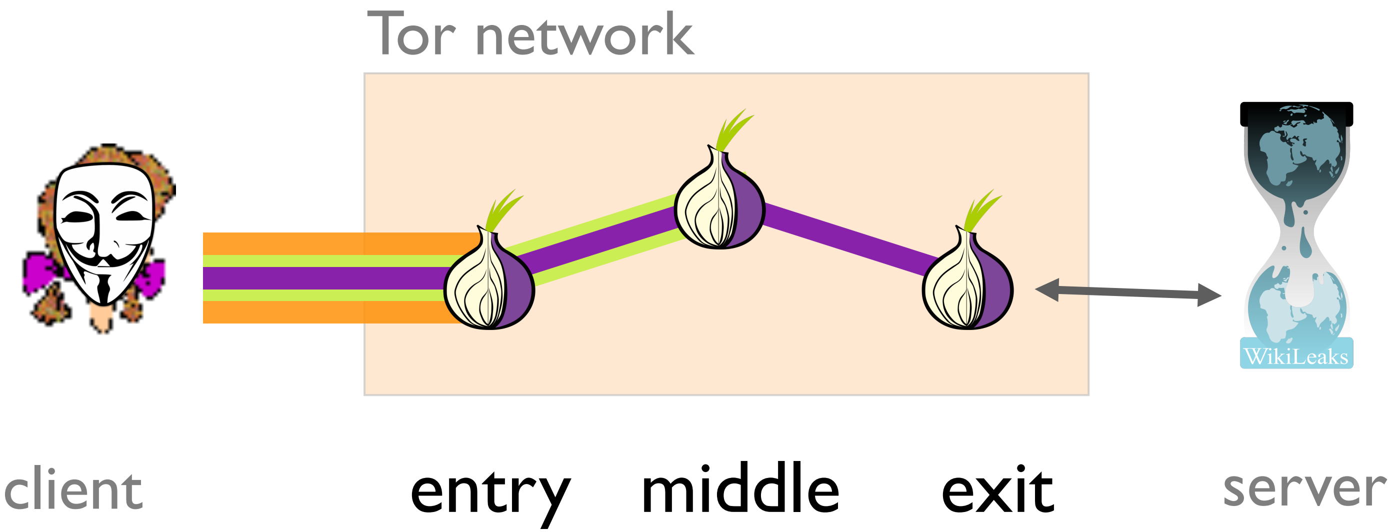
exit

server

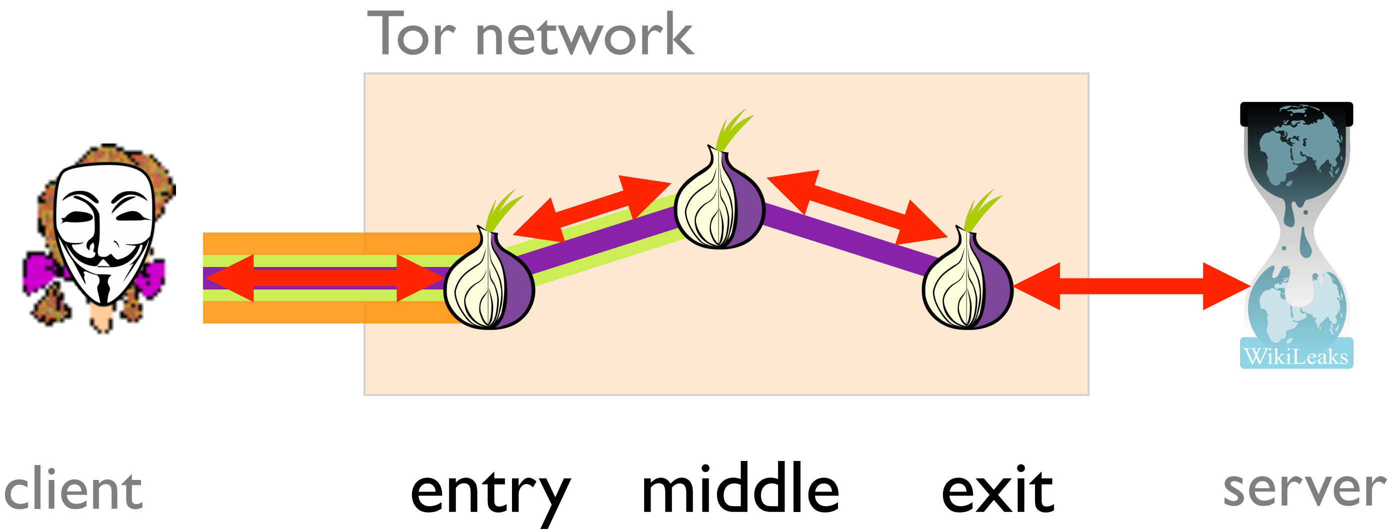
A client starts by selecting 3 relays, one of each type

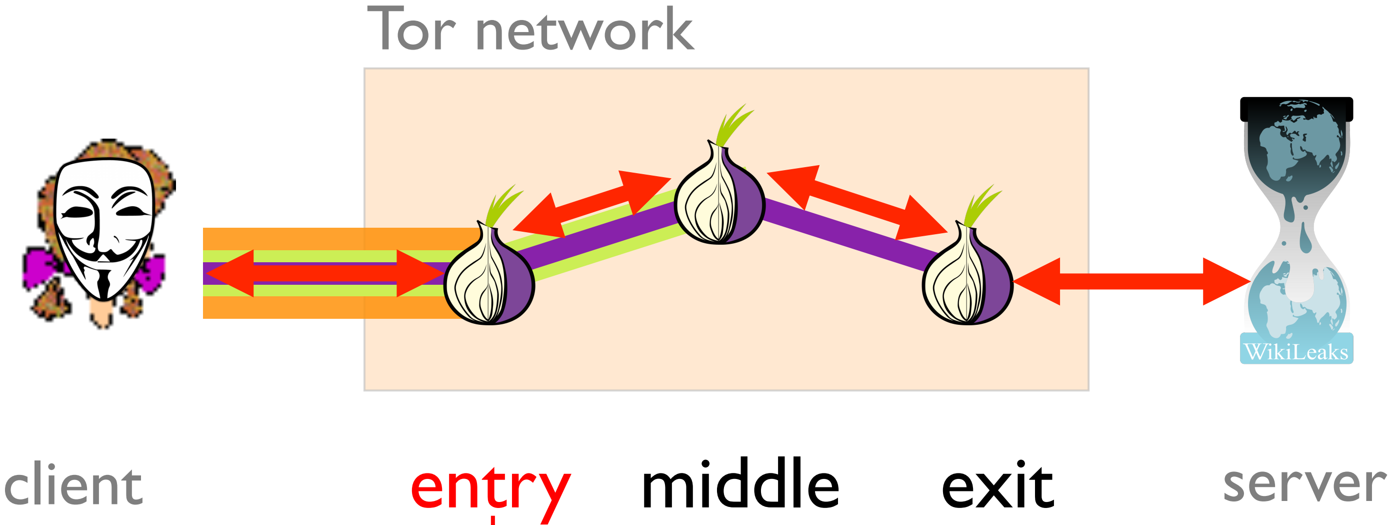


The client then *magically* builds an encrypted circuit through them



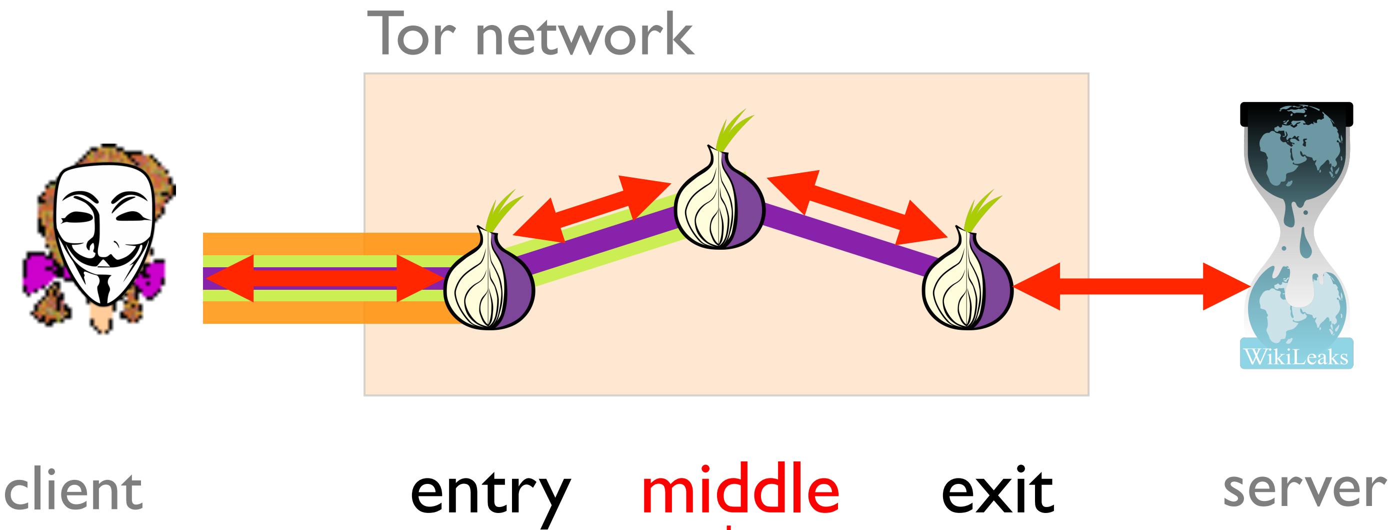
Not a single Tor node knows the client – server association





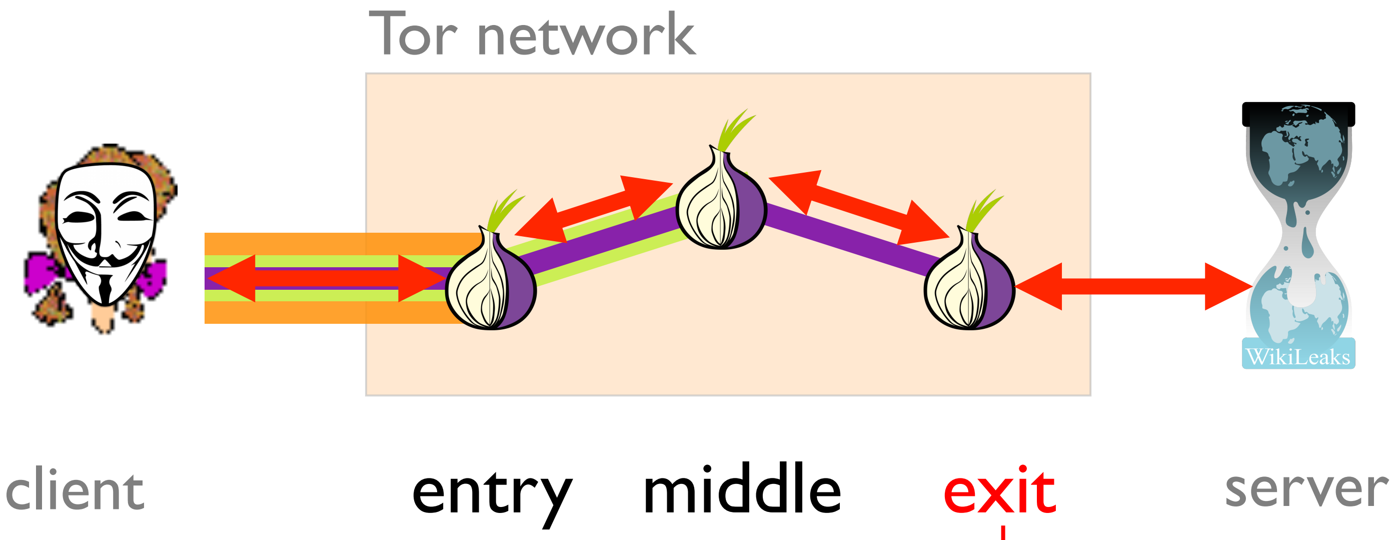
knows the source,

not the destination



knows neither the source,

nor the destination



Tor network

client

entry

middle

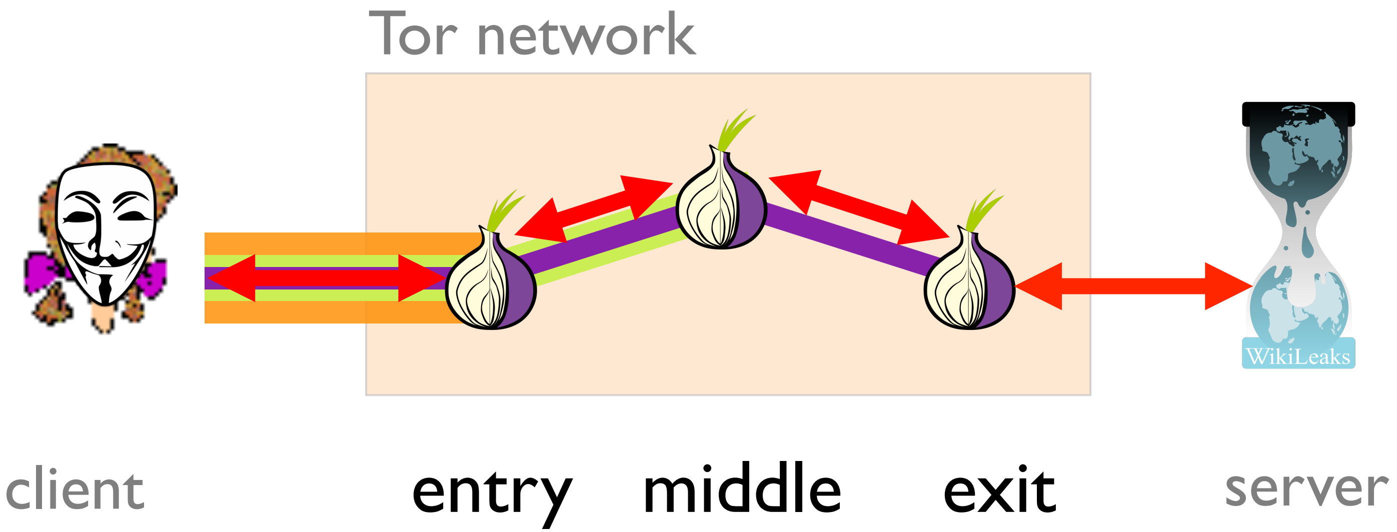
exit

server

knows the destination,

not the source

Anonymous communication takes place by forwarding traffic across consecutive tunnels



How exactly this encrypted circuit is built?

Outline

1. Intro
2. Network Primer
3. What is TOR
4. Security Primer
5. How Tor Works
6. Attacks against TOR

Transport Level Security (TLS) is a crypto protocol with three security properties

Connection is encrypted

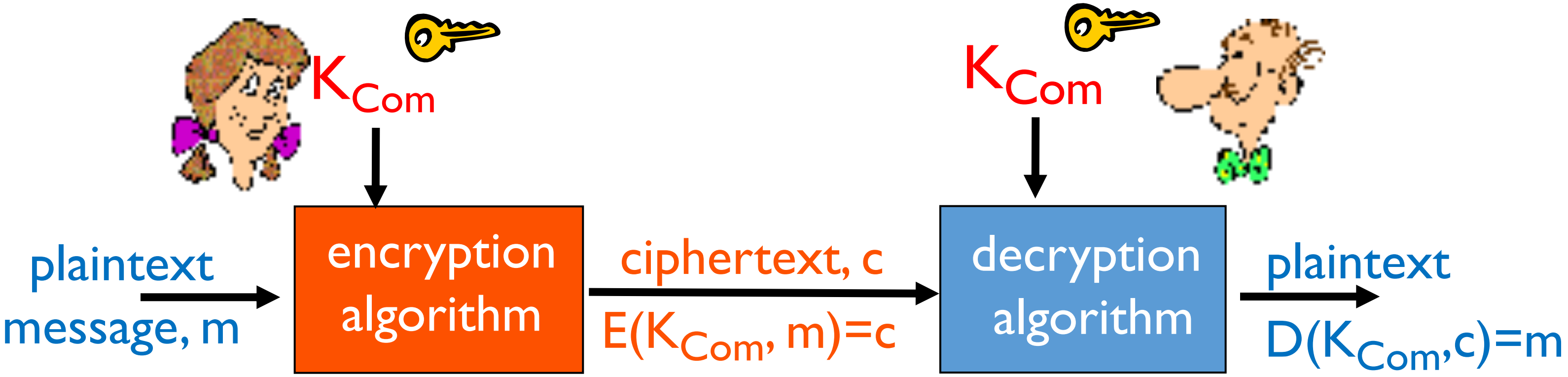
Entities can be authenticated

Messages can be validated

Widely used in web browsing, email, IM, and VoIP

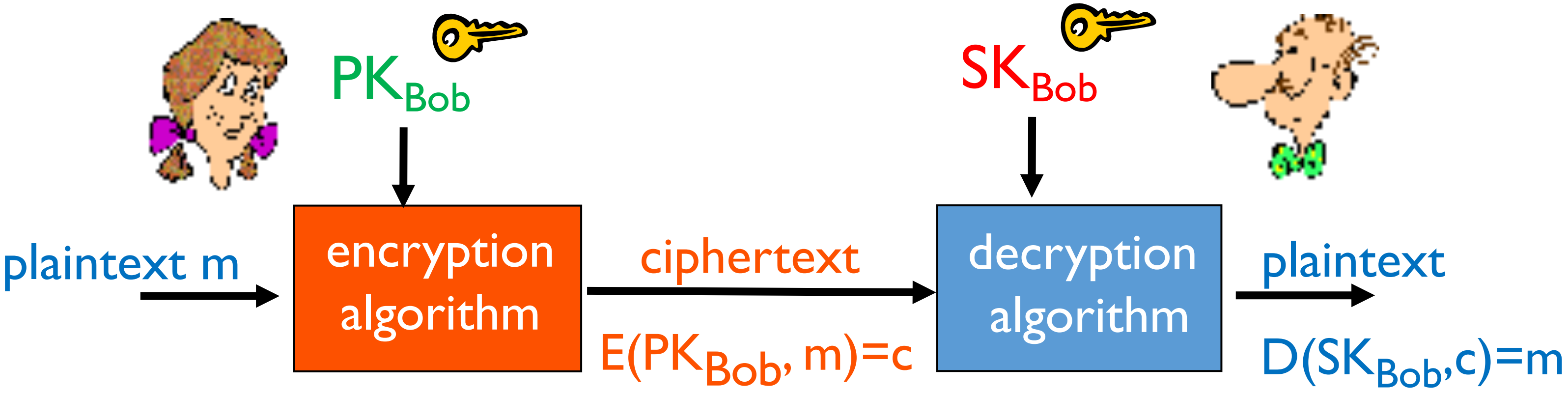
- HTTPS is an implementation of TLS on top of HTTP

TLS uses symmetric encryption



How do Alice and Bob establish the shared key?

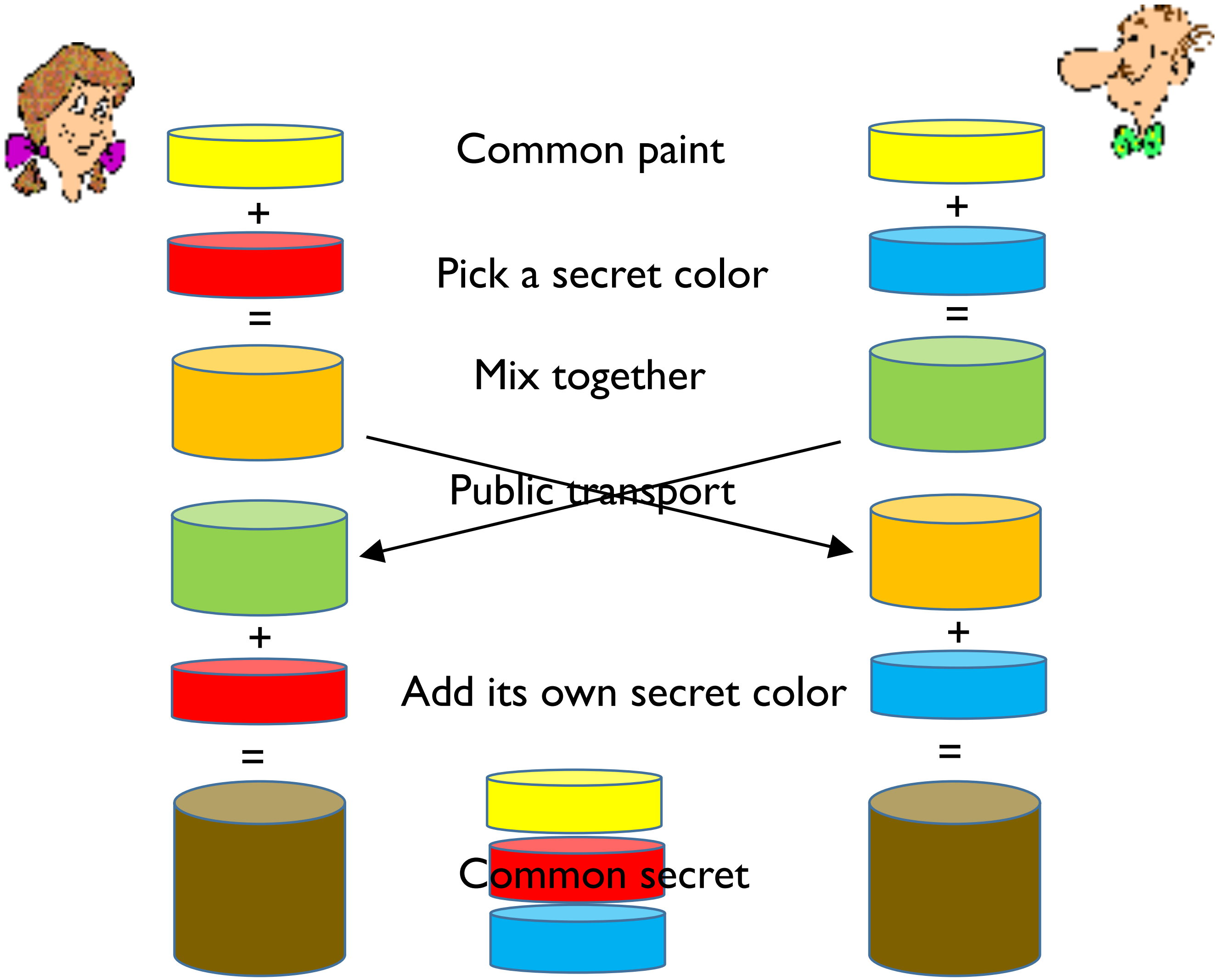
Public Key (aka asymmetric) Encryption



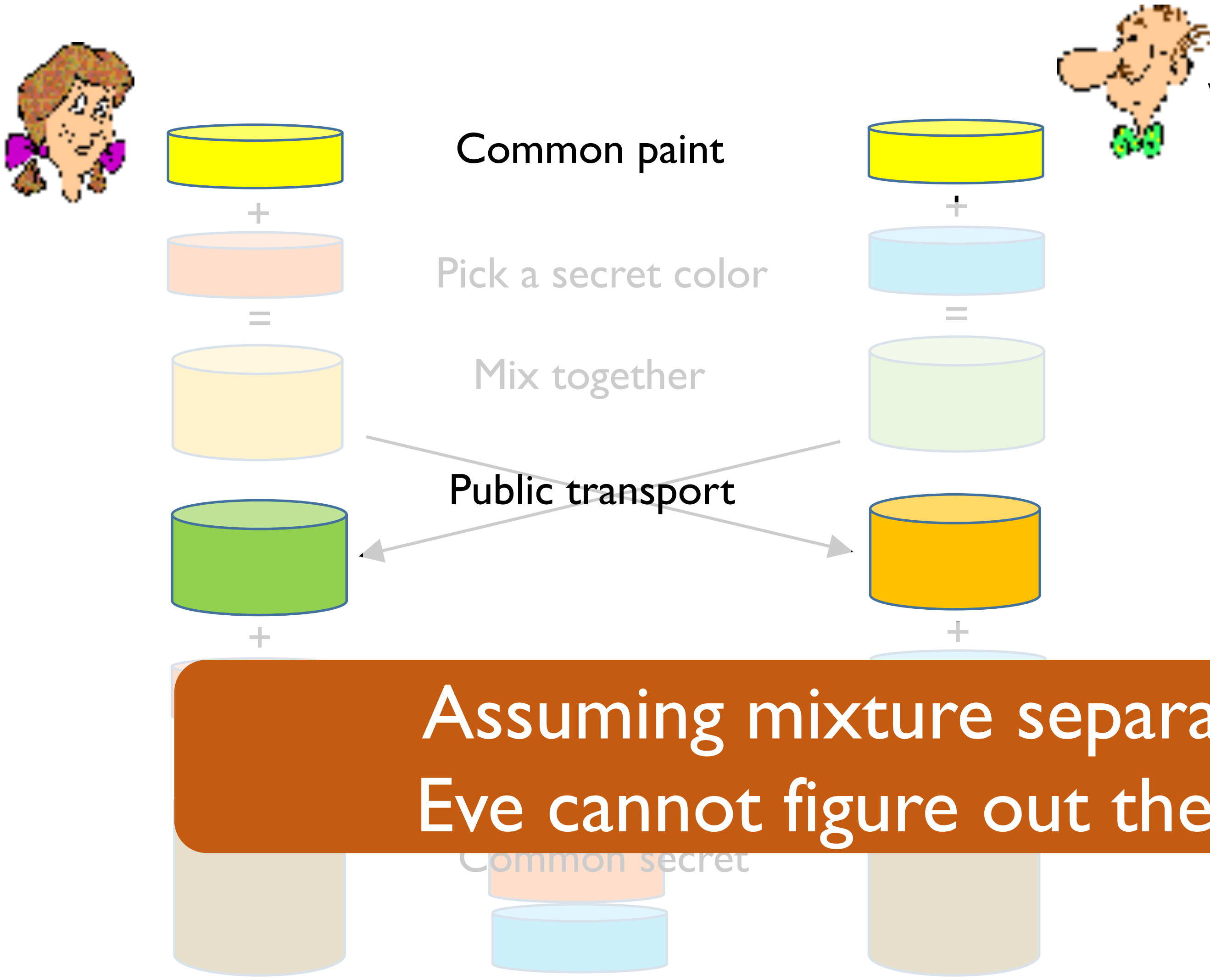
ex) RSA, Elliptic Curve, etc.

PK public key
SK private key

Key Exchange: Diffie-Hellman's Nifty Idea

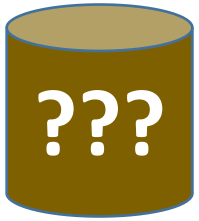


Key Exchange: Diffie-Hellman’s Nifty Idea



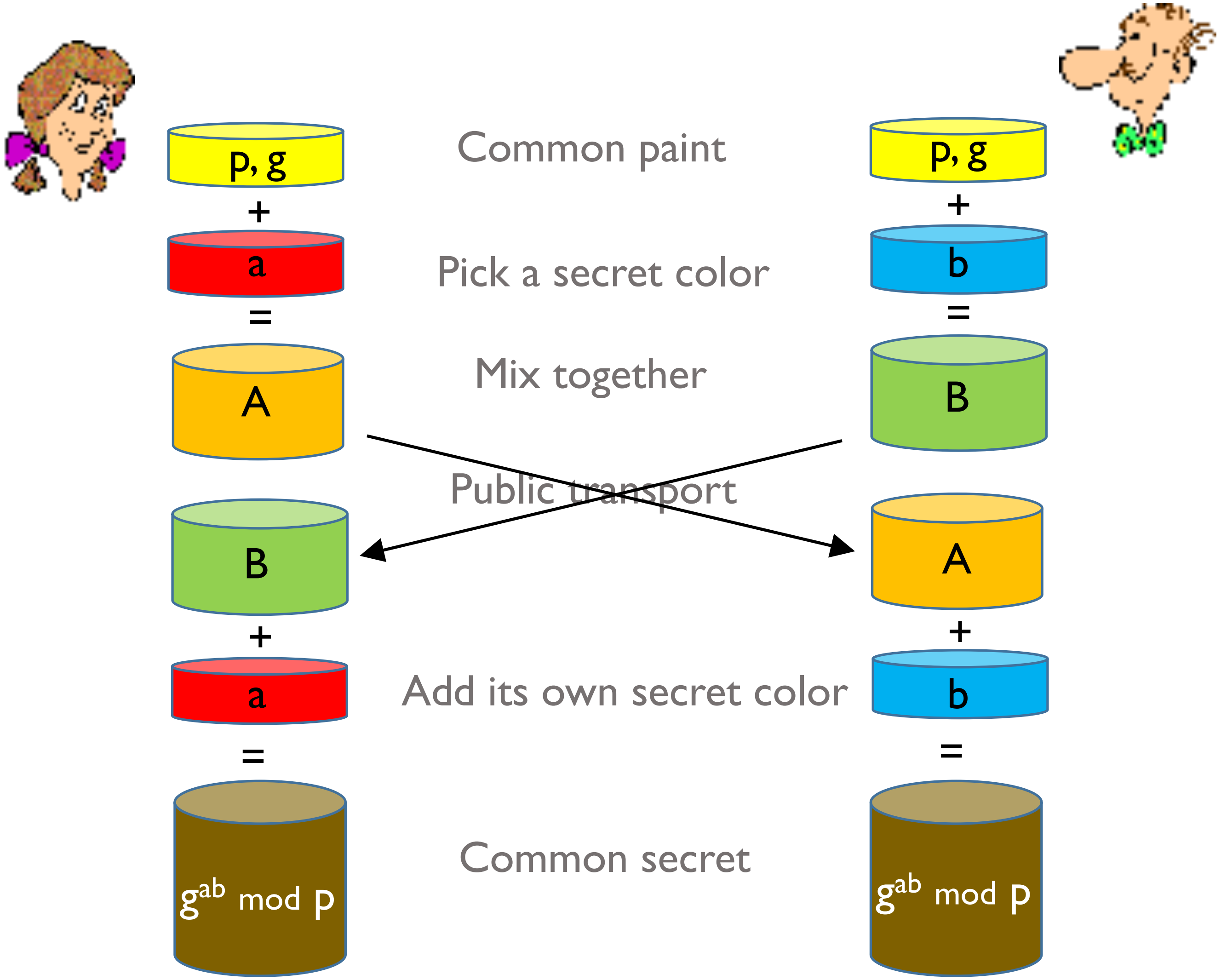
What Eve knows

- Common paint
- Mixtures transported



Assuming mixture separation is expensive
Eve cannot figure out the common secret!

Key Exchange: Diffie-Hellman's Nifty Idea



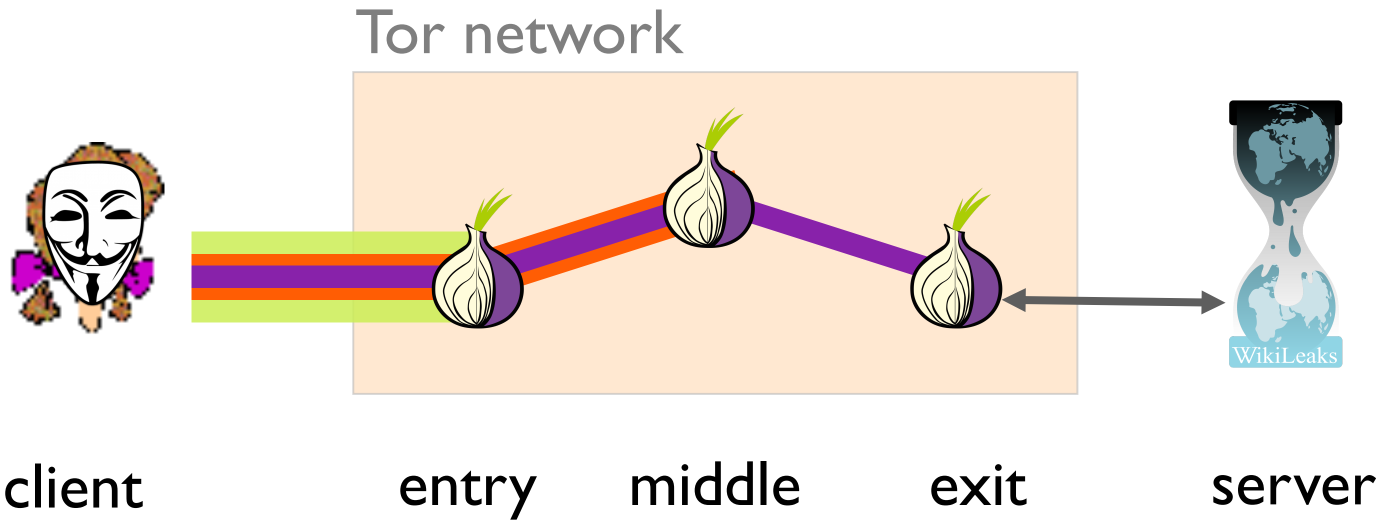
- p = a large prime
  g = a number [1.. p]
- a, b = random num [1..p-1]
- A = g^a mod p
  B = g^b mod p

- Alice computes B^a mod p
- Bob computes A^b mod p
- g^{ab} mod p is the shared key!

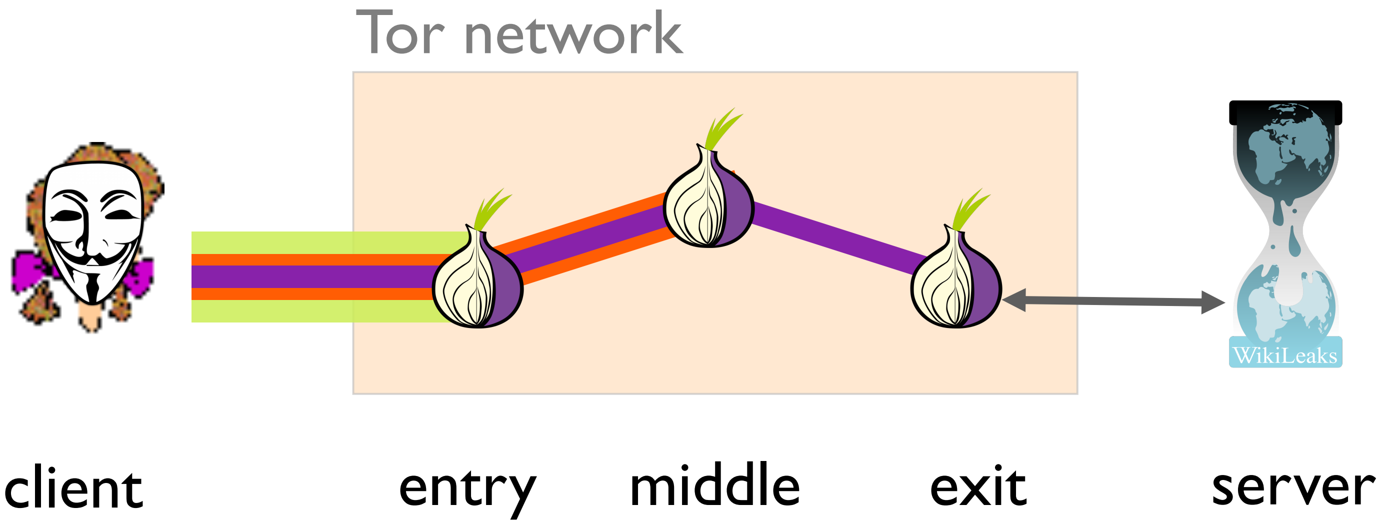
Outline

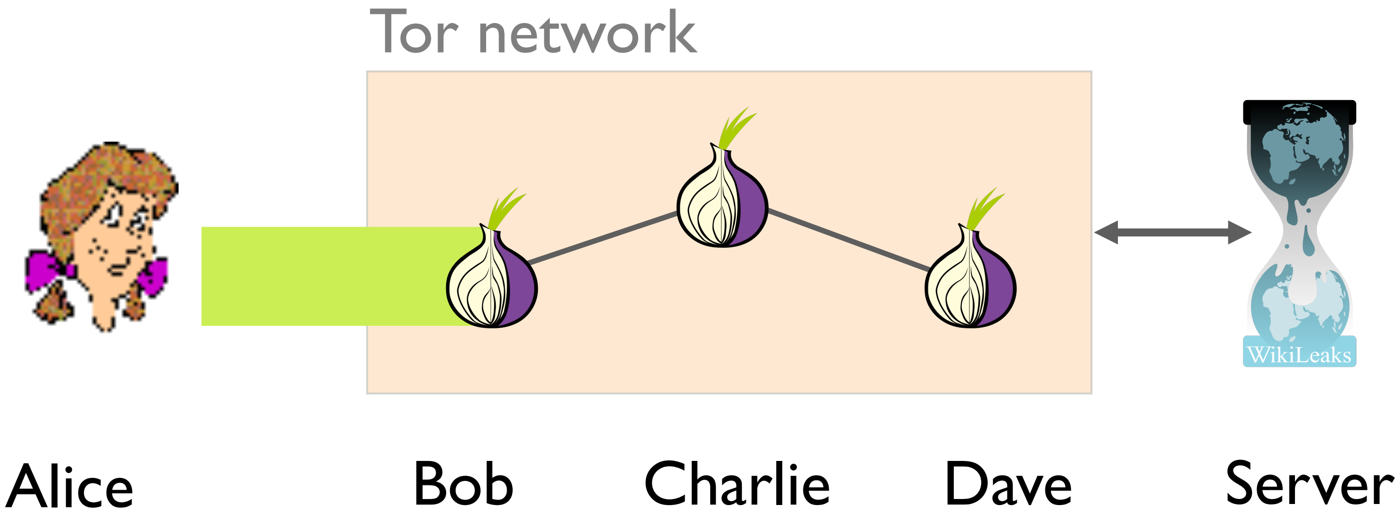
1. Intro
2. Network Primer
3. Security Primer
4. How Tor Works
5. Attacks and Censorship on Tor

How exactly this encrypted circuit is built?



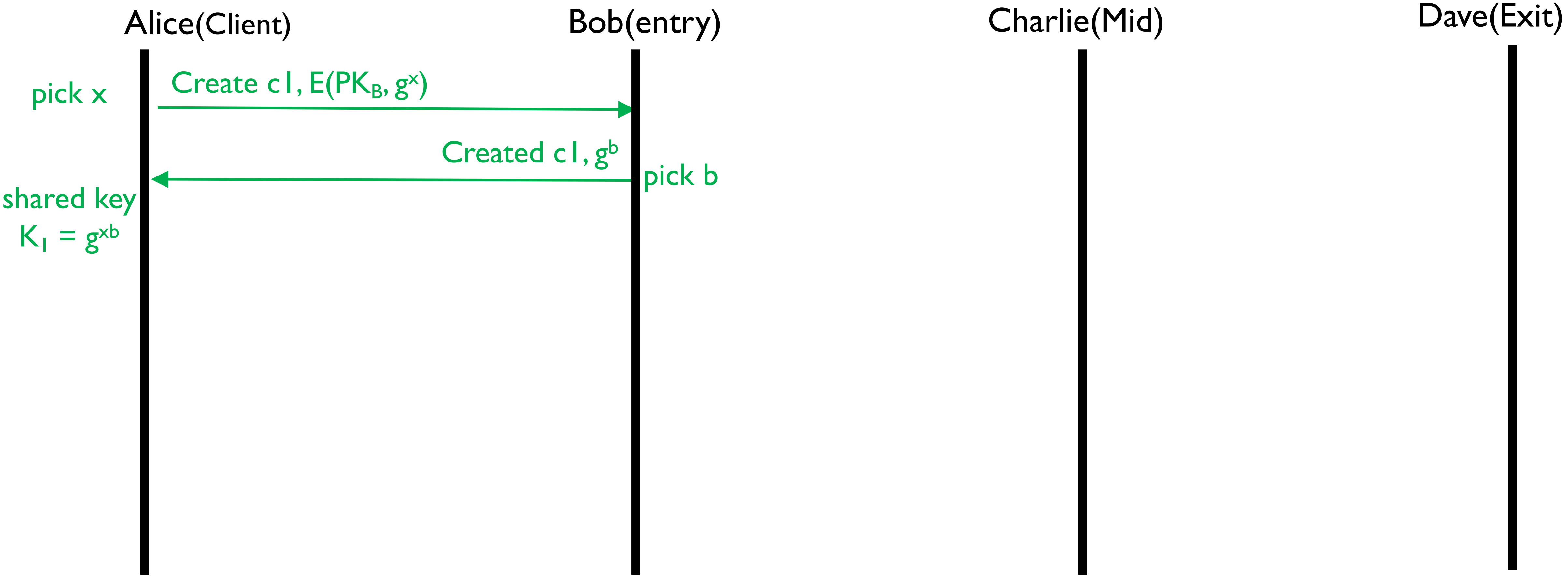
Goal:
Tor node should know only its previous and next hop




Tor network
WikiLeaks
Alice
Bob
Charlie
Dave
Server

Tor Circuit Construction: 1st hop

- How Alice – Bob establish shared session key K_1



Alice(Client)

Bob(entry)

Charlie(Mid)

Dave(Exit)

pick x

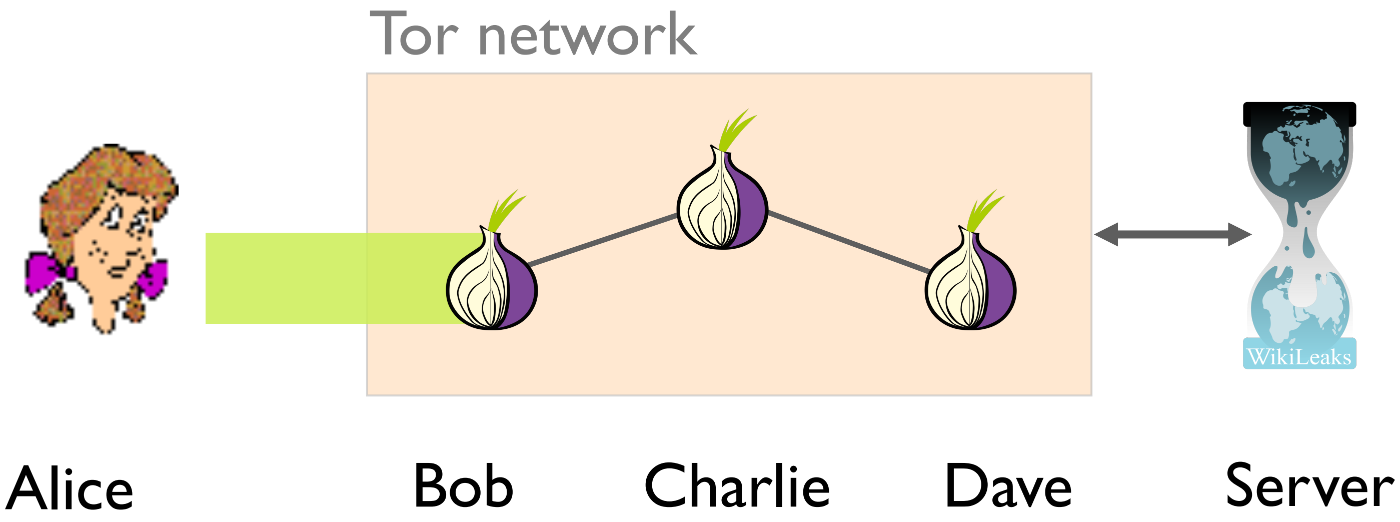
Create c1, $E(PK_B, g^x)$

Created c1, g^b

pick b

shared key

$K_1 = g^{xb}$



Tor network

Alice

Bob

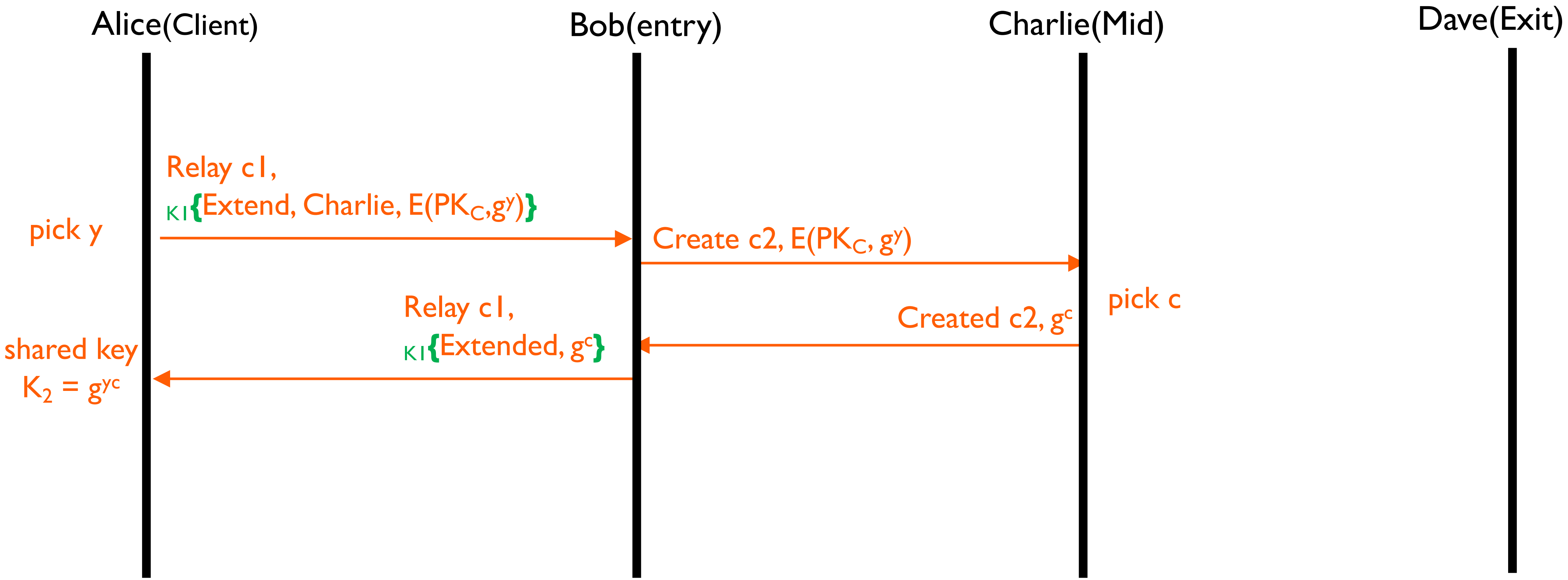
Charlie

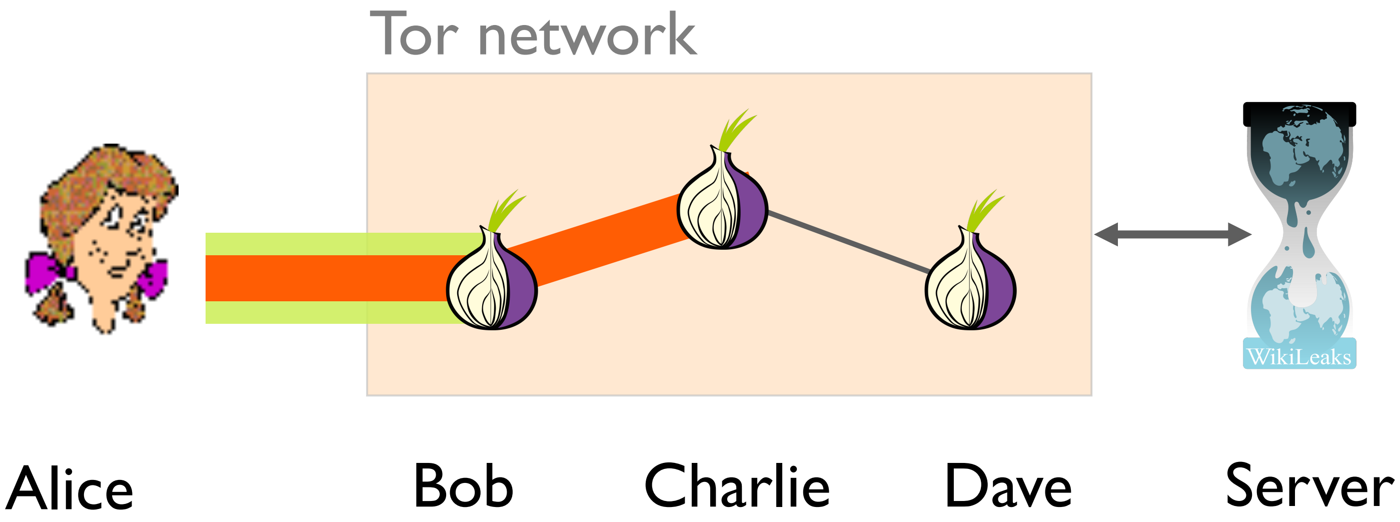
Dave

Server

Tor Circuit Construction: 2nd hop

- How Alice – Charlie establish shared session key K_2





Tor network

Alice

Bob

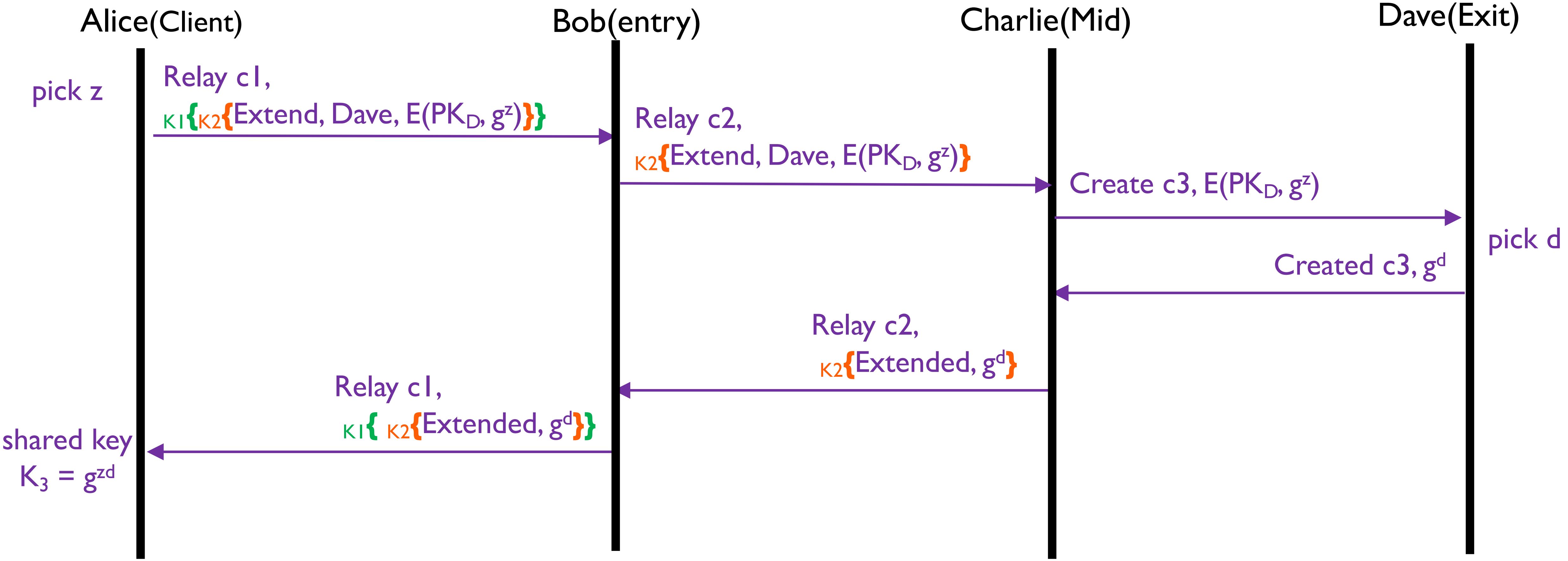
Charlie

Dave

Server

Tor Circuit Construction: 3rd hop

- How Alice – Dave establish shared session key K_3



Alice(Client)

Bob(entry)

Charlie(Mid)

Dave(Exit)

pick z

Relay c1, ${}_{K1}\{{}_{K2}\{\text{Extend, Dave, } E(PK_D, g^z)\}\}$

Relay c2, ${}_{K2}\{\text{Extend, Dave, } E(PK_D, g^z)\}$

Create c3, $E(PK_D, g^z)$

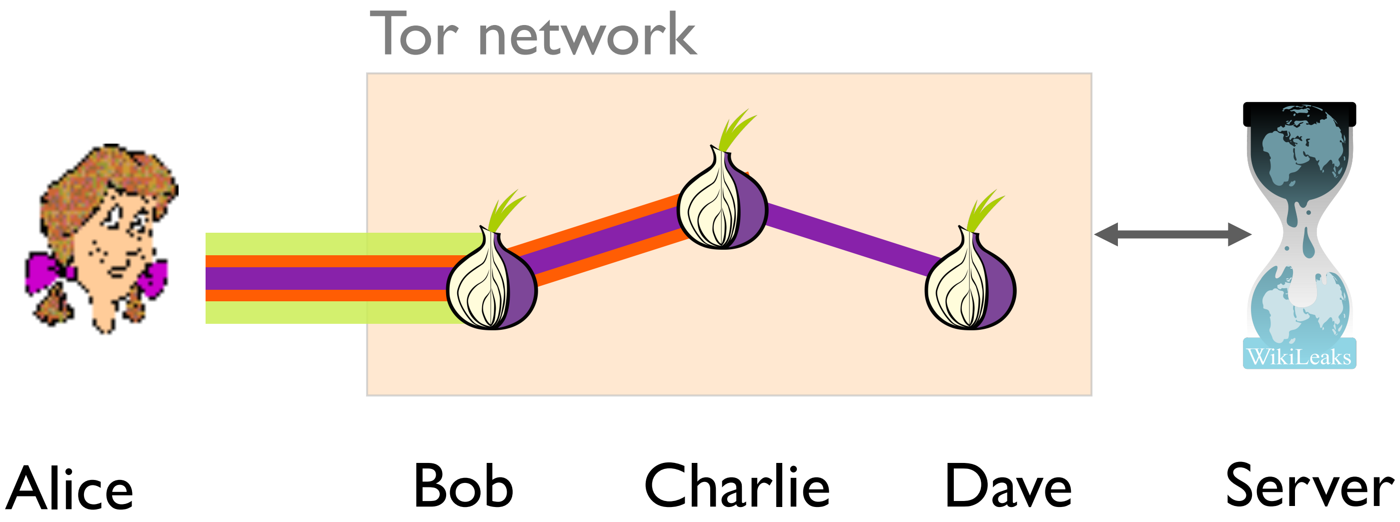
pick d

Created c3, g^d

Relay c2, ${}_{K2}\{\text{Extended, } g^d\}$

Relay c1, ${}_{K1}\{\ {}_{K2}\{\text{Extended, } g^d\}\}$

shared key $K_3 = g^{zd}$



Tor network

Alice

Bob

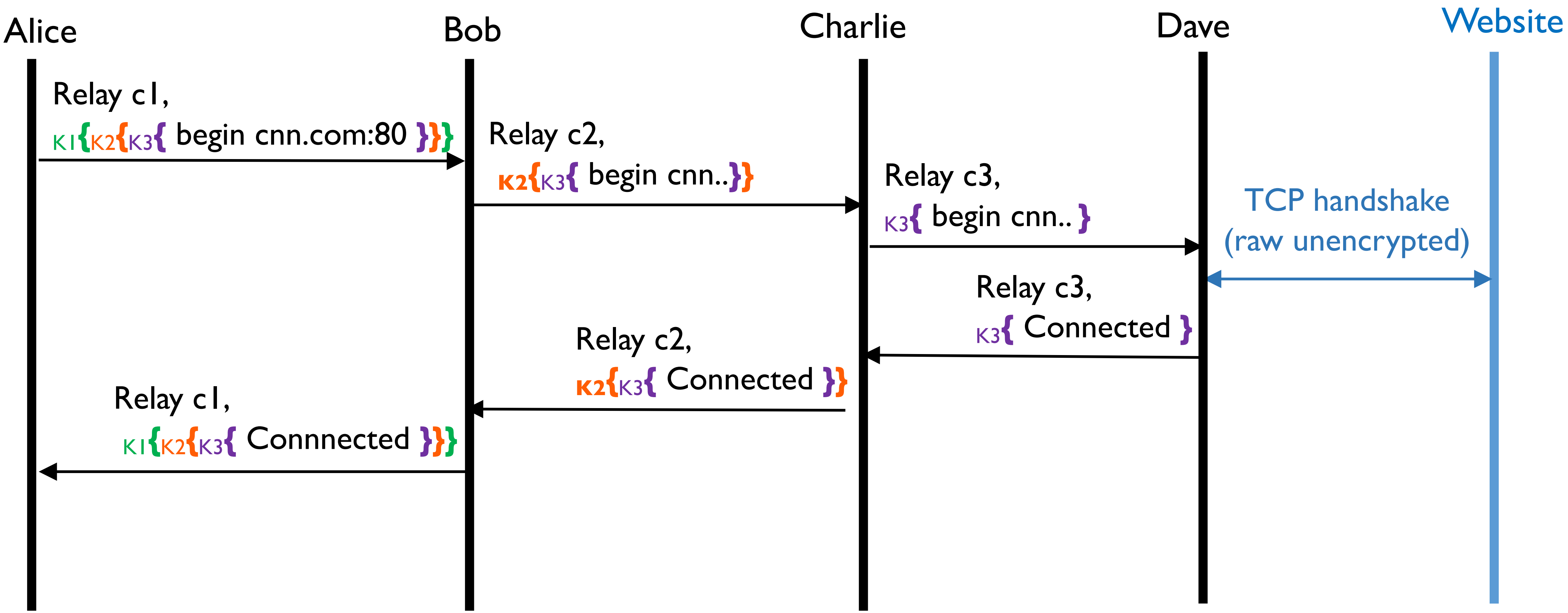
Charlie

Dave

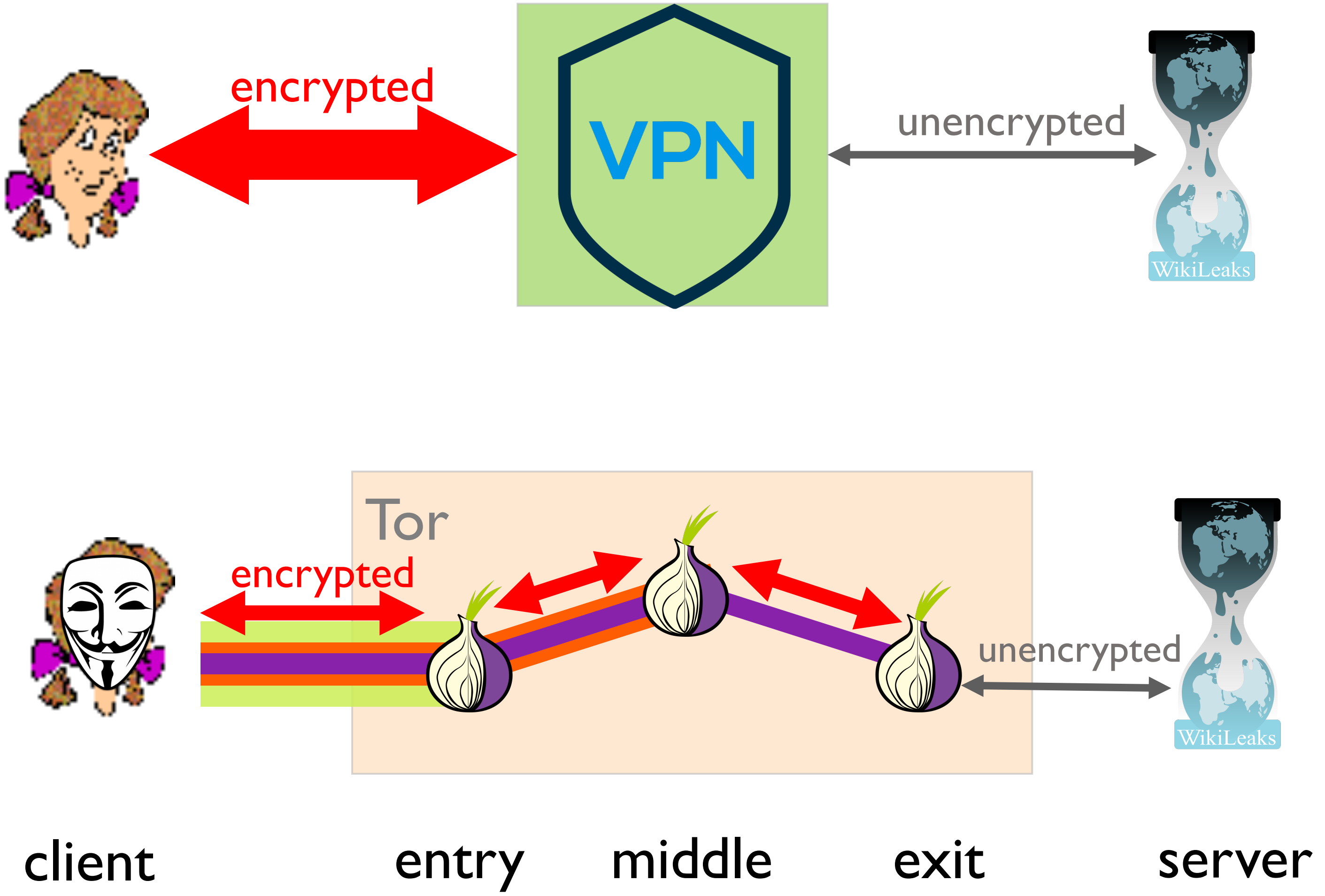
Server

Tor Packet Forwarding via 3 hop Circuit

- Alice – Bob, Alice – Charlie, Alice – Dave has shared session key K_1, K_2 and K_3



VPN vs Tor

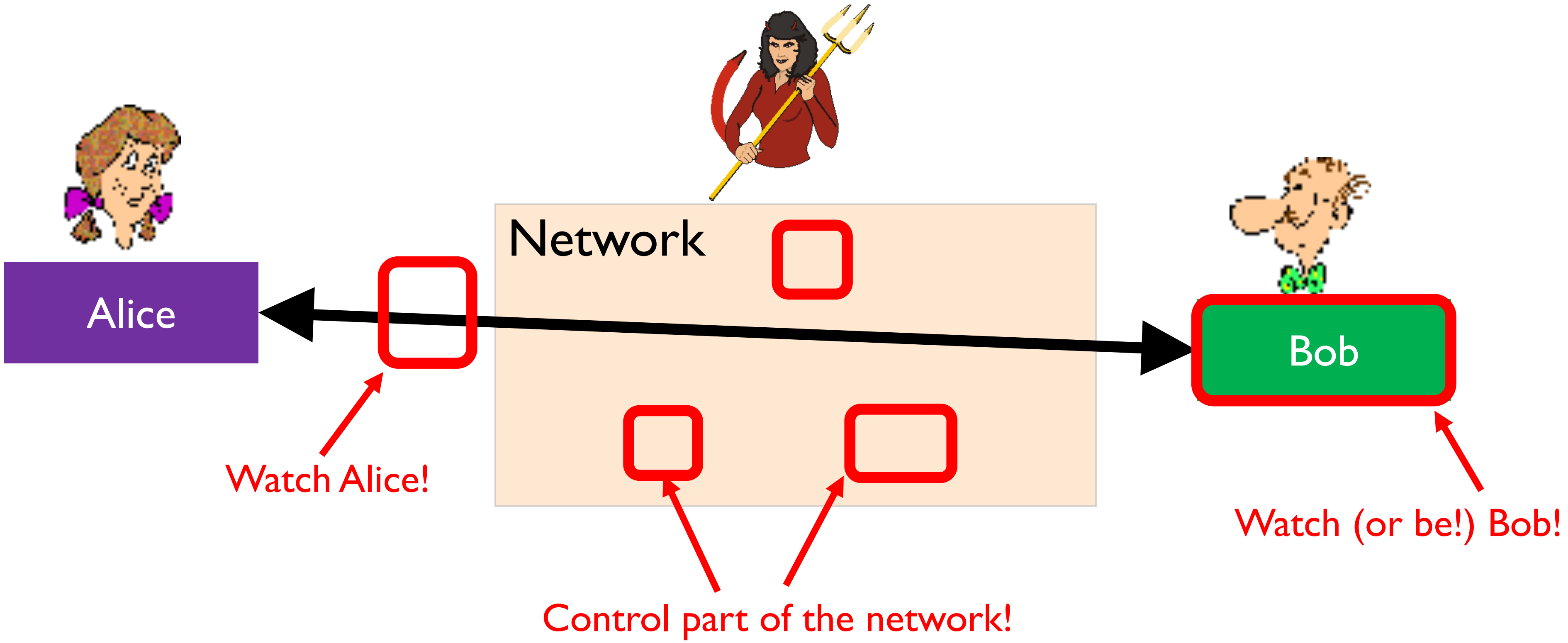

encrypted
VPN
unencrypted
WikiLeaks
Tor
encrypted
unencrypted
WikiLeaks
client
entry
middle
exit
server

Outline

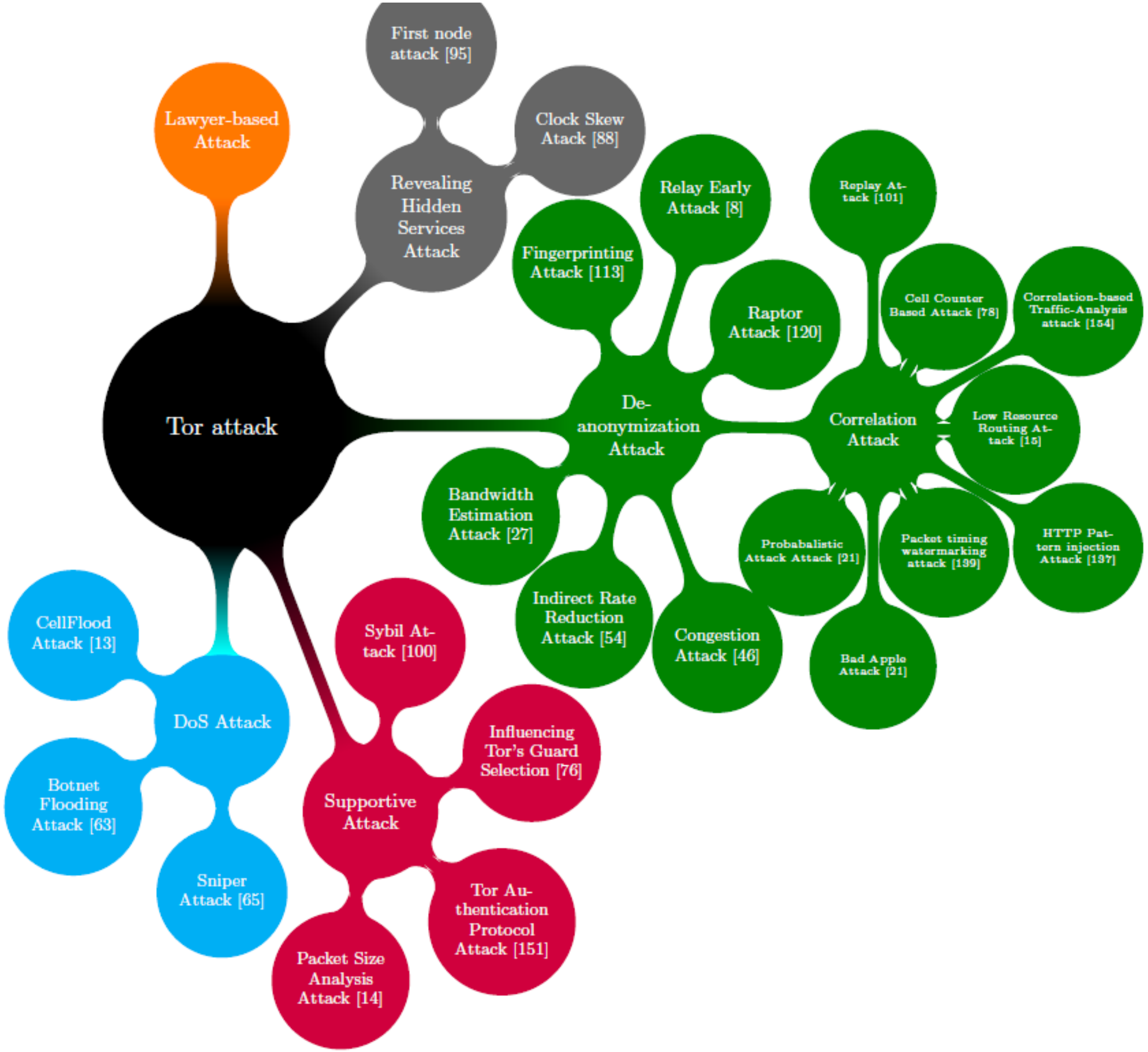
1. Intro
2. Network Primer
3. Security Primer
4. How Tor Works
5. 🤘 **Attacks and Censorship on Tor**
    1. 🤘 Attacks against Tor
    2. Censorship on Tor

WE ARE NOT SO
ANONYMOUS

Threat model: what can adversaries do?



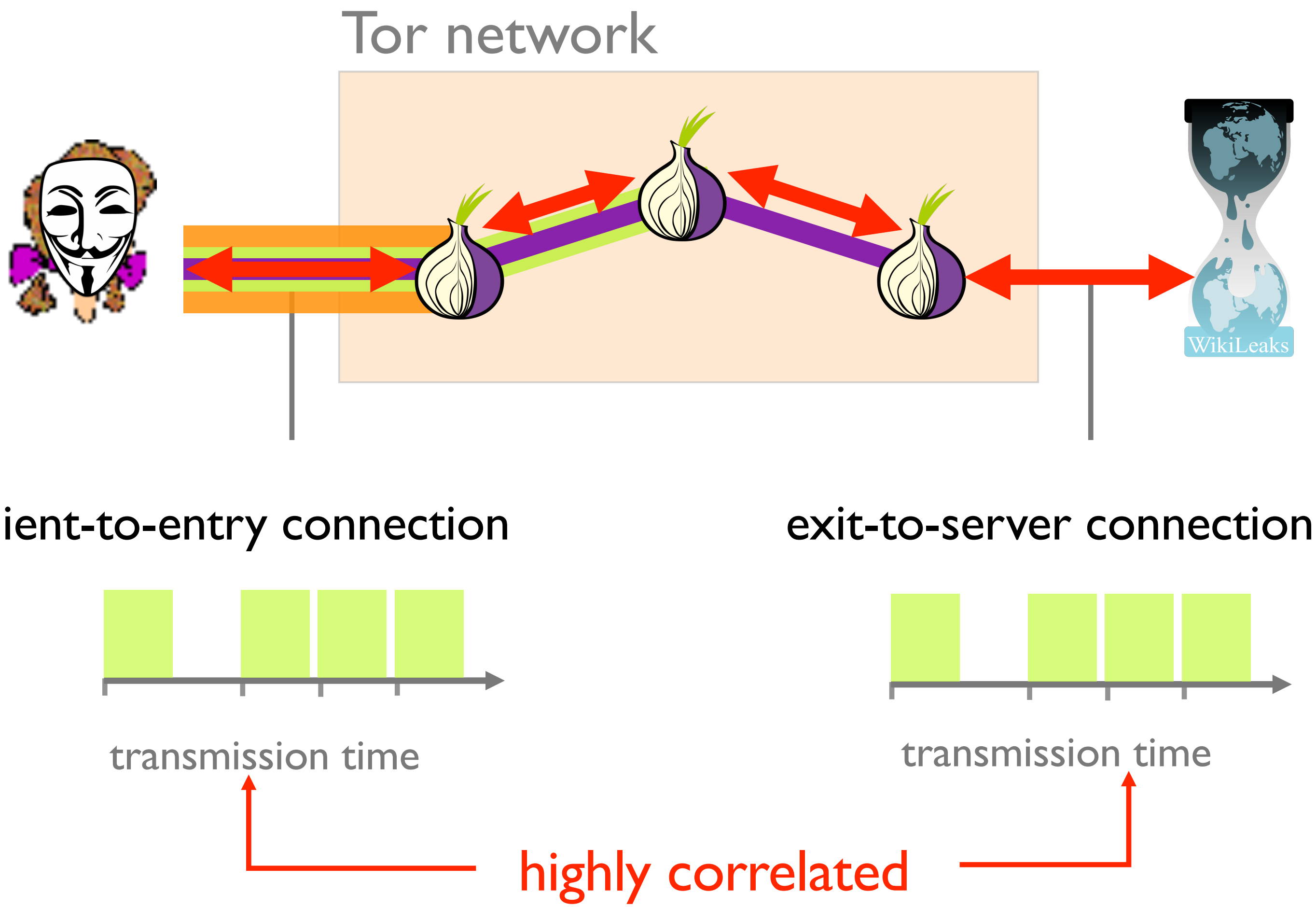
Tor is vulnerable to various types of attacks


Tor attack
Lawyer-based Attack
Revealing Hidden Services Attack
First node attack [95]
Clock Skew Atack [88]
De-anonymization Attack
Fingerprinting Attack [113]
Relay Early Attack [8]
Raptor Attack [120]
Bandwidth Estimation Attack [27]
Indirect Rate Reduction Attack [54]
Congestion Attack [46]
Correlation Attack
Replay Attack [101]
Cell Counter Based Attack [78]
Correlation-based Traffic-Analysis attack [154]
Low Resource Routing Attack [15]
HTTP Pattern injection Attack [137]
Packet timing watermarking attack [139]
Probabalistic Attack Attack [21]
Bad Apple Attack [21]
DoS Attack
CellFlood Attack [13]
Botnet Flooding Attack [63]
Sniper Attack [65]
Supportive Attack
Sybil Attack [100]
Influencing Tor's Guard Selection [76]
Tor Authentication Protocol Attack [151]
Packet Size Analysis Attack [14]

Traffic Correlation Attack

RAPTOR: Routing Attacks on Privacy in Tor (USENIX Sec'15)

Traffic entering and leaving Tor is highly correlated



Tor network

client-to-entry connection

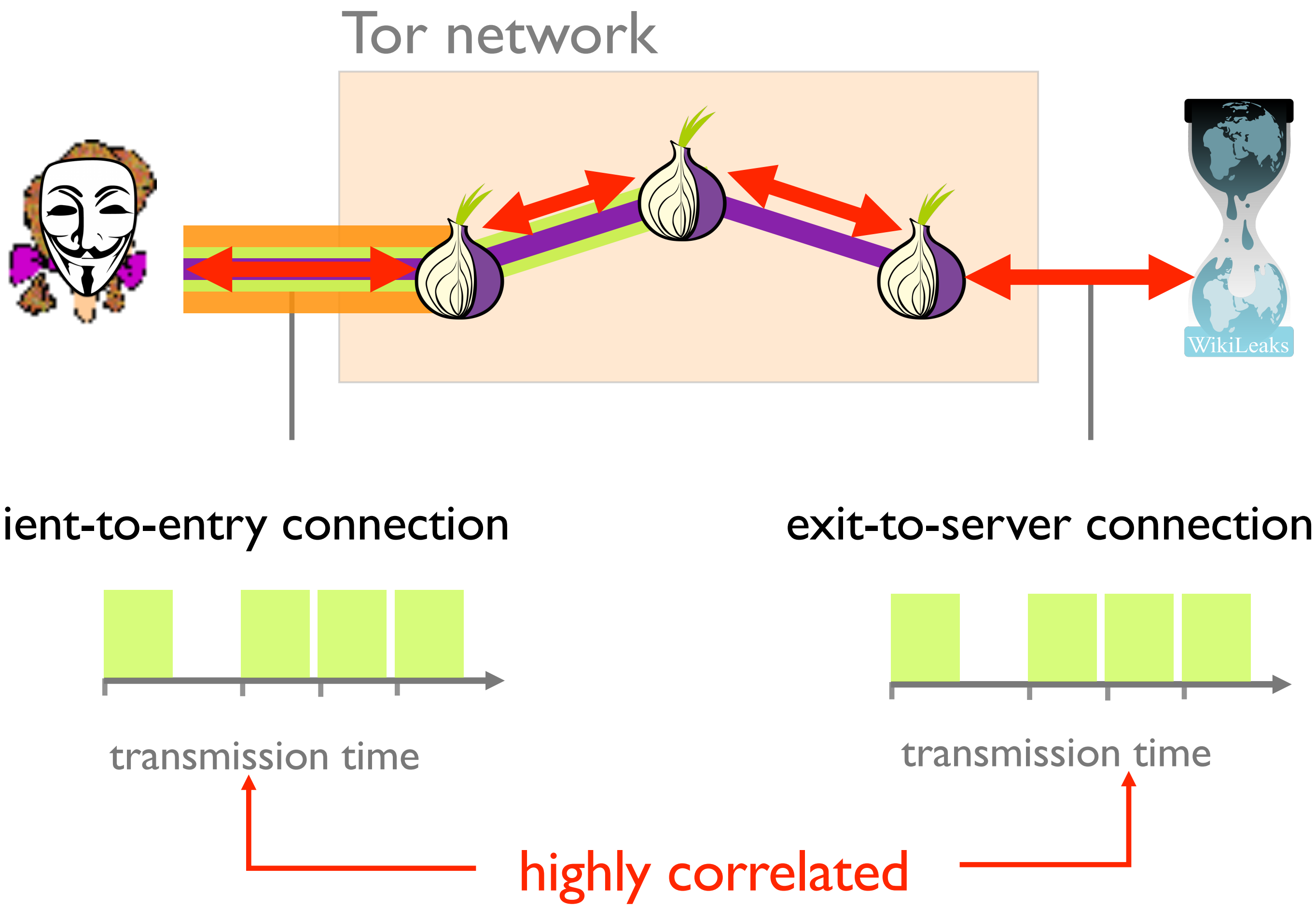
exit-to-server connection

transmission time

transmission time

highly correlated

Such traffic correlation attacks require to see client-to-entry and server-to-exit traffic



Tor network

client-to-entry connection

exit-to-server connection

transmission time

transmission time

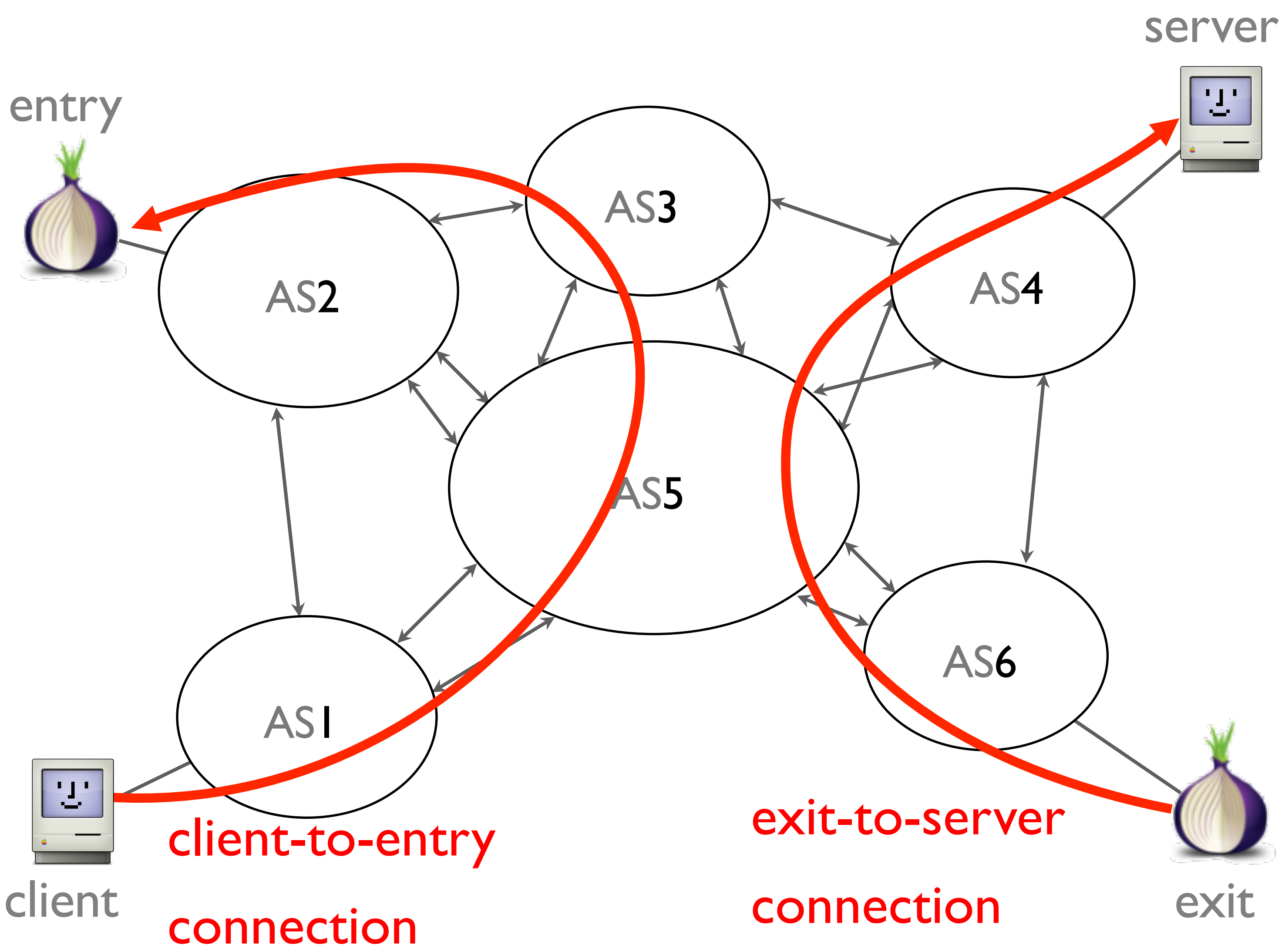
highly correlated

There are two ways to see client-to-entry and server-to-exit traffic

Own entry and exit
malicious relays

Own the links
malicious networks

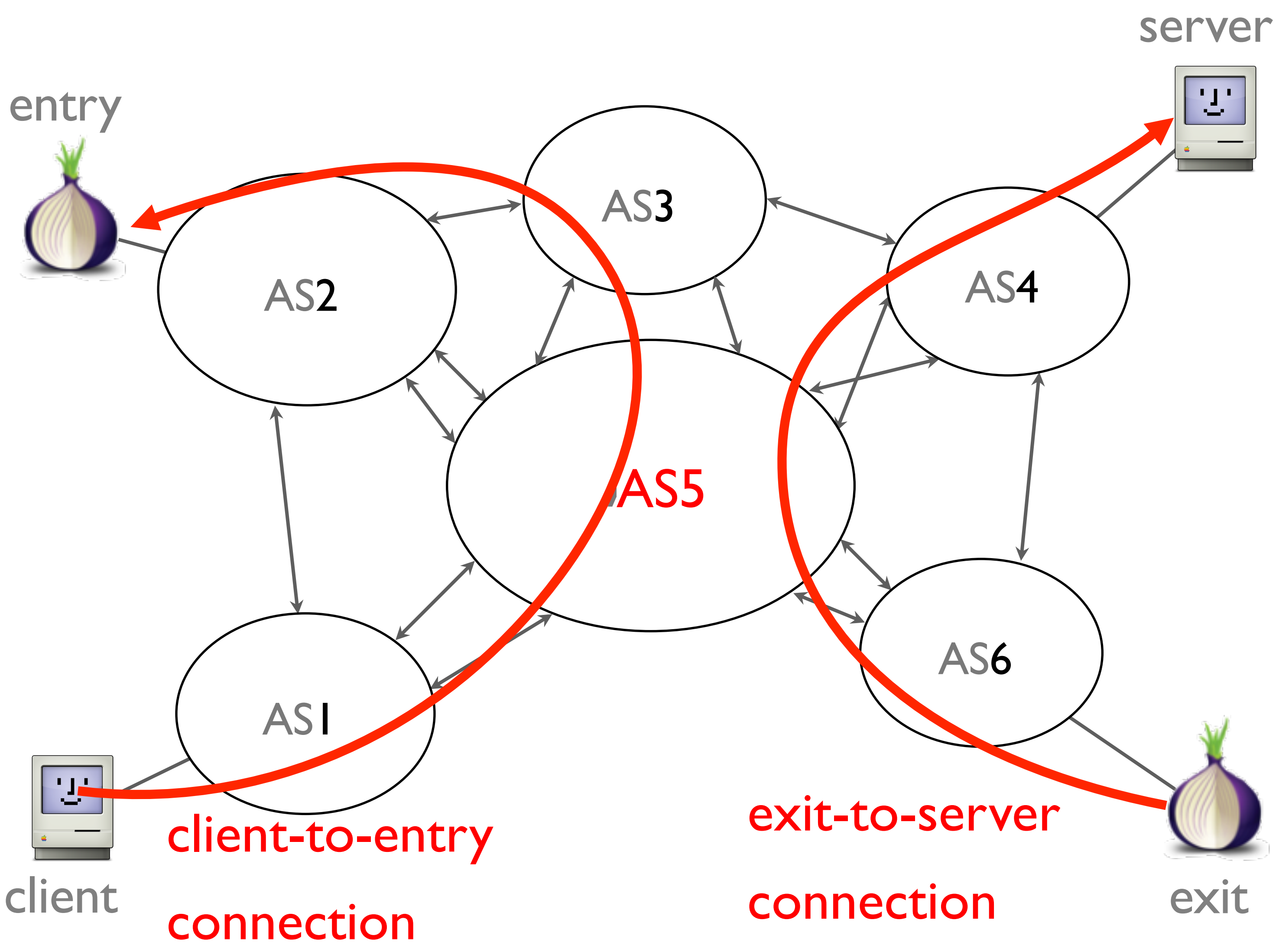
Tor connections get routed according to BGP



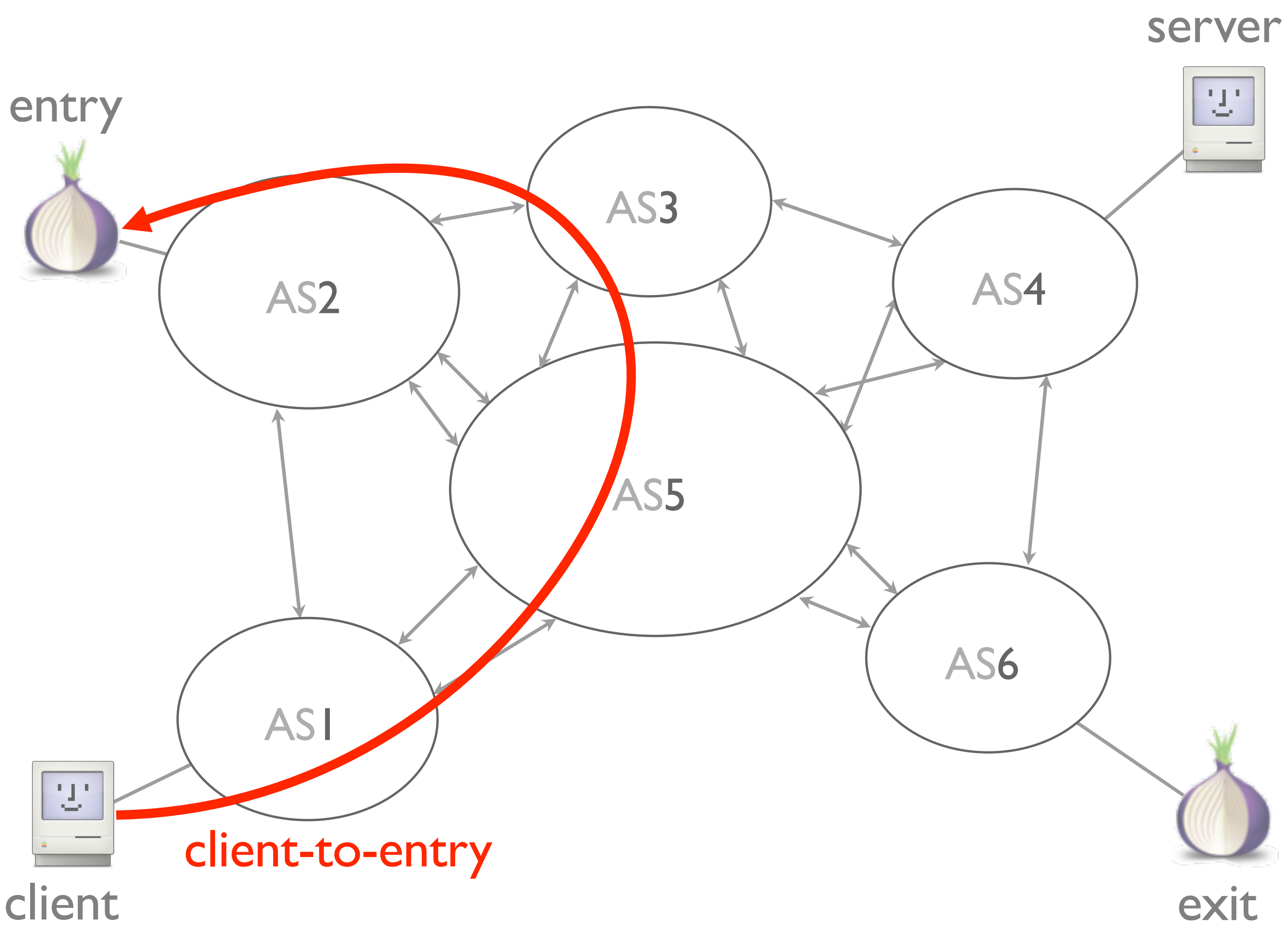
client-to-entry connection

exit-to-server connection

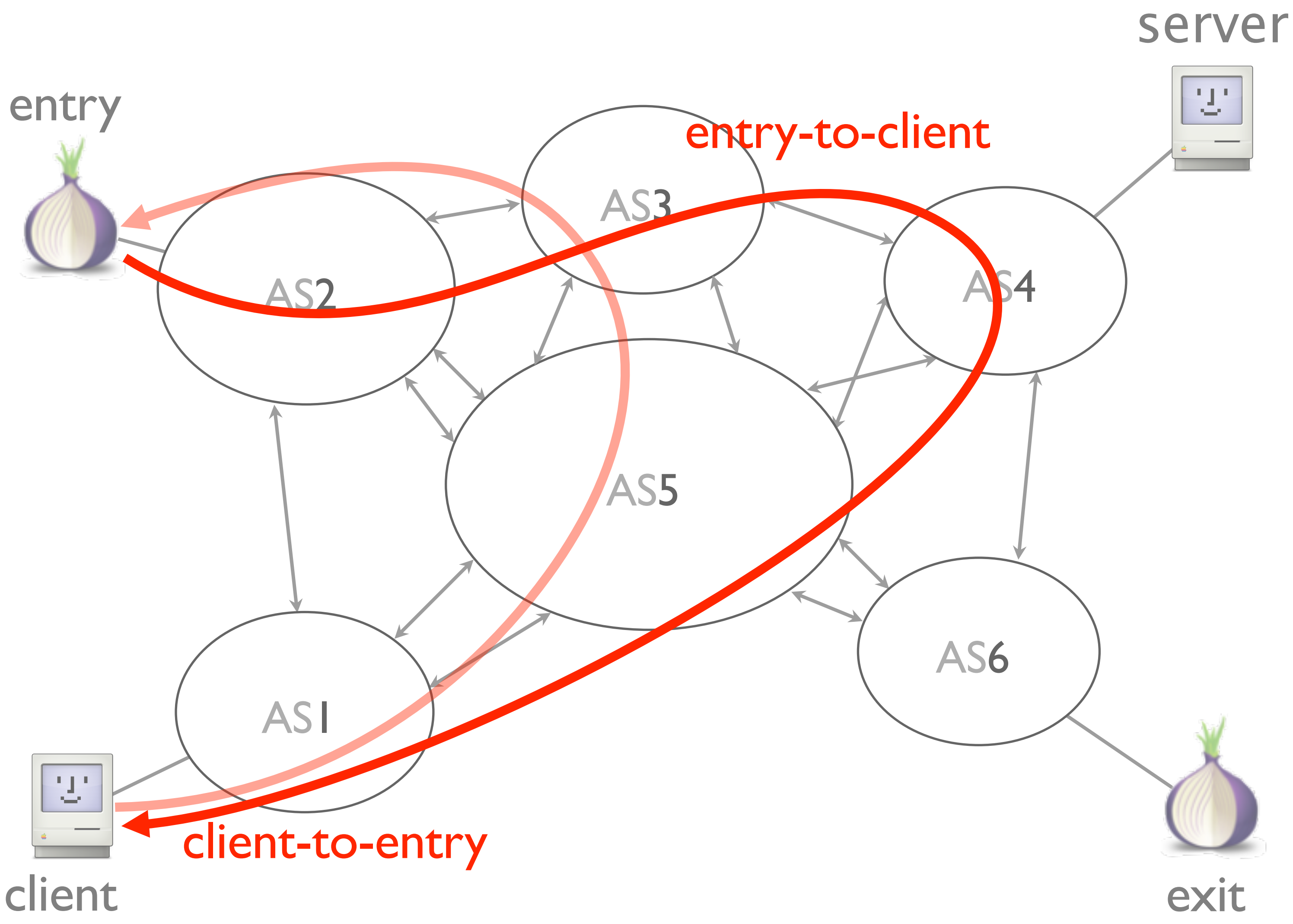
Who is able to perform traffic correlation?



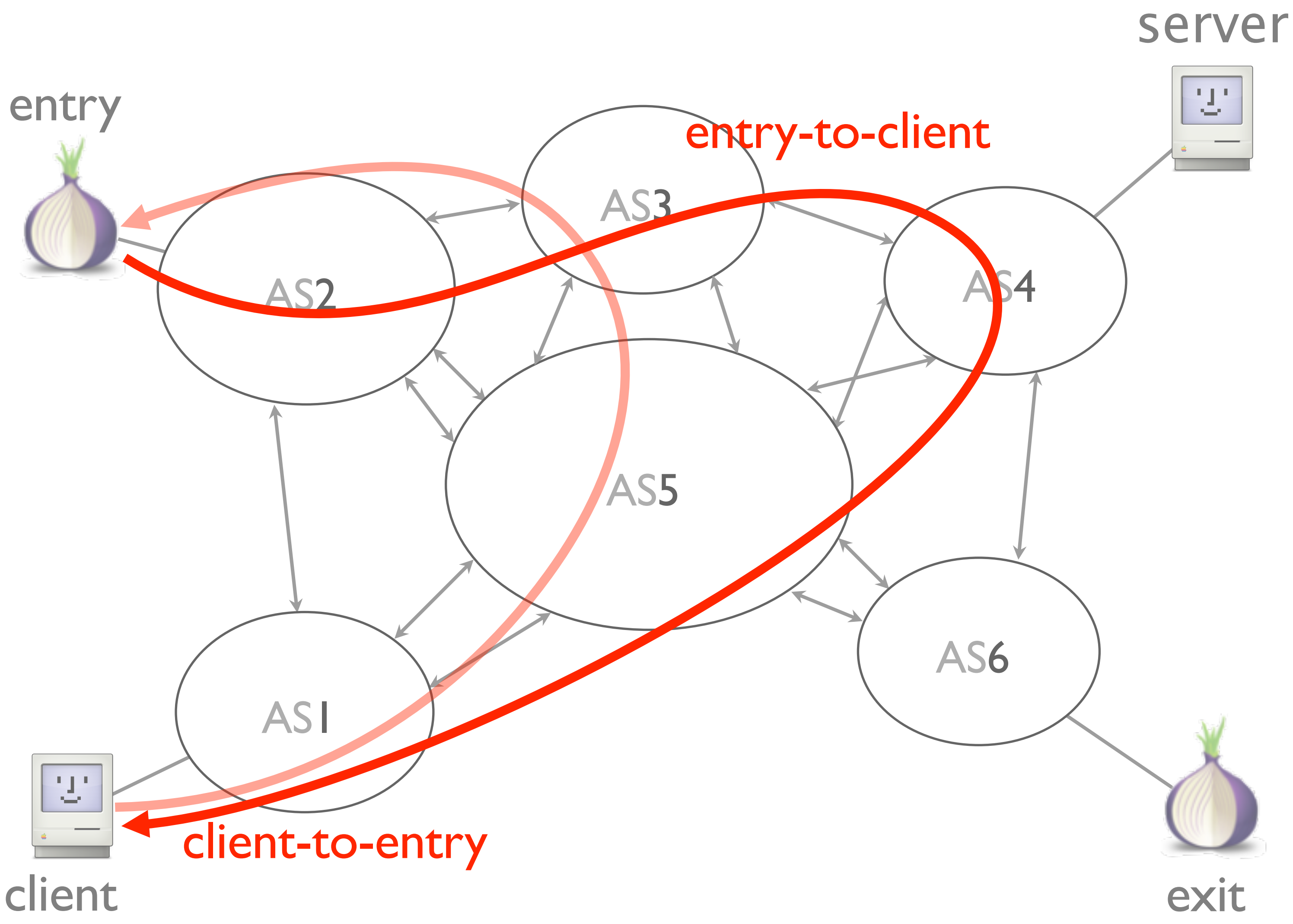
However, because of policies, routing is often asymmetric in BGP



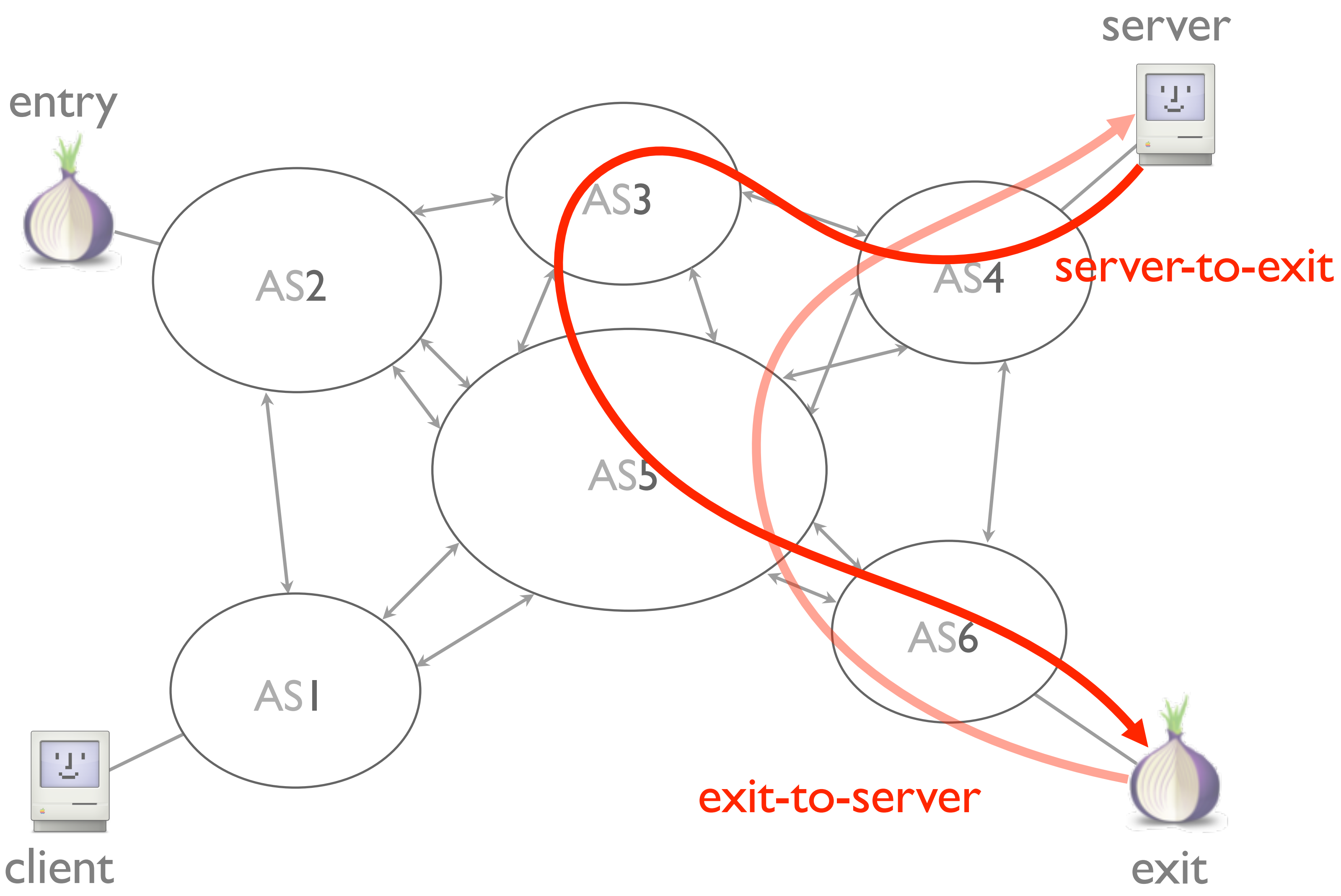
However, because of policies, routing is often asymmetric



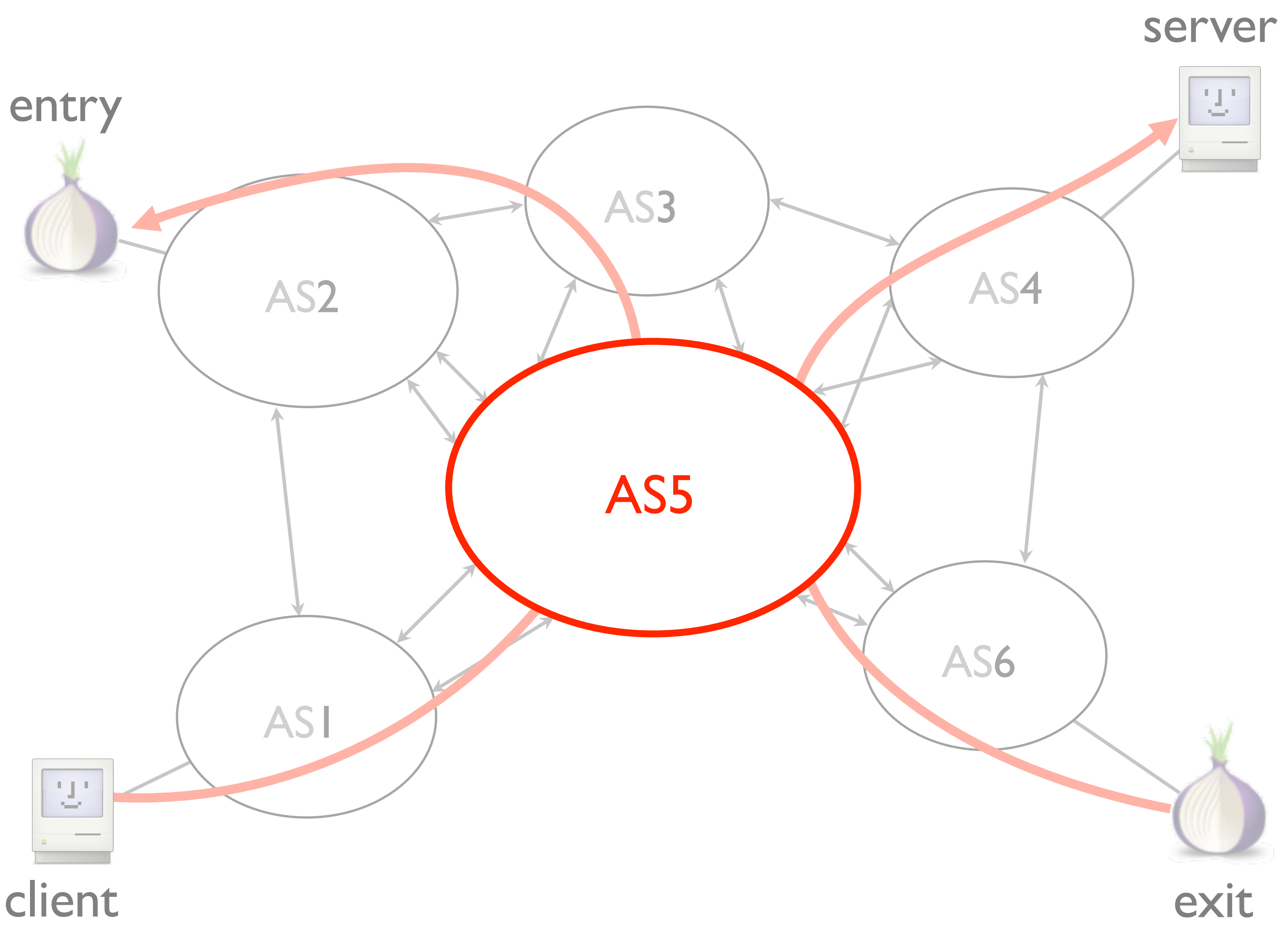
While AS4 does not see client-to-entry traffic, it sees entry-to-client traffic



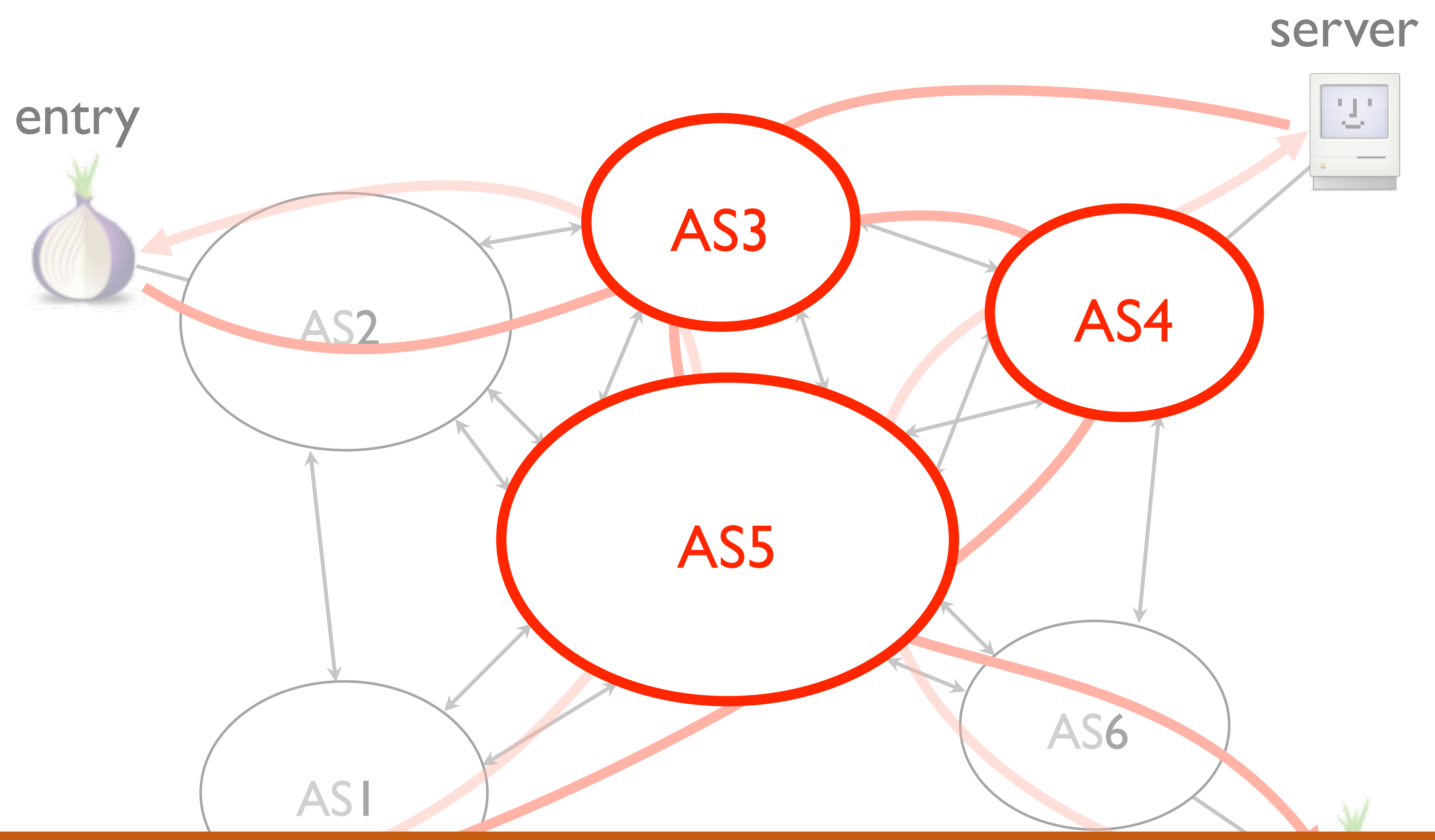
The same applies to server-to-exit traffic



Considering only one direction, only AS5 is potentially compromising



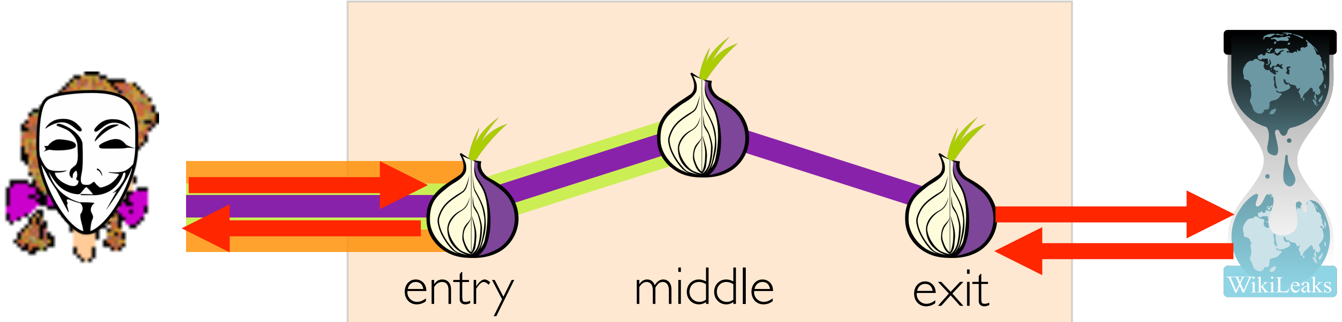
Considering both direction, AS3, AS4, and AS5 are potentially compromising

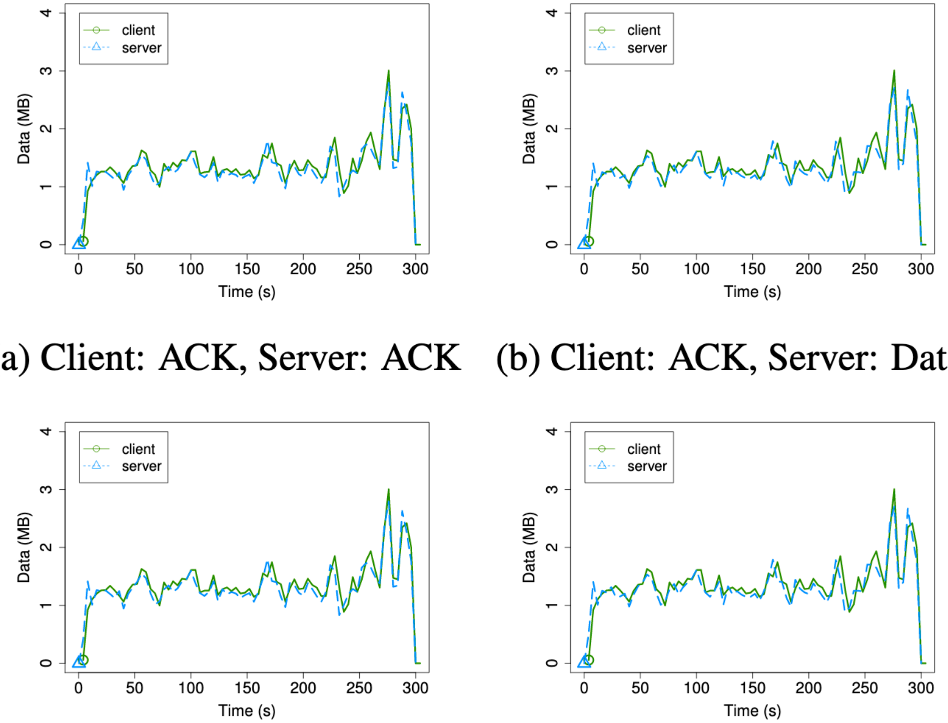


Asymmetric nature of BGP routing increases the numbers of AS-level adversaries

In terms of timing properties,
TCP data and ack packets are highly correlated

Observing any direction of the traffic at both ends is enough to deanonymize Tor users



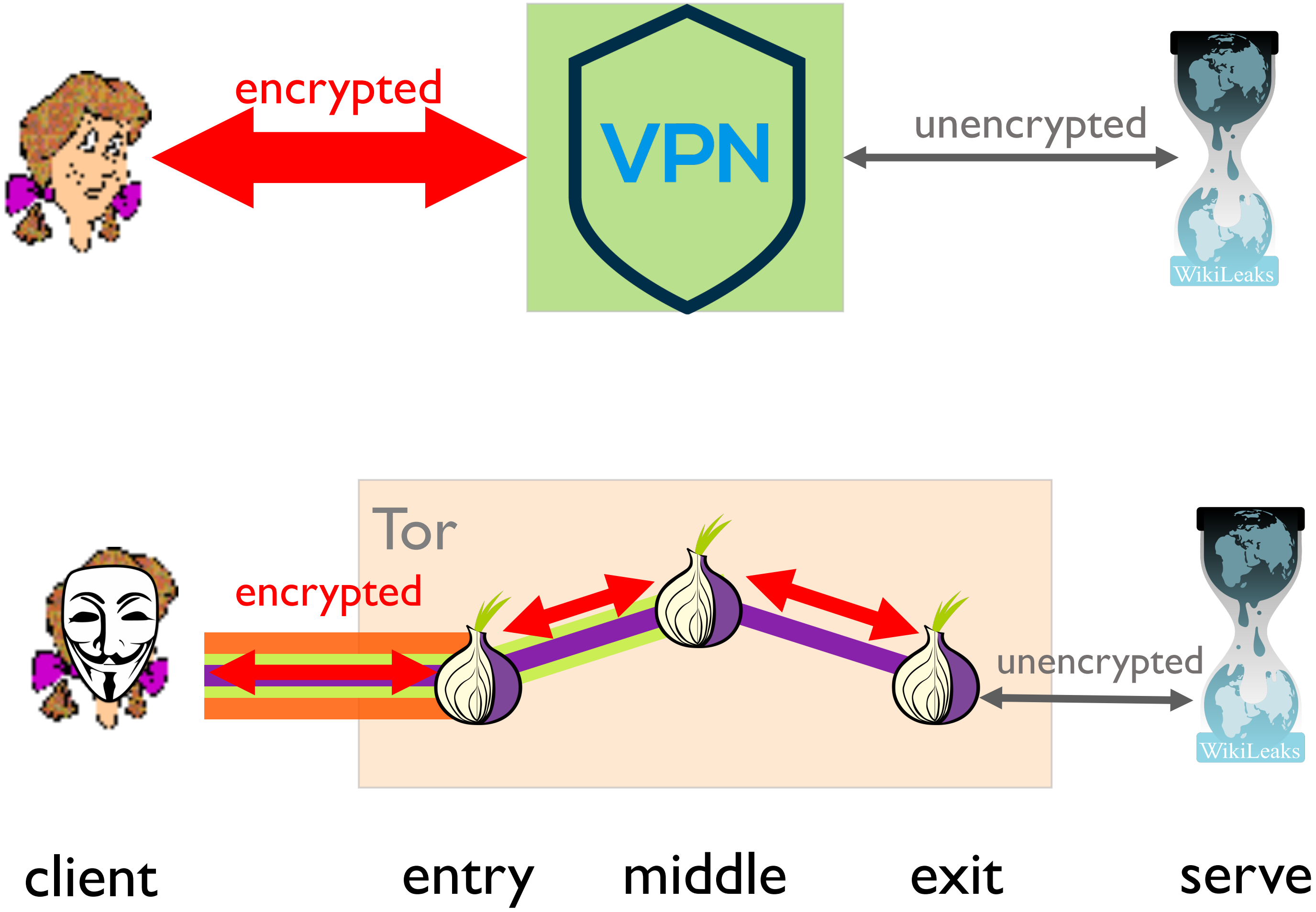


(a) Client: ACK, Server: ACK (b) Client: ACK, Server: Data

Server: Data

Authors were able to deanonymize ~95% of the pairs with no false positives

VPN vs Tor:
Which one is easier to do traffic correlation on?

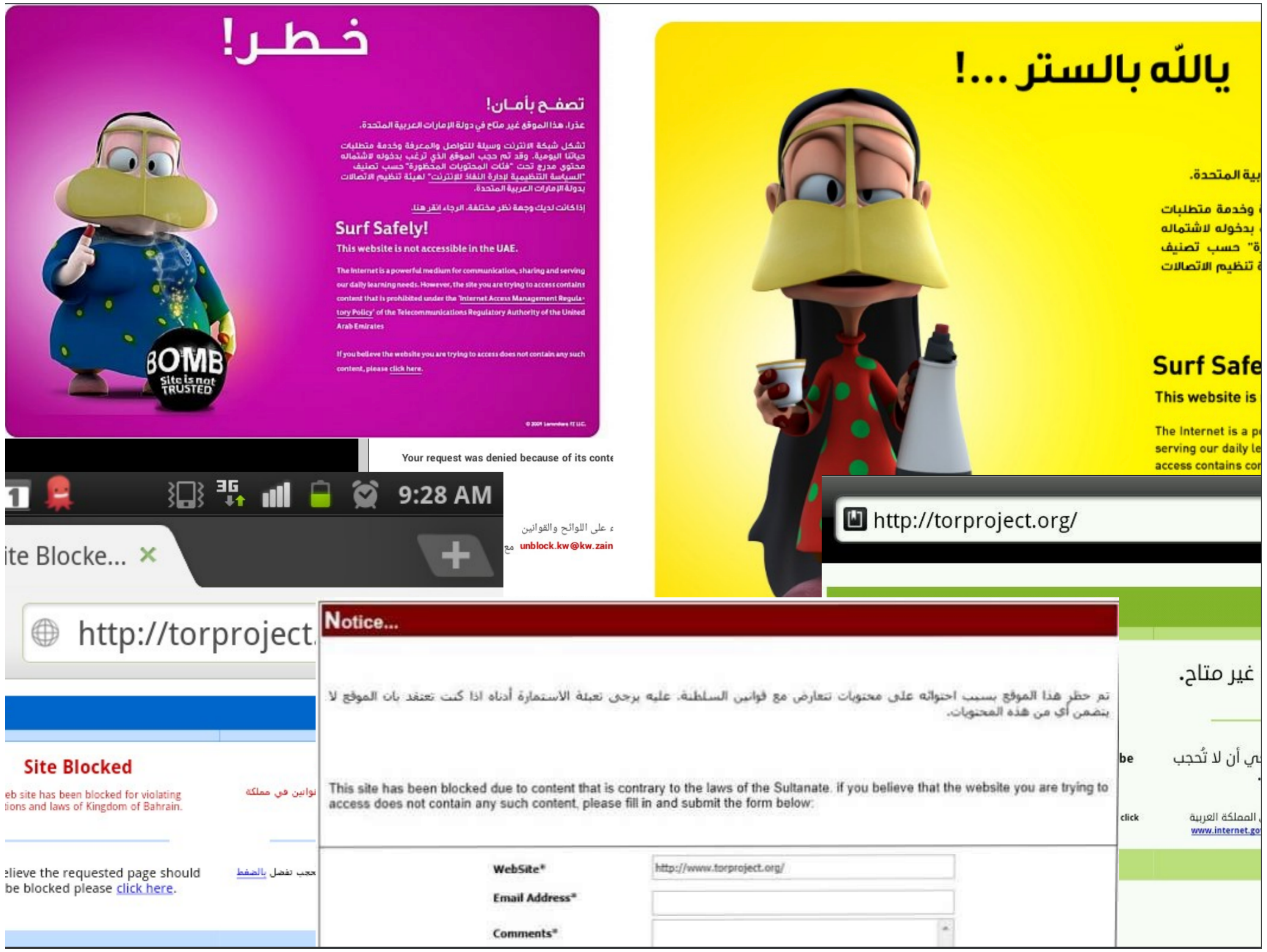


Outline

1. Intro
2. Network Primer
3. Security Primer
4. How Tor Works
5. **Attacks and Censorship on Tor**
   1. Attacks against Tor
   2. Censorship on Tor 🤘

Censorship Arms Race

察察
ChaCha
警警
JingJing



تصفـح بأمـان!

عذرا، هذا الموقع غير متاح في دولة الإمارات العربية المتحدة.

تشكل شبكة الانترنت وسيلة للتواصل والمعرفة وخدمة متطلبات حياتنا اليومية. وقد تم حجب الموقع الذي ترغب بدخوله لاشتماله محتوى مدرج تحت "فئات المحتويات المحظورة" حسب تصنيف "السياسة التنظيمية لإدارة النفاذ للإنترنت" لهيئة تنظيم الاتصالات بدولة الإمارات العربية المتحدة.

إذا كانت لديك وجهة نظر مختلفة، الرجاء انقر هنا.

Surf Safely!

This website is not accessible in the UAE.

The Internet is a powerful medium for communication, sharing and serving our daily learning needs. However, the site you are trying to access contains content that is prohibited under the 'Internet Access Management Regulatory Policy' of the Telecommunications Regulatory Authority of the United Arab Emirates

If you believe the website you are trying to access does not contain any such content, please click here.

Your request was denied because of its conte

ء على اللوائح والقوانين

مع unblock.kw@kw.zain

9:28 AM

ite Blocke... ✕

http://torproject.

Site Blocked

eb site has been blocked for violating tions and laws of Kingdom of Bahrain.

lieve the requested page should be blocked please click here.

http://torproject.org/

Surf Safe

This website is

The Internet is a p serving our daily le access contains co

غير متاح.

Notice...

تم حظر هذا الموقع بسبب احتوائه على محتويات تتعارض مع قوانين السلطنة. عليه يرجى تعبئة الاستمارة أدناه اذا كنت تعتقد بان الموقع لا يتضمن أي من هذه المحتويات.

This site has been blocked due to content that is contrary to the laws of the Sultanate. If you believe that the website you are trying to access does not contain any such content, please fill in and submit the form below:

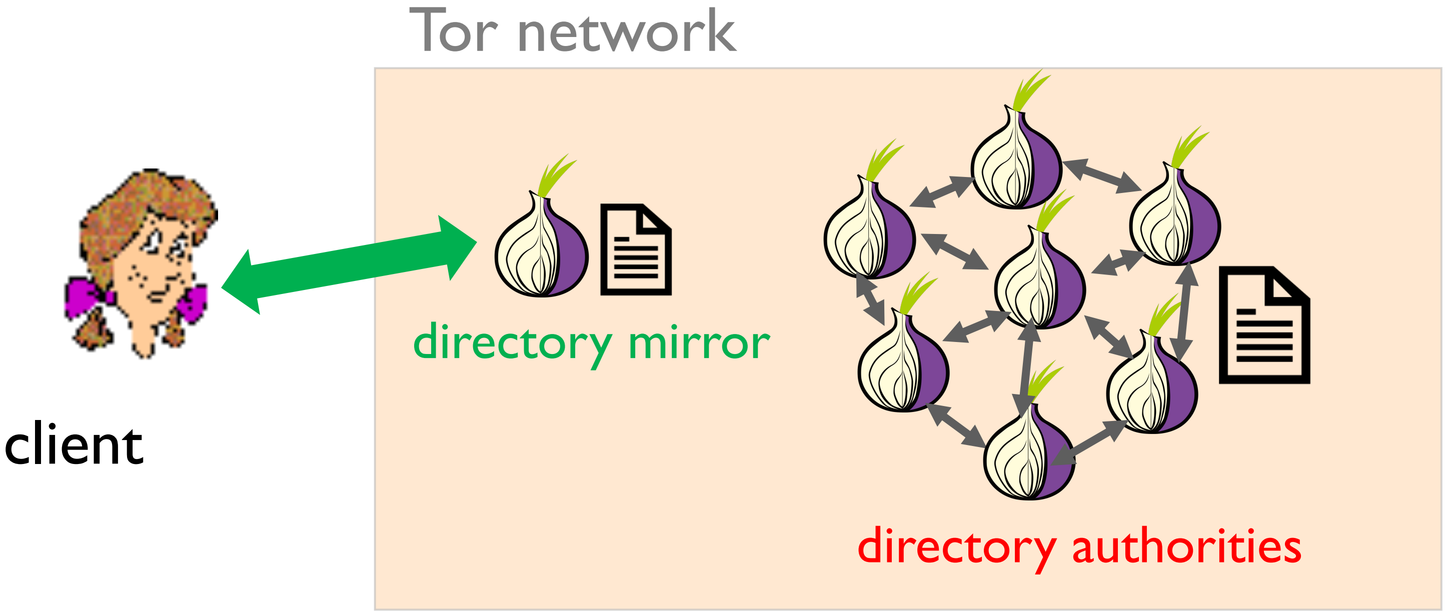
WebSite* http://www.torproject.org/

Email Address*

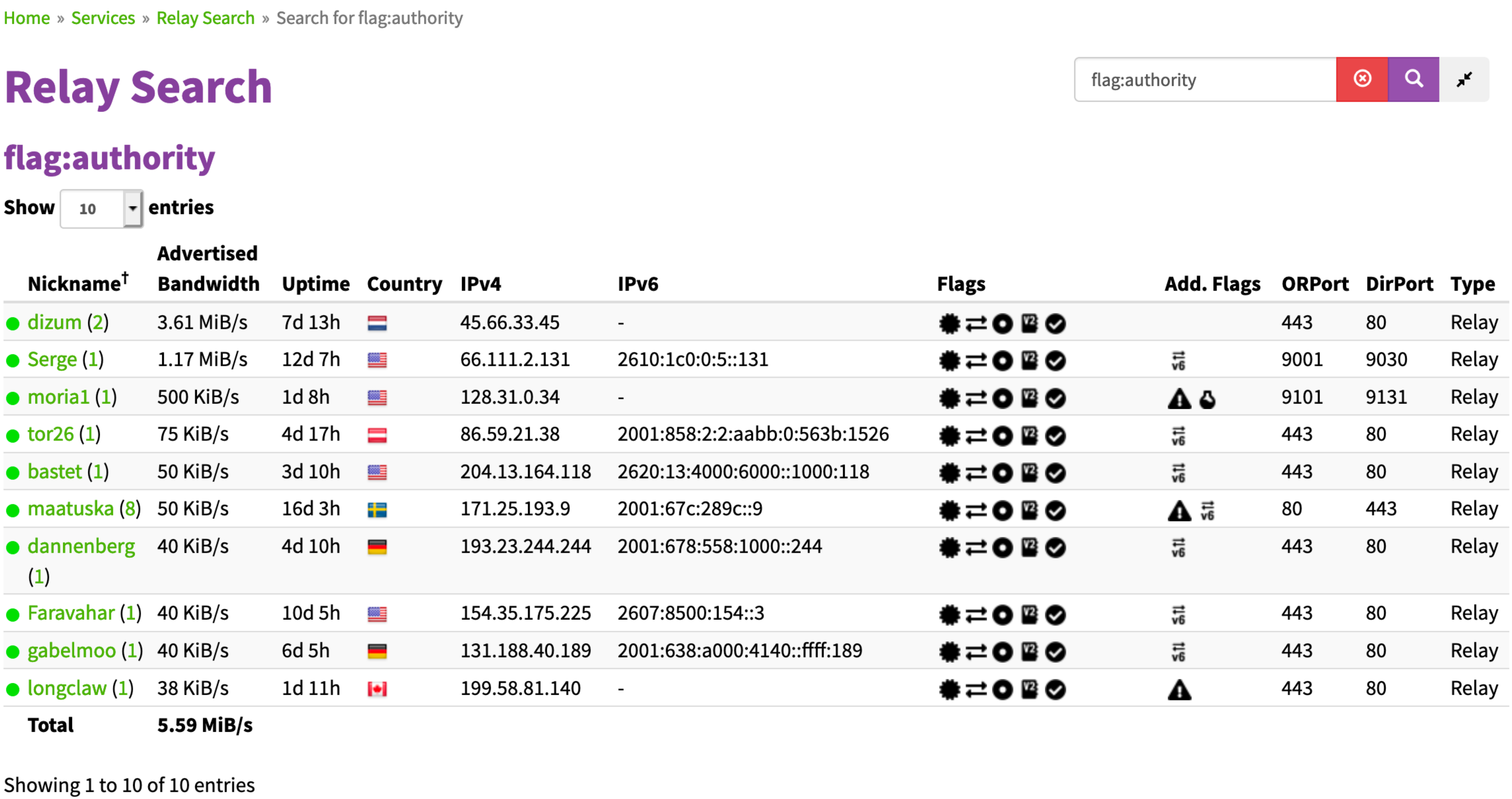
Comments*

How does Alice know
which relays are available to pick for her circuit?

Tor Directory Servers!



1. A set of directory authorities maintain a consensus doc for available relays
2. The consensus info is copied over to many directory mirrors
3. Alice connects to one of directory mirrors and fetches the available relay list



Home » Services » Relay Search » Search for flag:authority

Relay Search

flag:authority

Show 10 entries

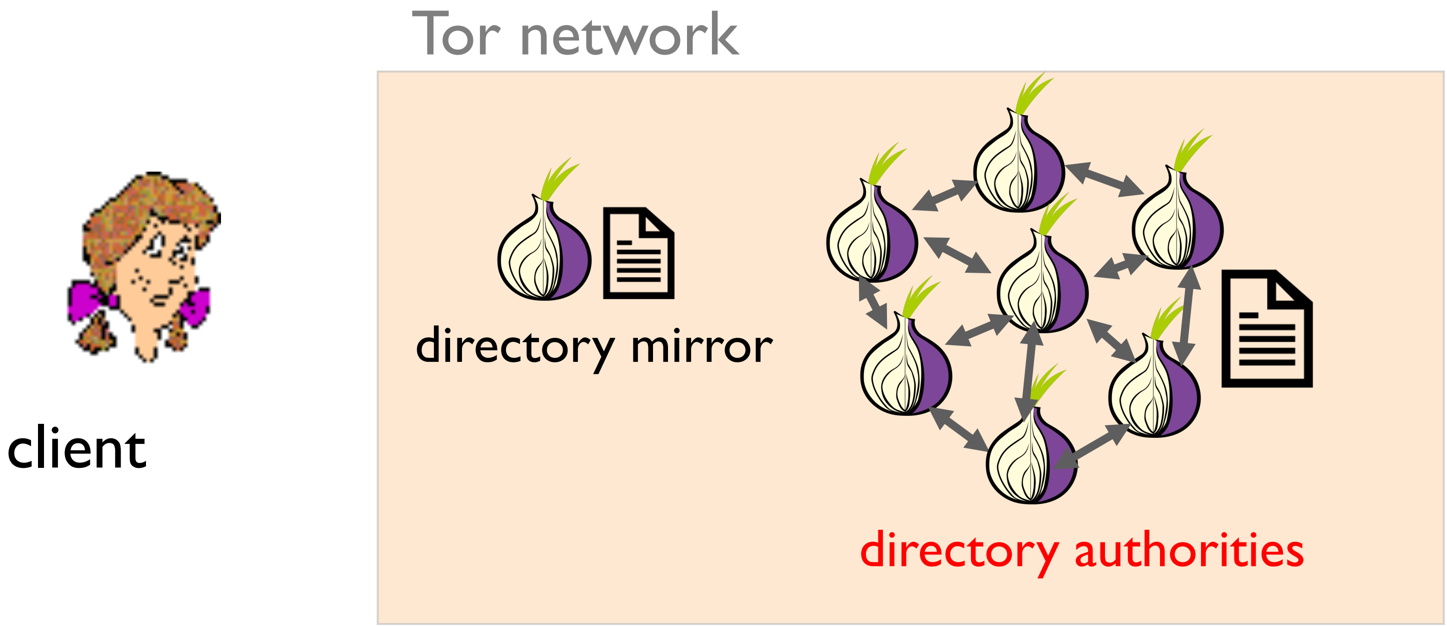
Nickname†	Advertised Bandwidth	Uptime	Country	IPv4	IPv6	Flags	Add. Flags	ORPort	DirPort	Type
● dizum (2)	3.61 MiB/s	7d 13h	🇳🇱	45.66.33.45	-	✹ ⇄ ◉ V2 ✔		443	80	Relay
● Serge (1)	1.17 MiB/s	12d 7h	🇺🇸	66.111.2.131	2610:1c0:0:5::131	✹ ⇄ ◉ V2 ✔	⇄v6	9001	9030	Relay
● moria1 (1)	500 KiB/s	1d 8h	🇺🇸	128.31.0.34	-	✹ ⇄ ◉ V2 ✔	⚠ ⚗	9101	9131	Relay
● tor26 (1)	75 KiB/s	4d 17h	🇦🇹	86.59.21.38	2001:858:2:2:aabb:0:563b:1526	✹ ⇄ ◉ V2 ✔	⇄v6	443	80	Relay
● bastet (1)	50 KiB/s	3d 10h	🇺🇸	204.13.164.118	2620:13:4000:6000::1000:118	✹ ⇄ ◉ V2 ✔	⇄v6	443	80	Relay
● maatuska (8)	50 KiB/s	16d 3h	🇸🇪	171.25.193.9	2001:67c:289c::9	✹ ⇄ ◉ V2 ✔	⚠ ⇄v6	80	443	Relay
● dannenberg (1)	40 KiB/s	4d 10h	🇩🇪	193.23.244.244	2001:678:558:1000::244	✹ ⇄ ◉ V2 ✔	⇄v6	443	80	Relay
● Faravahar (1)	40 KiB/s	10d 5h	🇺🇸	154.35.175.225	2607:8500:154::3	✹ ⇄ ◉ V2 ✔	⇄v6	443	80	Relay
● gabelmoo (1)	40 KiB/s	6d 5h	🇩🇪	131.188.40.189	2001:638:a000:4140::ffff:189	✹ ⇄ ◉ V2 ✔	⇄v6	443	80	Relay
● longclaw (1)	38 KiB/s	1d 11h	🇨🇦	199.58.81.140	-	✹ ⇄ ◉ V2 ✔	⚠	443	80	Relay
Total	**5.59 MiB/s**									

Showing 1 to 10 of 10 entries

Top Countries where Tor relays are located

- The US
- Germany
- France
- Russia
- Netherlands
- United Kingdom

Assume you are to censor Tor
How would you do so?



client

Tor network

directory mirror

directory authorities

How to block users from connecting to Tor

- Blocking connections to all the directory authorities
- Blocking connections to all relays published by the directory authorities
- Filter packets based on Tor's network fingerprint
- Prevent users from finding the Tor browser
  (usually by blocking the website)

Great Firewall of China

- Chinese national level firewall blocks all traffics to Tor
- How to solve this problem?

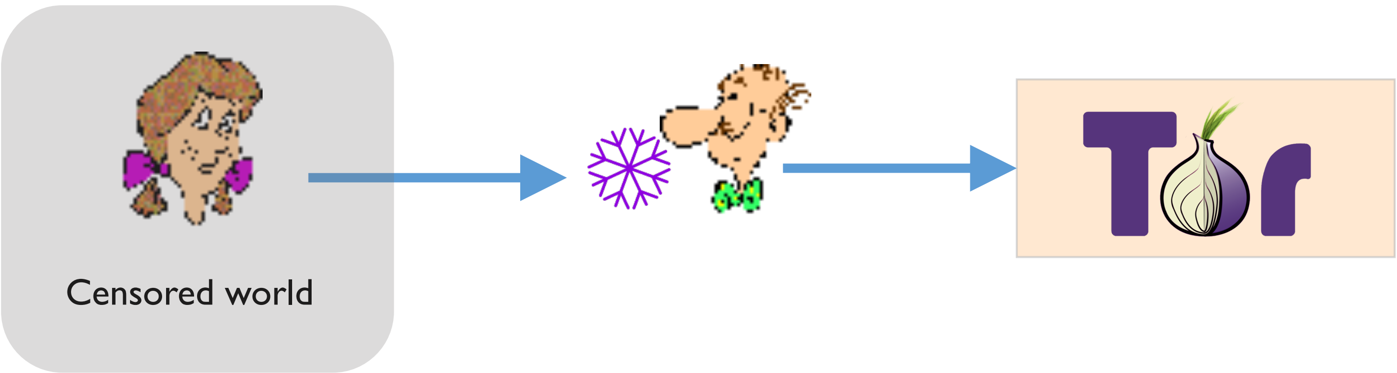
Use bridge nodes!

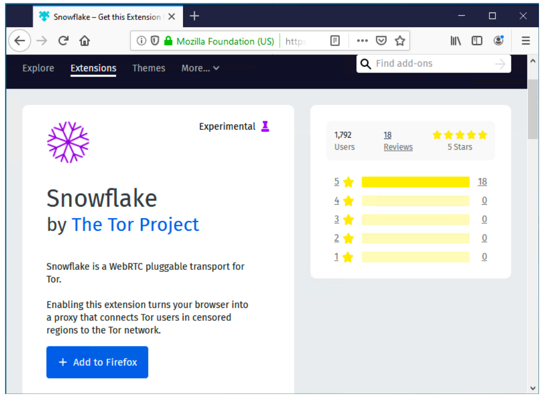
- All Tor nodes are public EXCEPT bridge nodes
- NO complete public list of the bridges
- Makes it difficult to block all the bridges
- How to obtain bridge node info?
  - Tor browser knows some by default
  - Send email to bridges@bridges.torproject.org to get some of them using gmail, Riseup!, or Yahoo account

China’s Active Probing Attacks against Bridges

- Follows a real Tor user connecting to a bridge node (by doing DPI)
- Tries to connect to the suspected node by initiating TLS handshake
- If success, then it has confirmed it’s a Tor node
- Block all connections to that node!

Snowflake enables a user in non-censored world help a user in the censored world connect to Tor





1,792 Users

18 Reviews

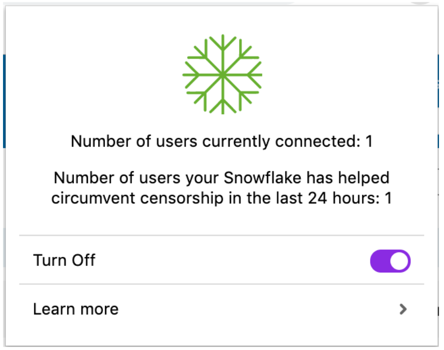
5 Stars

5 ★	18
4 ★	0
3 ★	0
2 ★	0
1 ★	0

Snowflake
by The Tor Project

Snowflake is a WebRTC pluggable transport for Tor.

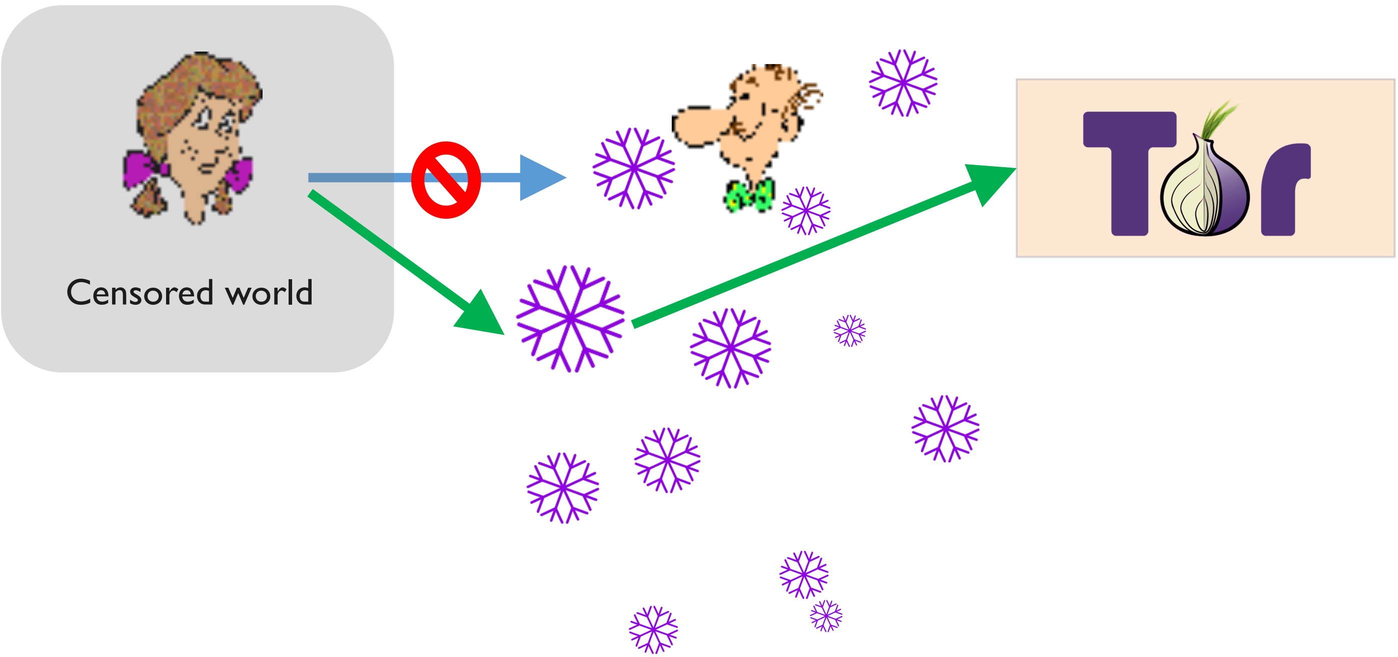
Enabling this extension turns your browser into a proxy that connects Tor users in censored regions to the Tor network.



Number of users currently connected: 1

Number of users your Snowflake has helped circumvent censorship in the last 24 hours: 1

Having blizzards of highly ephemeral snowflakes makes it hard to track and block them all



SOMETIMES

What did we learn today?

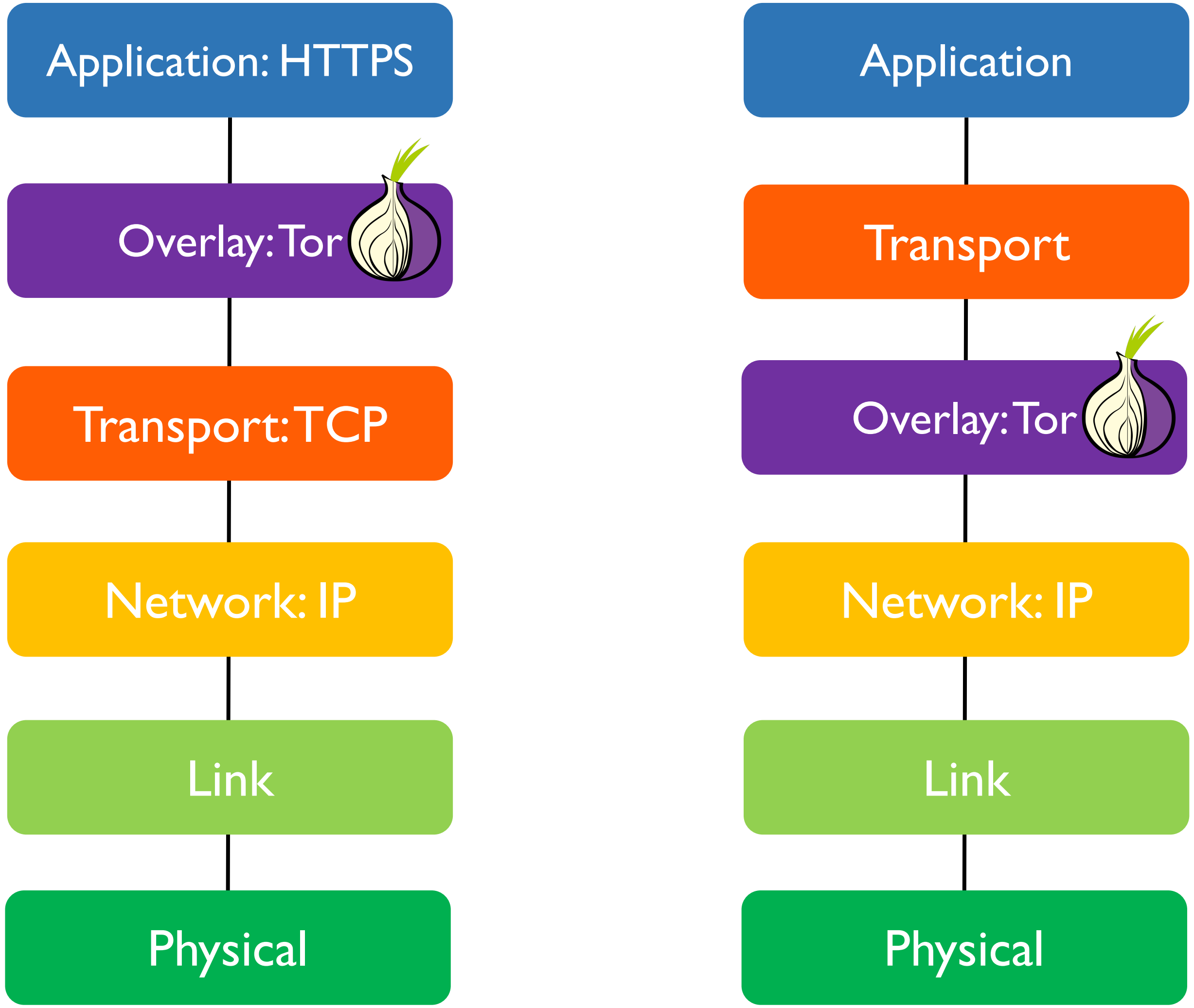
- Tor enables anonymous communication over the Internet
- Tor uses 3 hop encrypted circuit to provide anonymity
- Tor is vulnerable to various attacks and censorship attempts
- Tor is a constantly evolving network protocol to resists them

References

- Tor design paper: DINGLEDINE, R., et. al, Tor: The second-generation onion router. In Proceedings of USENIX Sec'04
- Tor spec: https://gitweb.torproject.org/torspec.git/tree/tor-spec.txt
- Tor Project: https://www.torproject.org/
- RAPTOR paper: Sun, Y., et al, RAPTOR: Routing Attacks on Privacy in Tor. In Proceedings of USENIX Sec'15
- Talks by Tor authors
  - DEFCON27: The Tor Censorship Arms Race The Next Chapter
  - MIT CSS Anonymous Communication Lecture

Backup Slides

Tor is an overlay network designed to provide anonymous communication



Tor’s defense against Censorship

- **obfs4** adds another layer encryption between client and bridge that makes Tor traffic unrecognizable (looks like some random bytes)
- **meek** first connects to a real HTTPS web server (in the Amazon cloud or the Microsoft Azure cloud) and from there connects to the actual bridge