Anonymity

CSIE 7190 Cryptography and Network Security, Spring 2019

https://ceiba.ntu.edu.tw/1072csie_cns

cns@csie.ntu.edu.tw

Hsu-Chun Hsiao

Housekeeping

Final project topic & members due today!

4/16: 1st midterm exam

4/23 2pm: Reading critique #7 deadline

Reading critique #7

Write a critique on one of the following:

- K. Butler, T. R. Farley, P. McDaniel, and J. Rexford, "A survey of BGP security issues and solutions," in Proceedings of the IEEE, 2010.
- O. Nordström and C. Dovrolis, "Beware of BGP attacks," ACM SIGCOMM Comput. Commun. Rev., vol. 34, no. 2, p. 1, 2004.
- An Illustrated Guide to the Kaminsky DNS Vulnerability http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html

Text only, one page

Midterm

Time: 14:20-17:20 (3 hours)

Format: 簡答 (similar to your handwriting homework)

Scope:以上課投影片和上課內容為主

- Crypto primitives (hash, MAC, encryption, signatures…)
- Authentication
- Anonymity and privacy

Agenda

Anonymity

Censorship circumvention

Privacy

Midterm Review

"On the Internet, nobody knows you're a dog."

課堂小調查

uTest APPLAUSE

MOBILE RESEARCH PROJECT

Hi [redacted]

Ganesh Singla has invited you to sign-up for an exclusive research project being conducted by Applause Inc. Participants in this study have the opportunity to **earn as much as they want each month with no work required!** Sound too good to be true? Read on!

How It Works

It takes **5 minutes to join** the program:

1. Register for the project here using this email address [redacted]@gmail.com)
2. Follow the instructions (after "Submit") to install the mobile application on your phone
3. Keep the application installed while you use your phone as usual.

Install it and forget it - no additional work necessary!

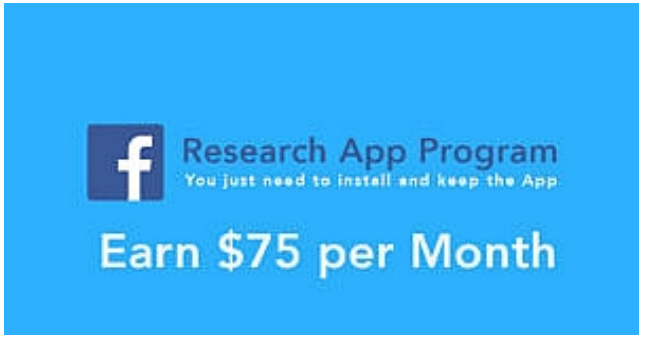
How You Make Money

Once installed, you will be paid for each month the application remains installed and active on your phone.

In addition to being paid for your participation, you will also be paid every month for every participating referral you provide. If you refer even just **5 friends** to the project within 30 days of joining the program, you could earn up to **$75 per month** for just keeping the application installed!

The best part? There is no limit to how many referrals you can provide! Refer 5, 10, 100 or more friends and see your monthly payout increase with every referral that joins!

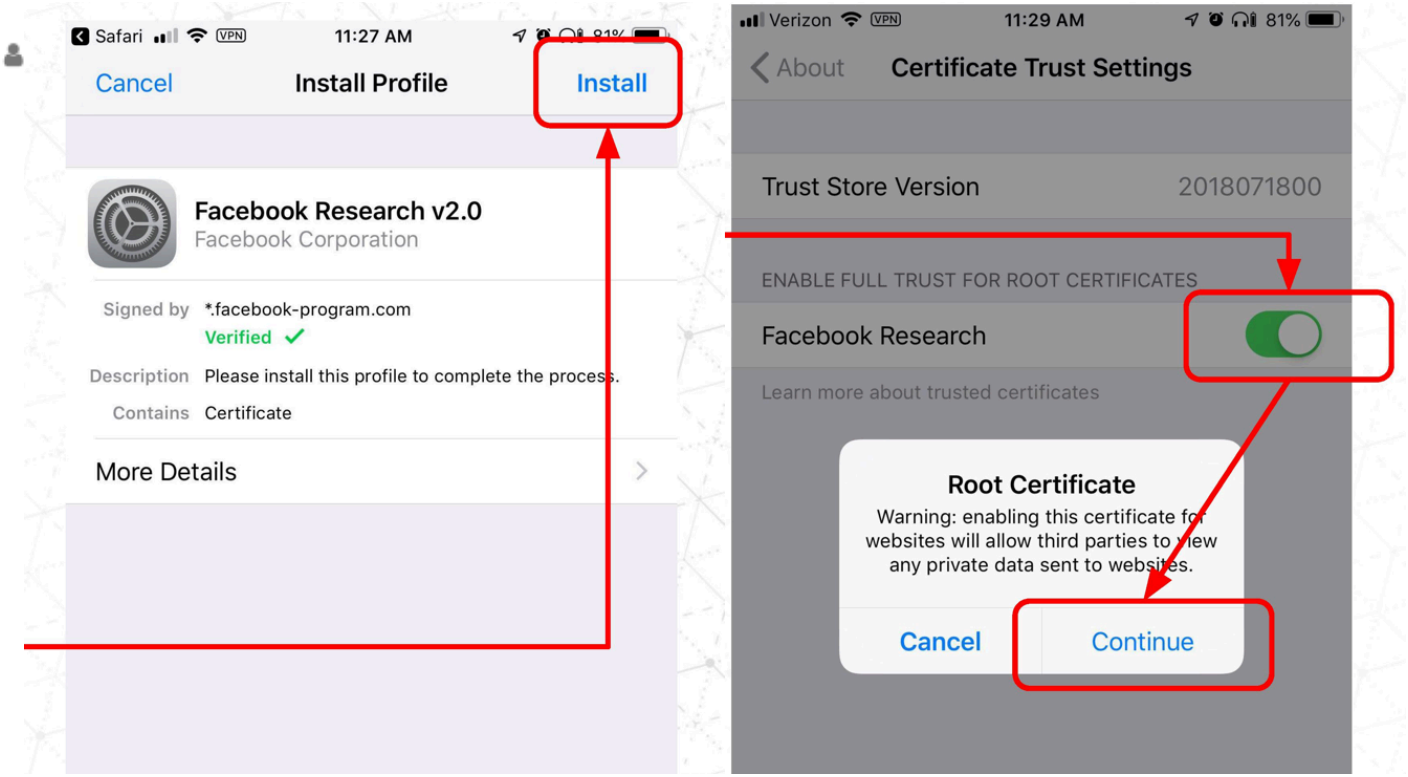
Click here to learn more and to sign-up for the project!



How much is your privacy worth?

Facebook Paid Teens $20 to Install 'Research' App That Collects Private Data

January 30, 2019



https://techcrunch.com/2019/01/29/facebook-project-atlas/
https://thehackernews.com/2019/01/facebook-research-app.html

How much is your privacy worth?

AT&T charges $29 more for gigabit fiber that doesn't watch your Web browsing

AT&T goes head to head against Google in KC on fiber and targeted ads.

違反GDPR重罰首例，Google遭法國重罰5千萬歐元

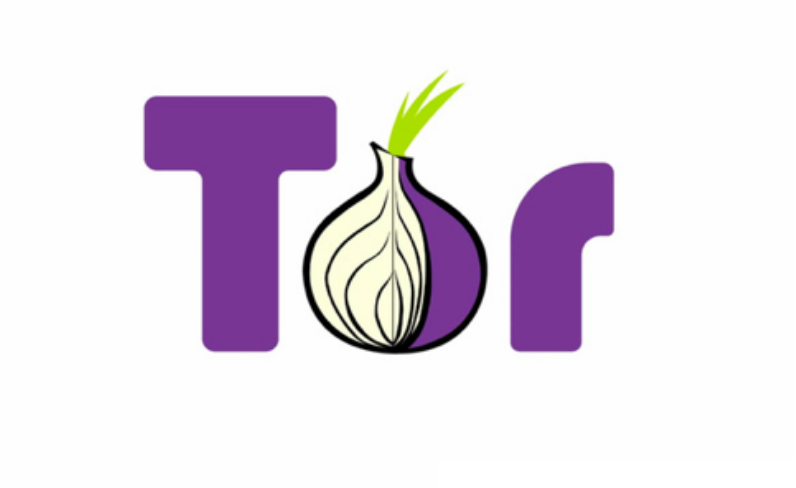
法國資料主管機關以Google對用戶資訊透明度不足、剝奪使用者控管個資能力等原因而違反GDPR規範，對其判處5千萬歐元罰款

文/ 林妍溱 | 2019-01-22 發表

讚 5.3 萬 按讚加入iThome粉絲團 讚 1,381 分享

GDPR = General Data Protection Regulation

Have you used these?



Security與Privacy之間的權衡

FBI Debates Whether To Share iPhone Hack With Apple

April 8, 2016 9:05 PM

Filed Under: encryption, FBI, iPhone

(Photo by Chip Somodevilla/Getty Images)
(Photo by Chip Somodevilla/Getty Images)

WASHINGTON (AP) – The FBI has not d
Apple Inc. details about how the bureau l
California terrorism investigation, the bur

James Comey discussed the situation du
evening at Kenyon College in Ohio. He c
iPhone a "technological corner case" and
in Apple's software works only on a "narr
5C, running version 9 of Apple's mobile c
or older models.

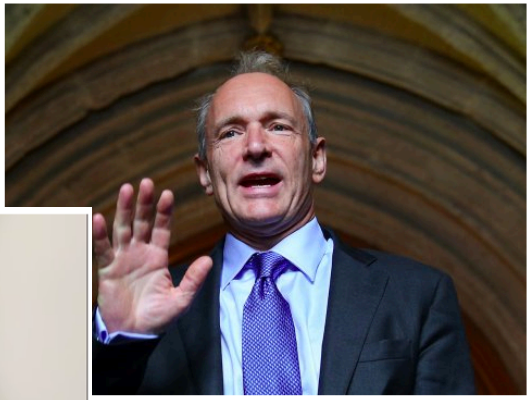
The inventor of the web warned that calls to weaken encryption are a 'bad idea'

Rob Price

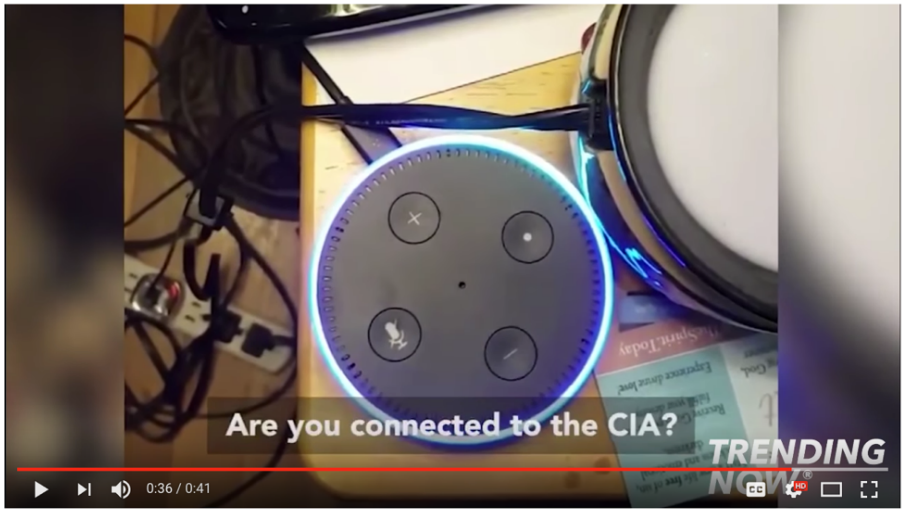
Apr. 4, 2017, 12:36 PM 489

FACEBOOK LINKEDIN TWITTER EMAIL PRINT

LONDON — Tim Berners-Lee, the inventor of the web, has waded into Britain's debate over encryption, calling any attempts to weaken the technology a "bad idea."



ners-Lee inventor of the World Wide Web.



Amazon Alexa Shuts Down When Asked "Alexa Are You Connected To The CIA" /TrendingNow

29,566 views 146 8

https://www.youtube.com/watch?v=B5MvzLFC8C8

How China Turned a City Into a Prison

https://www.nytimes.com/interactive/2019/04/04/world/asia/xinjiang-china-surveillance-prison.html

Terminology

One perspective:

- (Online) **privacy** is a right to control over *how*, *where* and *with whom* information about yourself is disclosed
- **Anonymity** is being non-identifiable, a kind of privacy regarding identities
- **Security** is mechanisms for realizing privacy

Whatever your definition of privacy is, privacy violations can cause real problems for relationships, job searching, insurances, personal safety, etc.

Anonymity

Mix networks

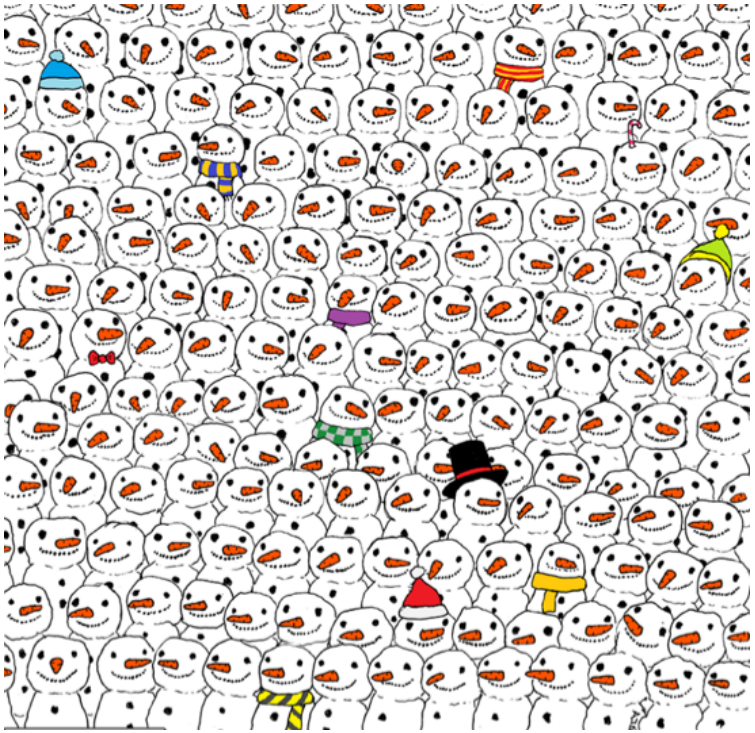
DC-nets

Tor

Quantifying anonymity

Anonymity = being non-identifiable **within a set of subjects**

- You cannot be anonymous by yourself!
- The set of subjects = **anonymity set**
- An observer can see an action (e.g., an email sent) but anyone in the anonymity set may be the actual sender

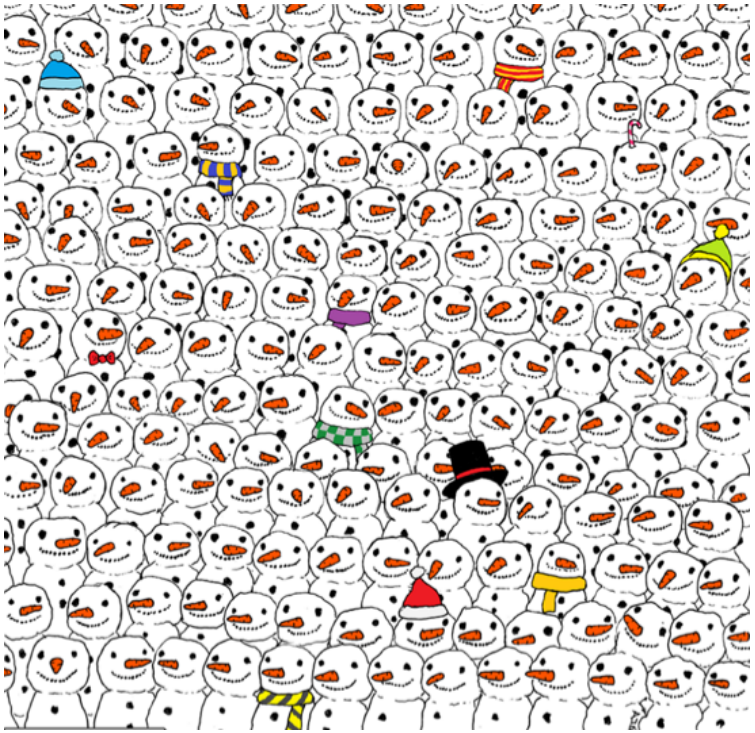


Quantifying anonymity

Anonymity = being non-identifiable **within a set of subjects**

The size of anonymity set can be one metric of anonymity

Larger anonymity set = stronger anonymity



Who needs anonymity?

Criminals?

Whistle-blowers (e.g., WikiLeaks)

Journalists

Political dissidents

Government agencies

Ordinary people, too!

- Avoid tracking by advertising companies
- In countries without freedom of speech
- Anti-censorship
- Anonymous voting
- Anonymous digital cash

Staying anonymous online is hard

Too much identifiable info, too many bad guys.

Every packet you send/receive has your IP address

Servers track your browser via cookies or unique settings

- Browser fingerprint: {User Agent, Browser Plugin Details, Time Zone, Screen Size and Color Depth, …}

Every layer needs to be protected

https://panopticlick.eff.org

Your browser fingerprint **appears to be unique** among the 351,694 tested in the past 45 days. Currently, we estimate that your browser has a fingerprint that conveys **at least 18.42 bits of identifying information.**

Browser Characteristic	bits of identifying information	one in x browsers have this value
User Agent	12.23	4817.73
HTTP_ACCEPT Headers	3.16	8.96
Browser Plugin Details	2.9	7.46
Time Zone	5.4	42.09
Screen Size and Color Depth	4.54	23.23
System Fonts	3.73	13.27
Are Cookies Enabled?	0.22	1.16
Limited supercookie test	0.32	1.24
Hash of canvas fingerprint	8.09	273.27
Hash of WebGL fingerprint	5.51	45.55
DNT Header Enabled?	1.25	2.38
Language	0.88	1.84
Platform	3.1	8.6
Touch Support	0.64	1.56

Network-layer anonymity

Achieving network-layer anonymity is nontrivial

- Every on-path entity (router, ISP, server…) see your packets
- Every packet you send/receive has your IP address

Encryption only hides payload, not routing information

- Encrypting payload provides confidentiality of the data but doesn't provide sender/receiver anonymity

Routers need to see source IP and destination IP to correctly deliver packets

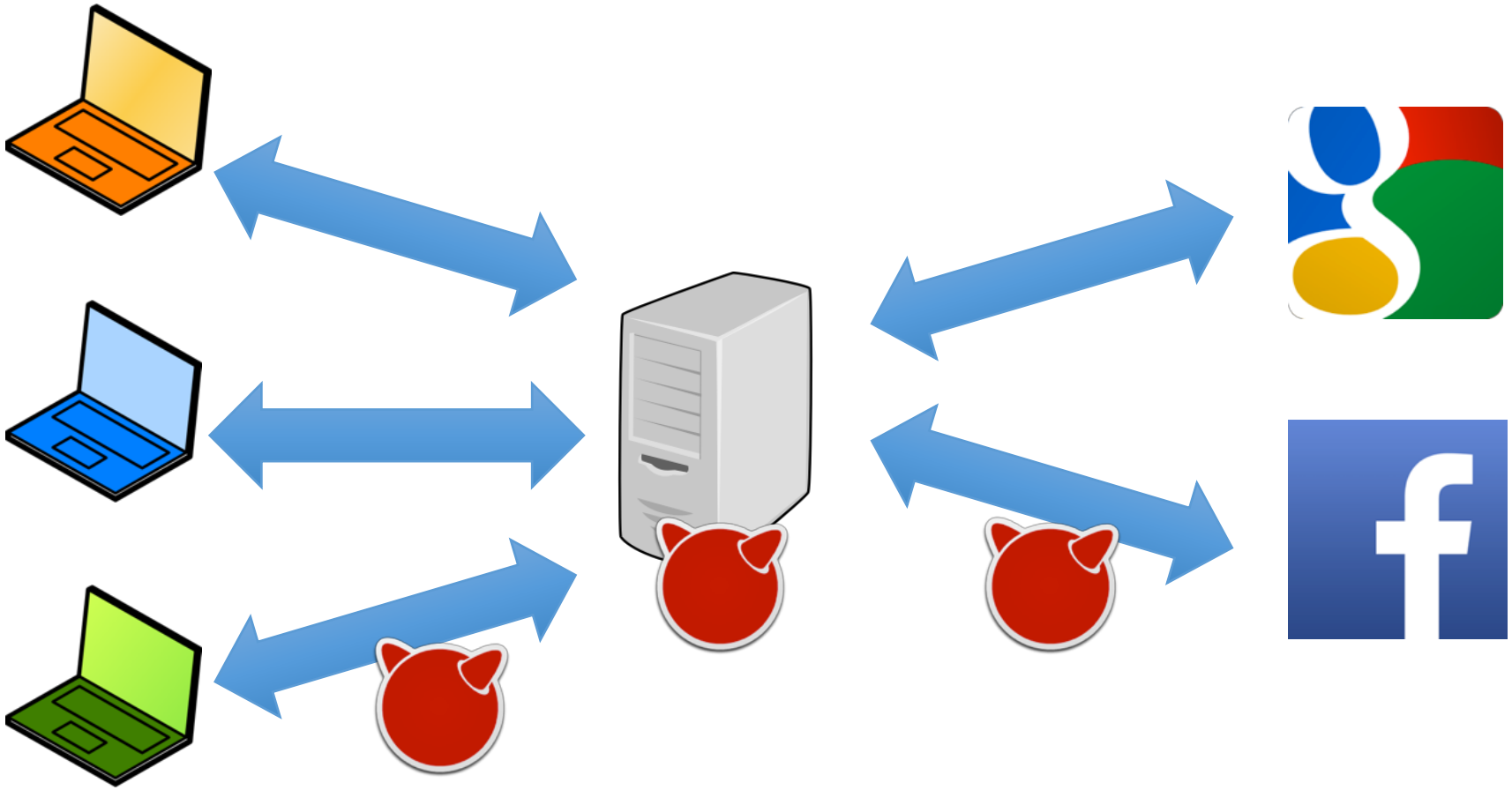
Network-layer anonymity

Desirable security properties:

- **Sender anonymity**: cannot identify the sender (IP) of a packet
- **Receiver anonymity**: cannot identify the receiver of a packet
- **Unlinkability**: cannot tell if A and B is talking to each other
- **Unobservability**: cannot tell if an entity is communicating at all (hard to achieve)

Note that the most common identifier at the network layer is IP address

How about using proxies or VPNs?



How about using proxies or VPNs?

Possible attacks and consequences

Attacker eavesdrops on traffic between sender and proxy:

- Sender is known, receiver anonymity

Attacker eavesdrops on traffic between proxy and receiver:

- Receiver is known, sender anonymity

Attacker compromises proxy or observes both sides of the traffic (traffic analysis):

- No anonymity at all

Attacker blocks traffic to/from proxy:

- Cannot even use it

How about using proxies or VPNs?

Advantages: simple, easy to use, (relatively) fast

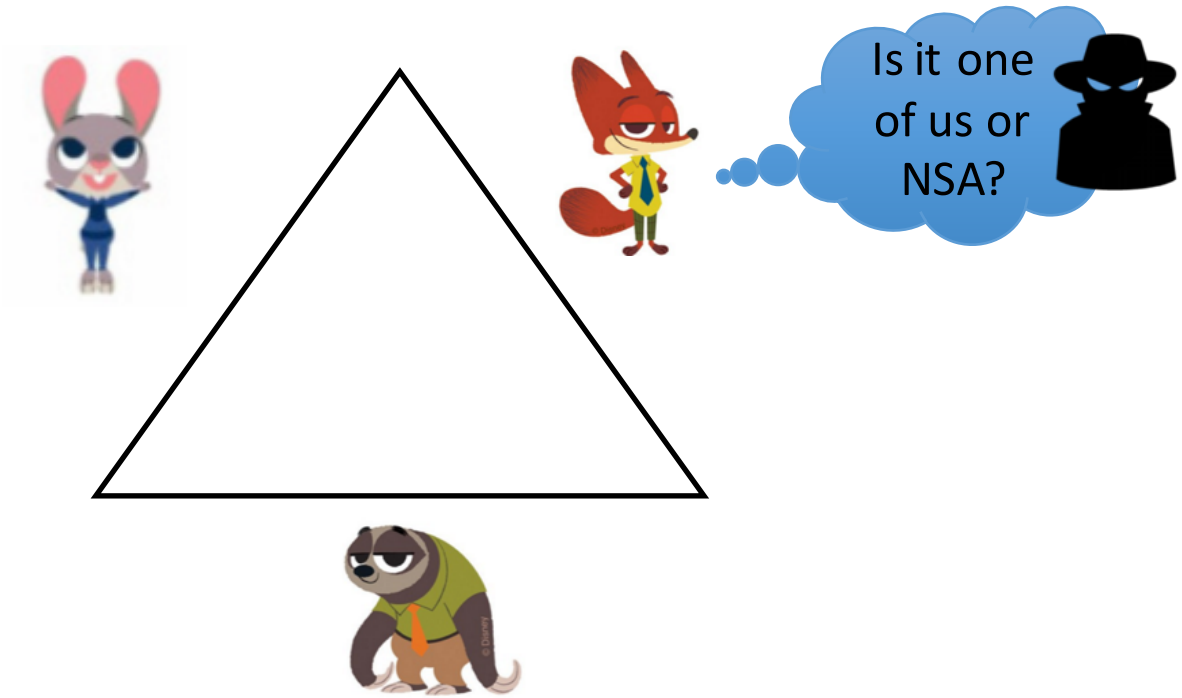
Disadvantages:

- Proxy is assumed trusted
- Single point of failure
- Anonymity set may be small

The Dining Cryptographers Problem

The Dining Cryptographers Problem

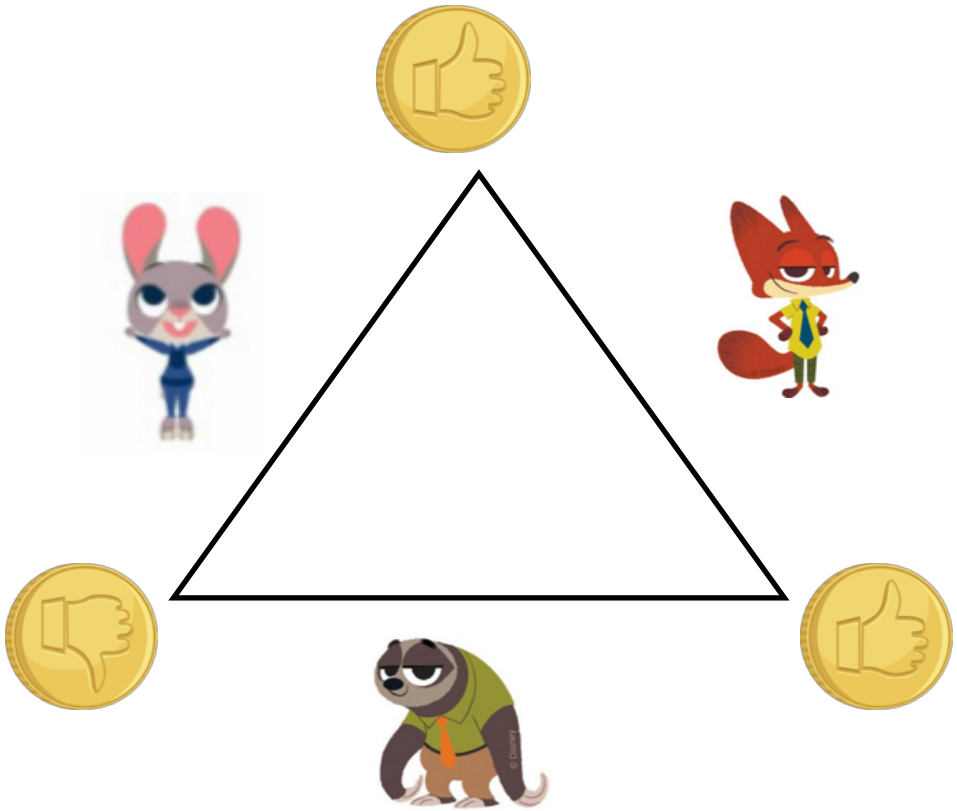
Three cryptographers are having dinner and are told the dinner is paid. One of the cryptographers might be paying for the dinner, or it might have been NSA. The three cryptographers respect each other's right to make an anonymous payment, but they wonder if NSA is paying…



David Chaum. "The dining cryptographers problem: unconditional sender and recipient untraceability." Journal of Cryptology, 1988.

The Dining Cryptographers Problem

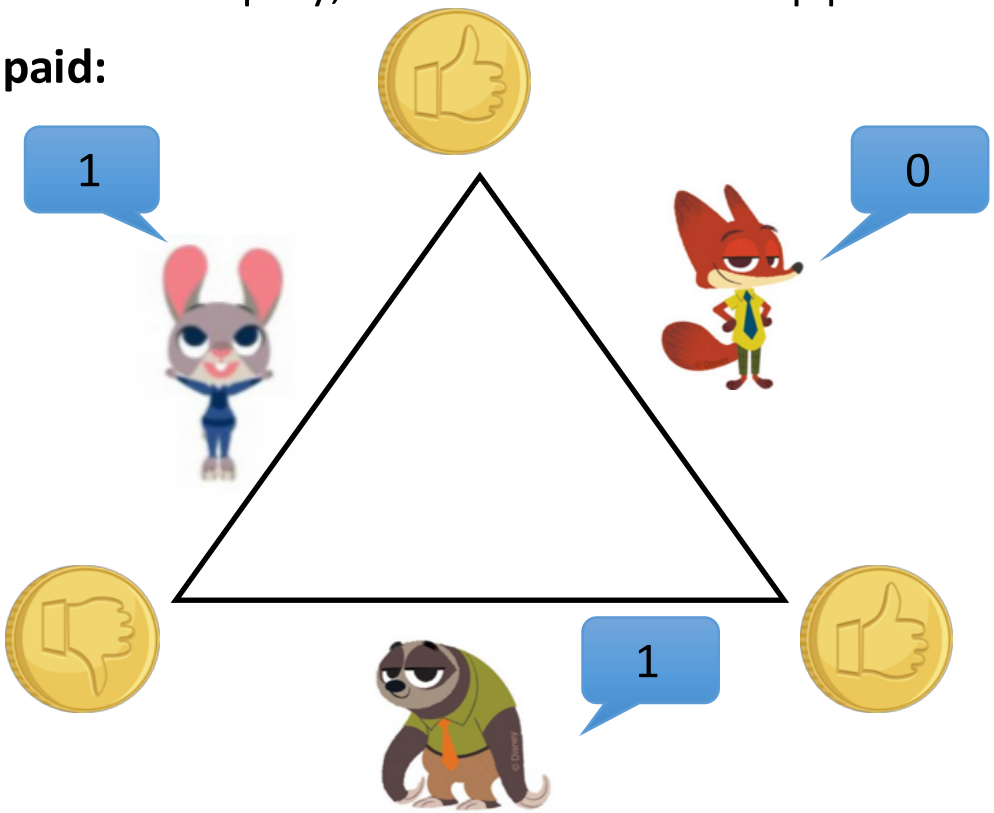
1. Each flips a coin (1 bit) and shows it to the left neighbor
   - Everyone sees two coins



The Dining Cryptographers Problem

2. Each person publicly announces a bit:
   - If the person didn't pay, announce XOR of the two coins.
   - If the person did pay, announce the opposite of the XOR.

Example when NSA paid:

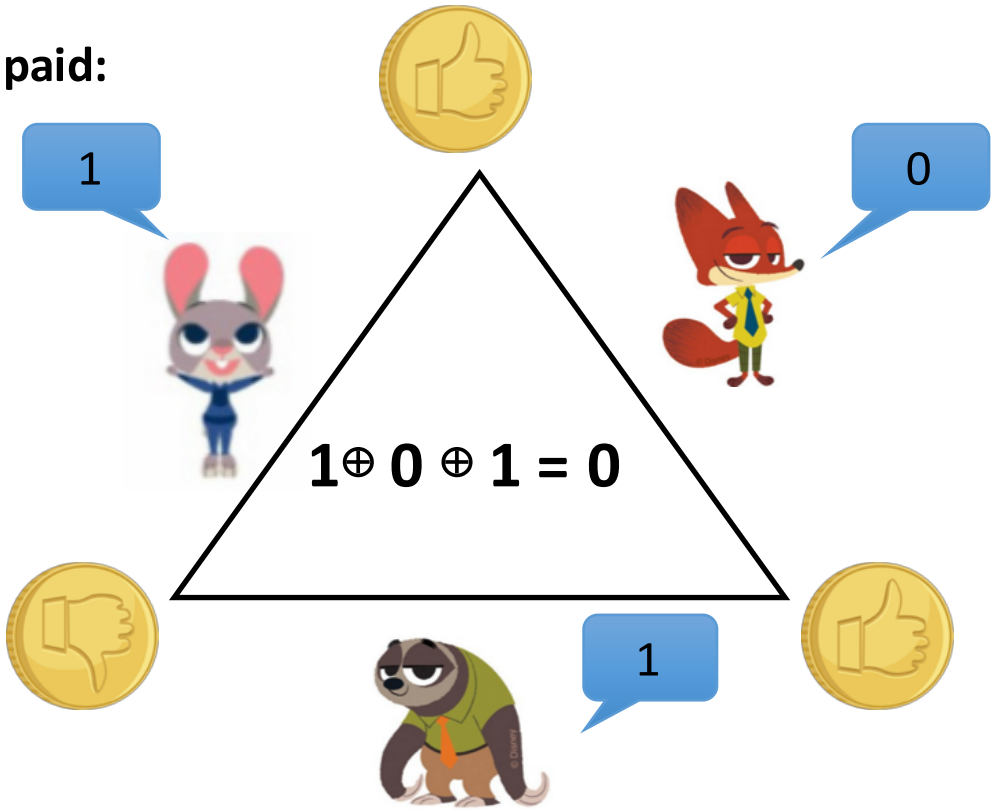


The Dining Cryptographers Problem

3. XOR all the announced bits
   - 0 -> NSA paid
   - 1 -> one of the cryptographers paid

Non-payer cannot tell who paid!

Example when NSA paid:

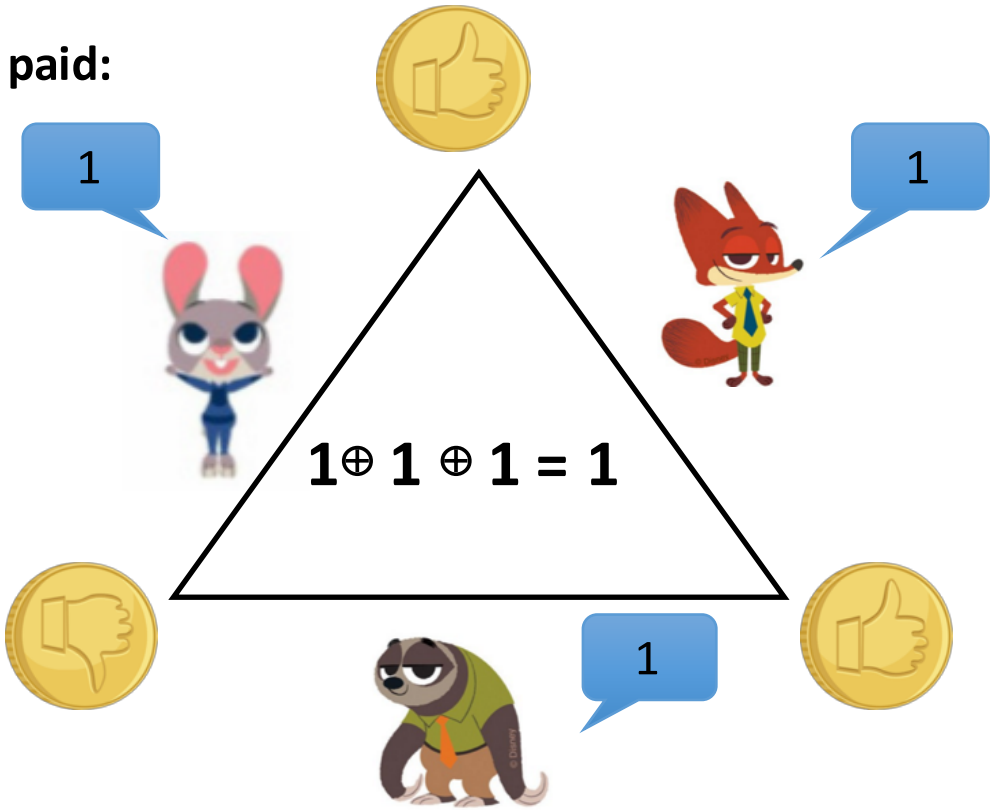


The Dining Cryptographers Problem

3. XOR all the announced bits
   - 0 -> NSA paid
   - 1 -> one of the cryptographers paid

Non-payer cannot tell who paid!

Example when Nick paid:



DC-nets

DC-nets are anonymous communication networks based on the dining cryptographers problem.

- Each run the sender can transmit 1-bit to the recipient.
- **Sender anonymity**: one-time-pad-like encryption
- **Receiver anonymity**: broadcast transmission

DC-nets achieve information-theoretic anonymity

- Stronger than computational security; the attacker cannot break it even with unbounded computational power

DC-nets

This can be generalized to a group of size *N*.

However, DC-nets are impractical. Why?

Not scalable

Mix Networks

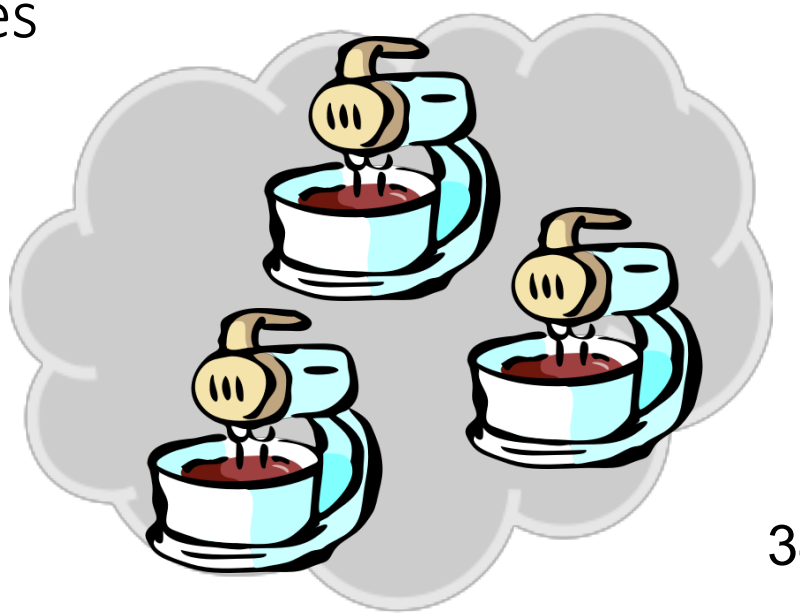
Mix networks

Proposed for anonymous email by David Chaum in 1981

- D. Chaum. "Untraceable electronic mail, return addresses, and digital pseudonyms". Communications of the ACM, February 1981.

A "mix" is designed to provide anonymity against a global passive adversary (who observes all traffic)

A mix network, or a cascade of mixes, is resilient to the compromise of a subset of mixes



A mix: traffic mixing

Decrypt/encrypt packets (details in the next slide)
Buffer packets for several seconds
Add dummy packets and paddings
Shuffle packets and send in random orders

Advantages: reduce correlation by timing, packet size, packet content, …

Disadvantages: increased latency

A mix

Suppose Nick wants to email Google via a mix

Step 1: Nick encrypts the message *M* twice, first by Google's public key, and then by the mix's public key

- R_is are random strings to ensure non-deterministic encryption (preventing traffic analysis by input/output content)

Step 2: the mix decrypts using its private key and forwards to Google

Step 3: Google decrypts to retrieve *M*

pk_{mix}, pk_g are public

$E_{pk_{mix}}(R_1||E_{pk_g}(R_0||M)||\text{“}Google\text{”})$

mix (knows sk_{mix})

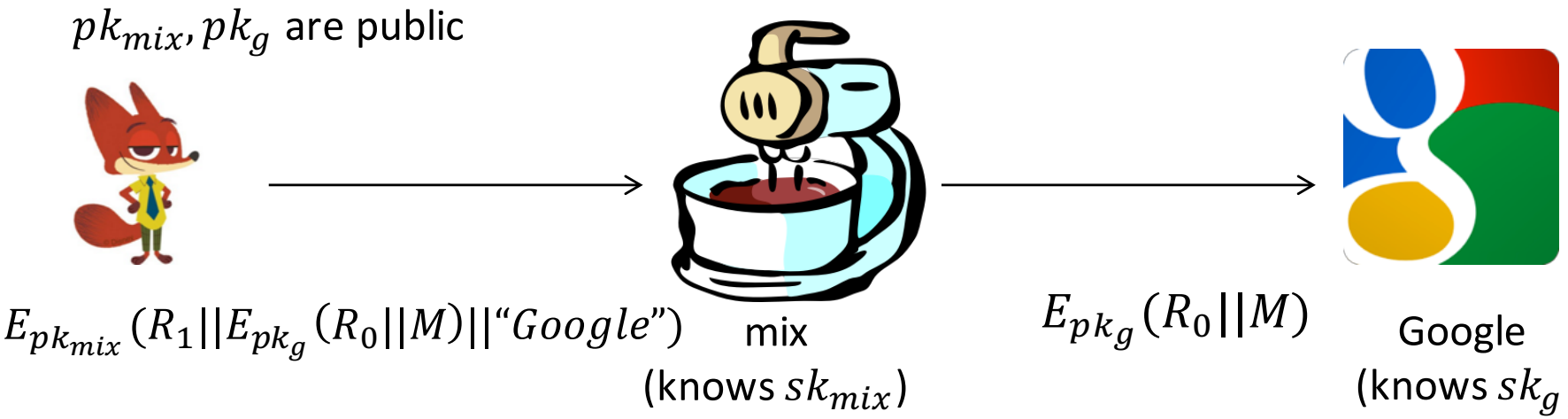
$E_{pk_g}(R_0||M)$

Google (knows sk_g)

A mix

A global passive adversary cannot link sender & receiver

What if the mix is compromised?

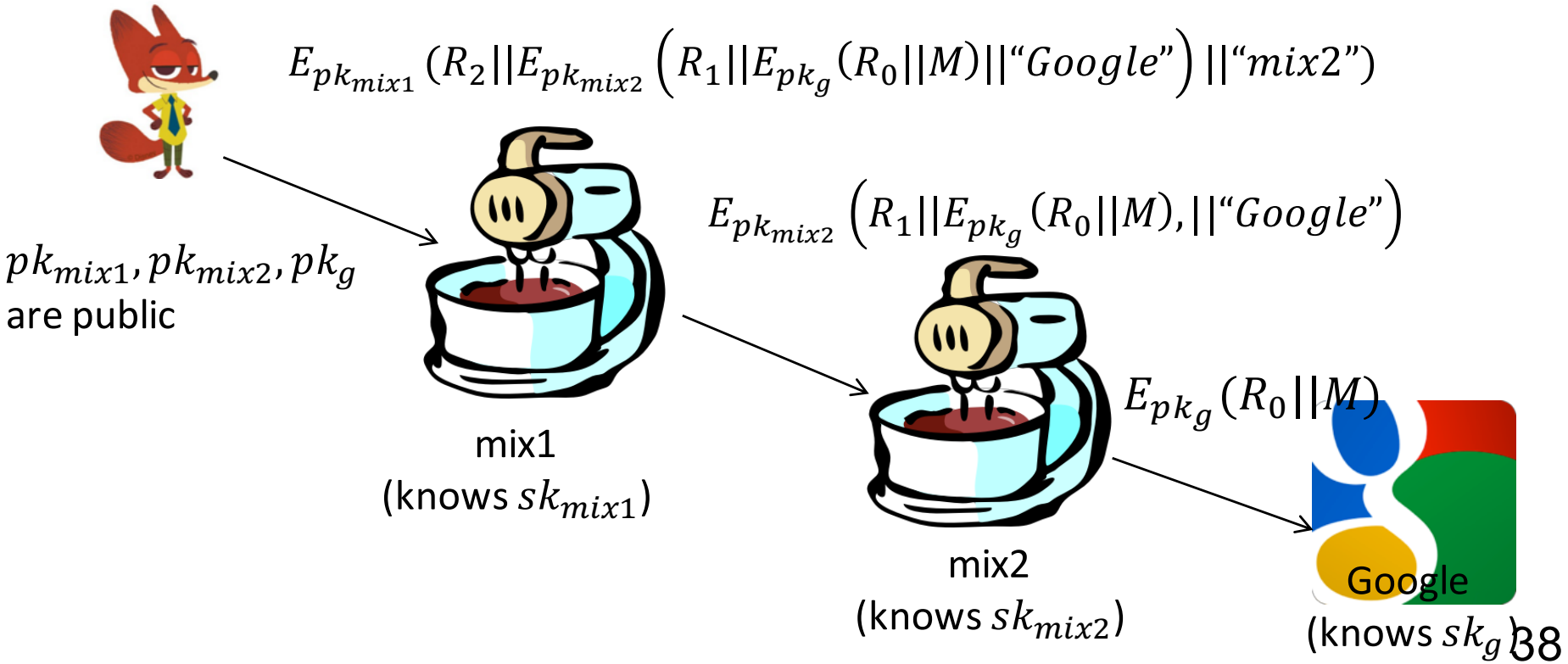


Mix network

Nick chooses a sequence of mixes

Layered encryption using public-key cryptography

Each mix only knows the previous hop and the next hop, resilient to compromise of a subset of mixes

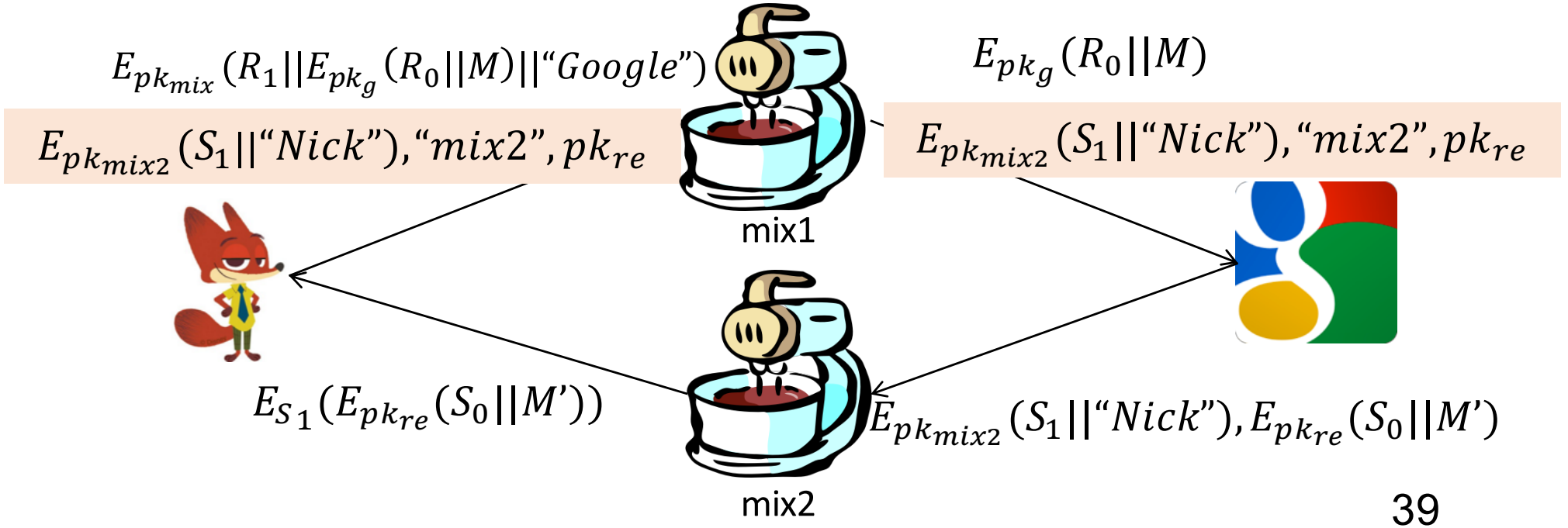


Mix network: Anonymous return address

The receiver doesn't know who the sender is. How can the receiver respond to the sender?

Using an untraceable return address (附上匿名的回郵信封)

- Nick picks S_1 as a symmetric key, and pk_{re} as a one-time public key



Mix network: Security analysis

Are these achieved (w.r.t. what threat model)?

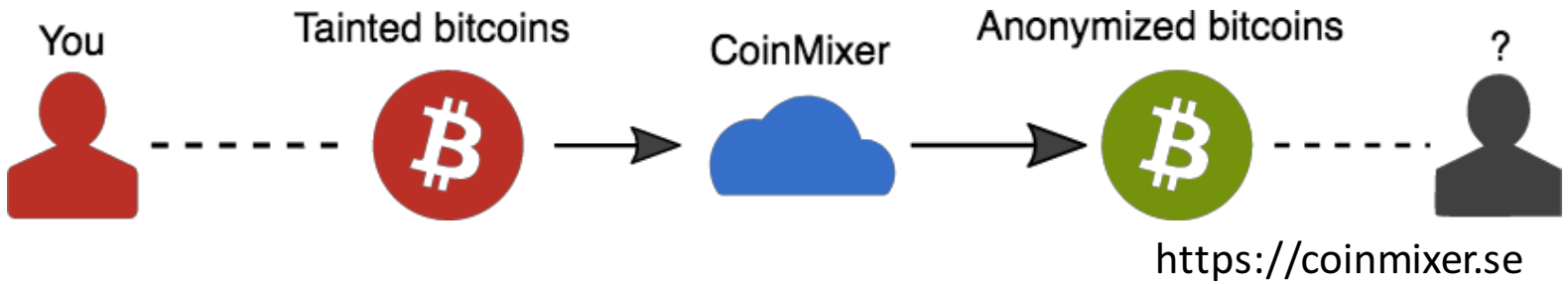
- **Sender anonymity**: cannot identify the sender (IP) of a packet
- **Receiver anonymity**: cannot identify the receiver of a packet
- **Unlinkability**: cannot tell whether A and B is talking to each other
- **Unobservability**: cannot tell if an entity is communicating at all

Discussion

Mix networks can defend against a strong attacker that observes all traffic and compromises some mixes

Mixnet implementation: Mixmaster, Mixminion (no longer under active development)

Other application: cryptocurrency mixing service



Discussion

However, operations in mix networks are costly

- Buffering and reordering causes high latency, which may be fine for email but bad for Internet browsing
- Public-key encryption/decryption are computationally expensive and thus introduce additional latency

High latency is intolerable for most of the online activities nowadays; Tor is designed to be a low-latency anonymity network.

Tor: The Onion Router

Dingledine, Roger, Nick Mathewson, and Paul F. Syverson. "Tor: The Second-Generation Onion Router." USENIX Security Symposium. 2004.

Tor

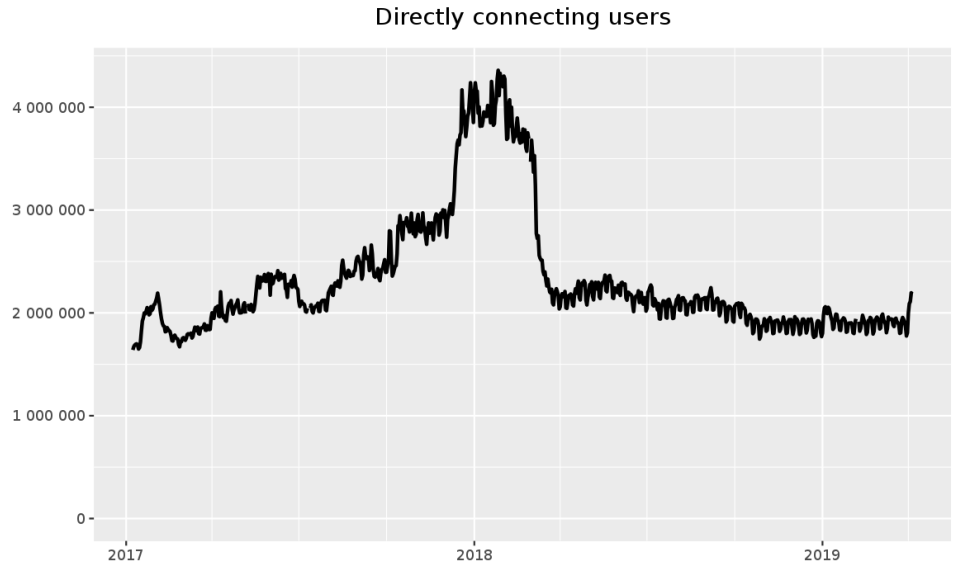
Designed for low-latency anonymity network

- Tor minimizes the use of public-key encryption
- Tor relays do not buffer or re-order packets

~7000 Tor relays

~2 million active users everyday

- Large anonymity set



Threat model

Assume an adversary who can observe some fraction of network traffic (but not all)

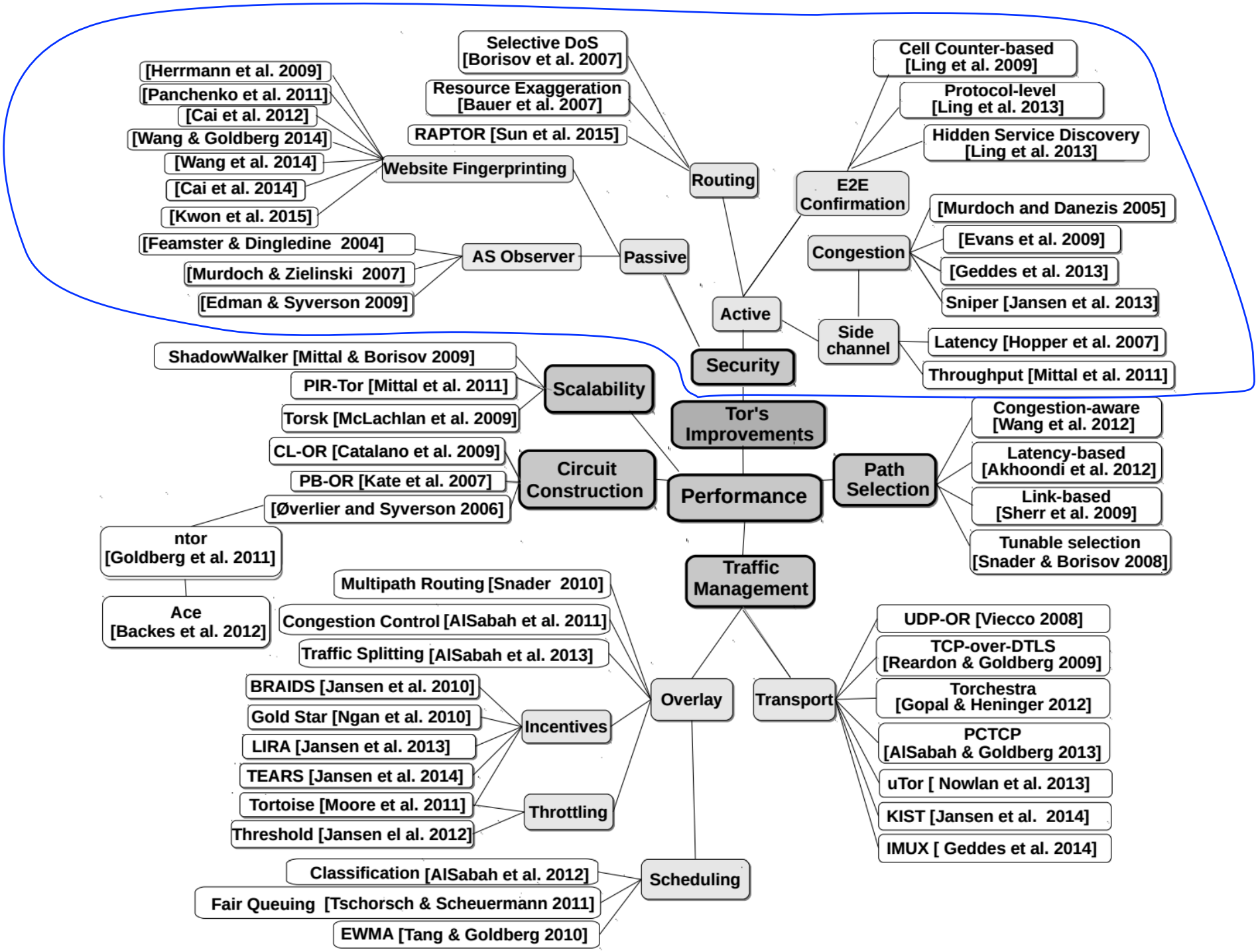
The attacker can generate, modify, delete, or delay traffic

The attacker can operate Tor relays of his own

The attacker can compromise some fraction of the relays

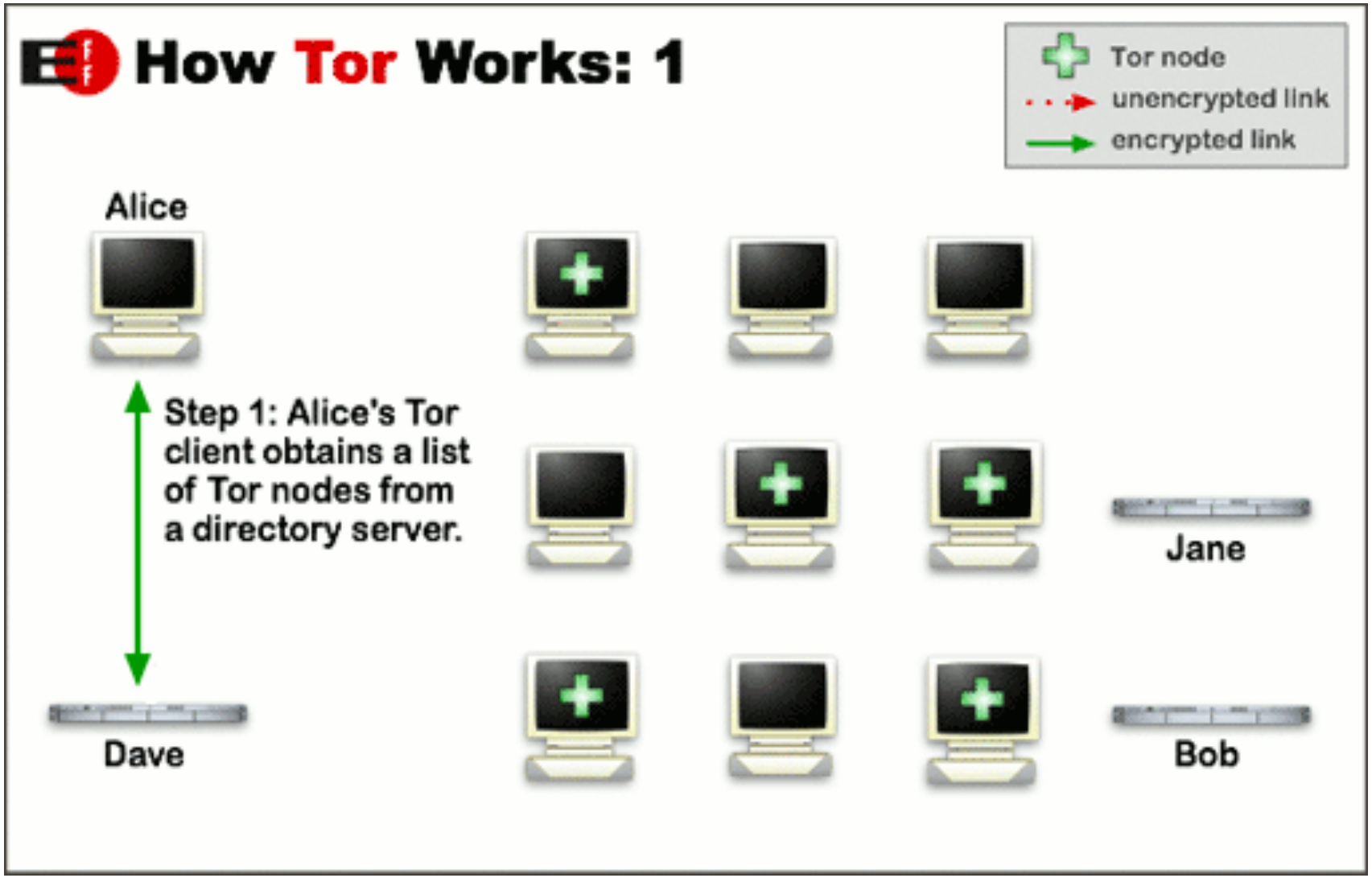
Can Tor defend against a strong attacker who can observe all traffic?

- No, and none of the low-latency systems can.

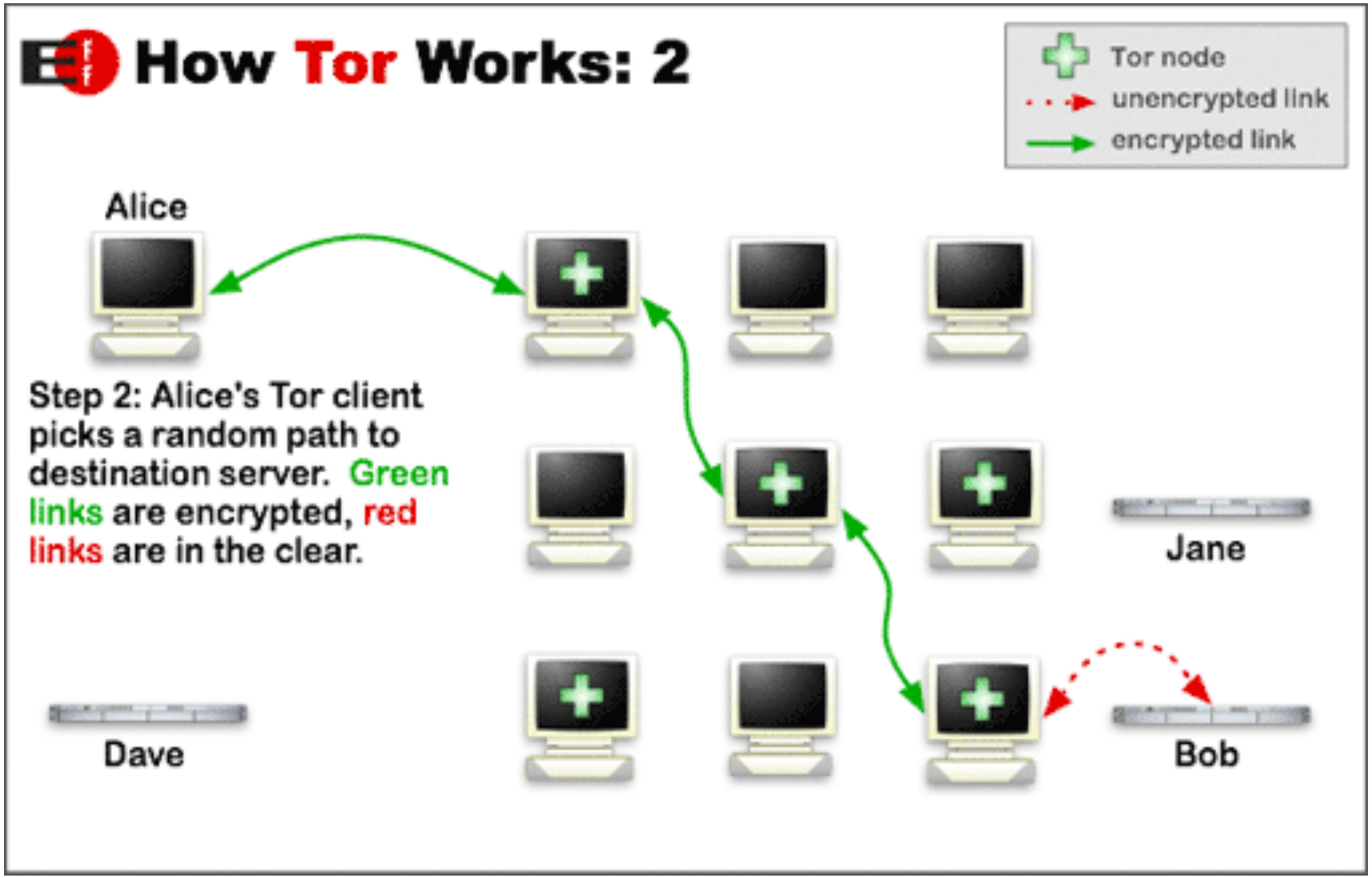


M. AlSabah and I. Goldberg, "Performance and security improvements for Tor: A survey," *ACM Computing Surveys (CSUR)* 49.2 (2016): 32.

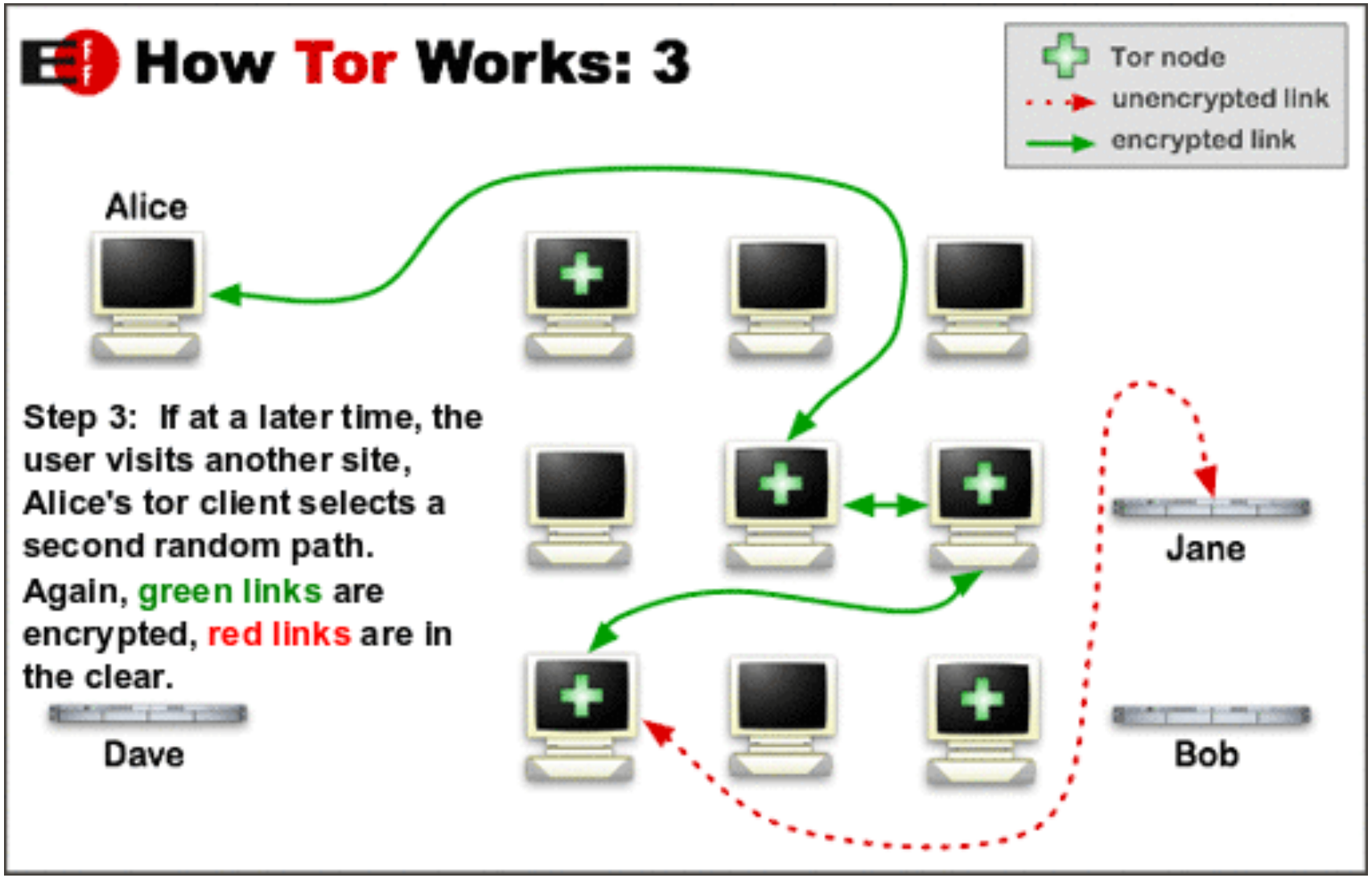
Tor overview



Tor overview

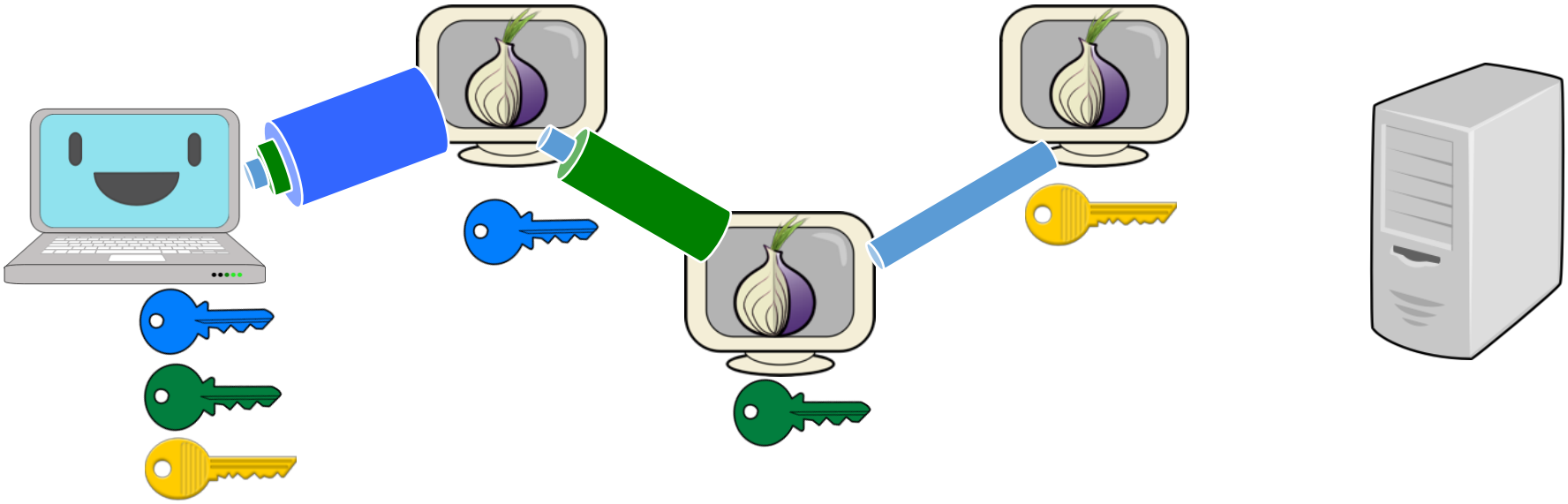


Tor overview



Tor circuit establishment

Alice's Tor client uses public-key cryptography to setup a "circuit" to Bob, establishing a pairwise symmetric key with each Tor relay on the circuit

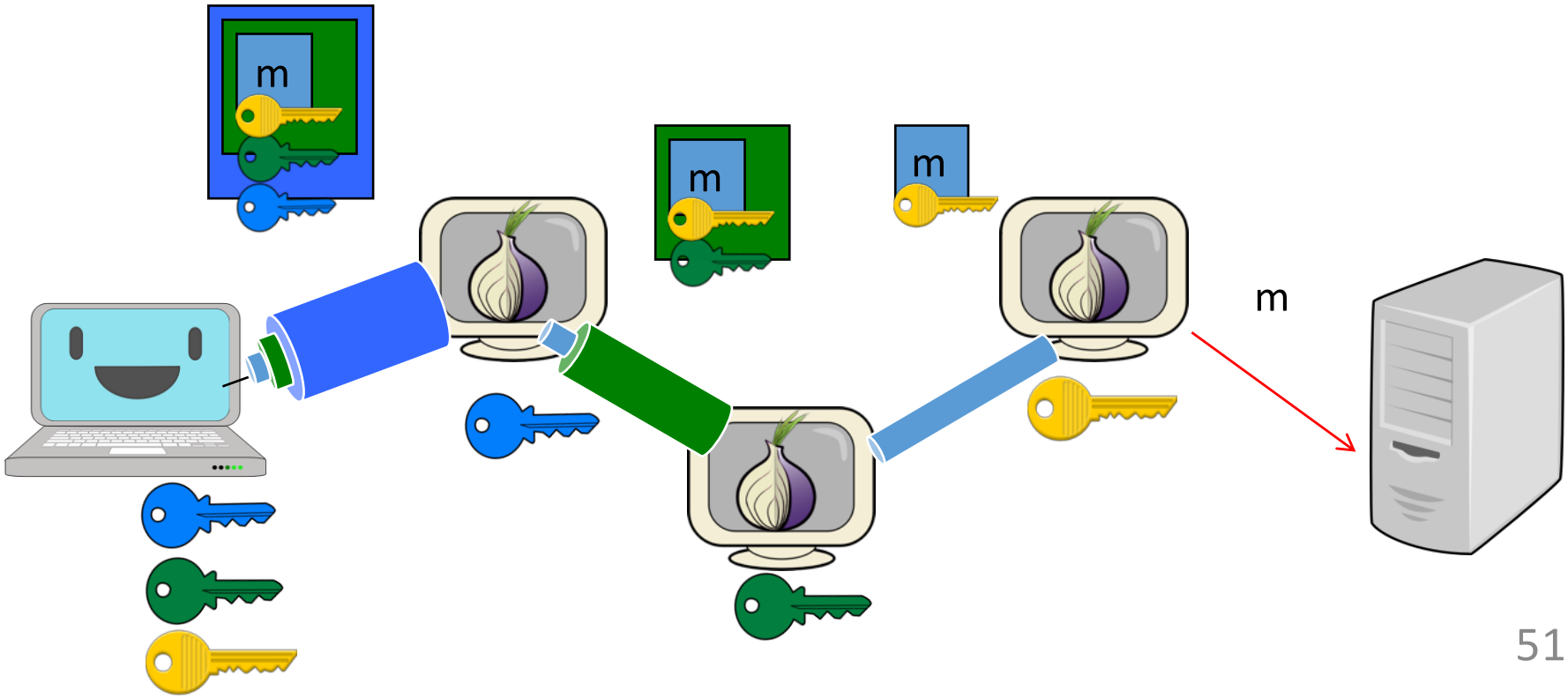


Use the established Tor circuit for anonymous communication

Layered encryption - use symmetric decryption/encryption to forward messages along the established circuit

Note: traffic should be protected using end-to-end encryption

- "Plaintext over Tor is still plaintext"



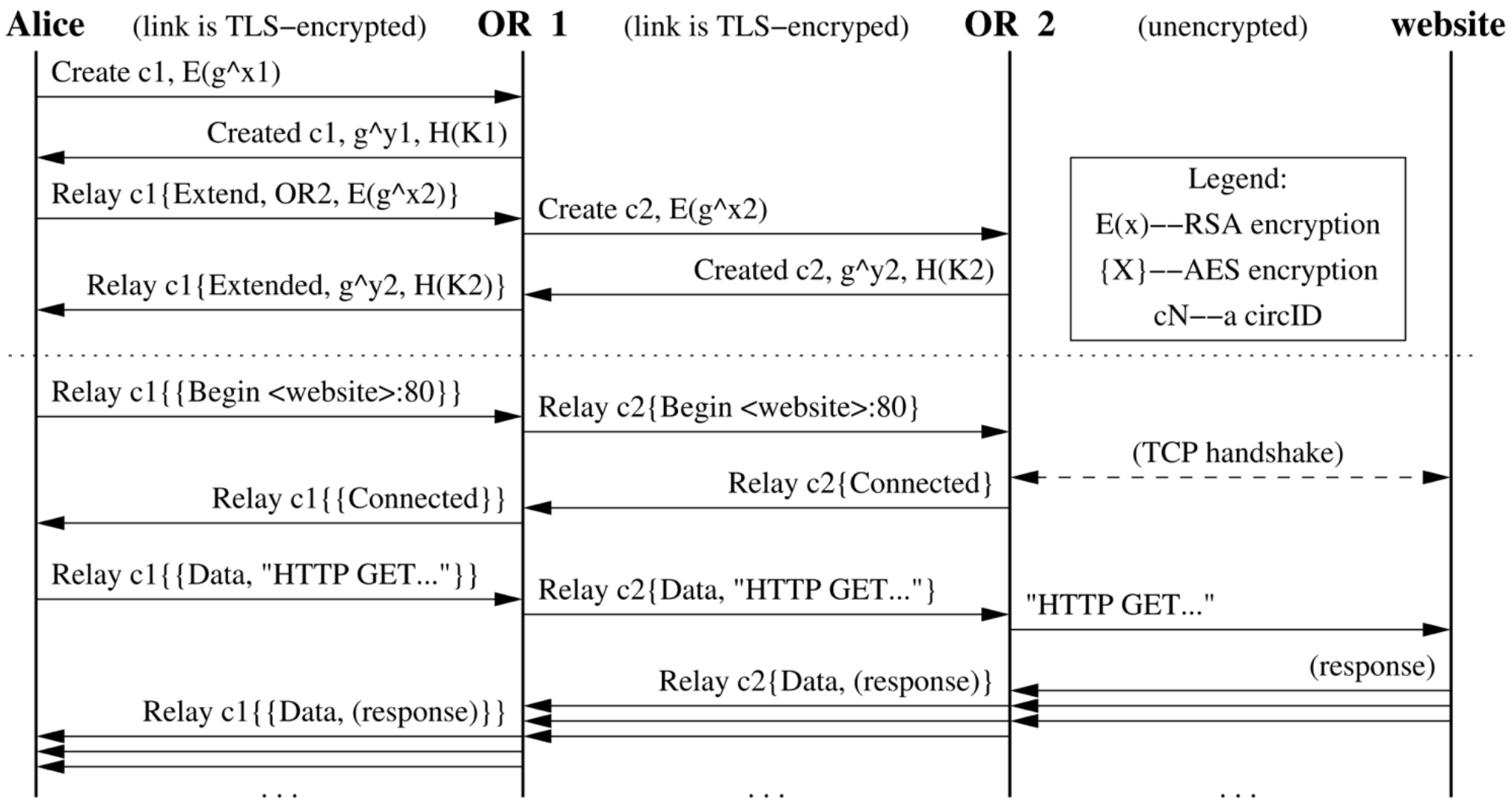


Figure 1: Alice builds a two-hop circuit and begins fetching a web page.

Tor hidden services

So far only the sender is hidden

Some servers also want to hide their IP addresses

- E.g., WikiLeaks, so that the law enforcement cannot easily trace it and take it down

Tor hidden services allow such servers to be accessible via Tor's .onion addresses only

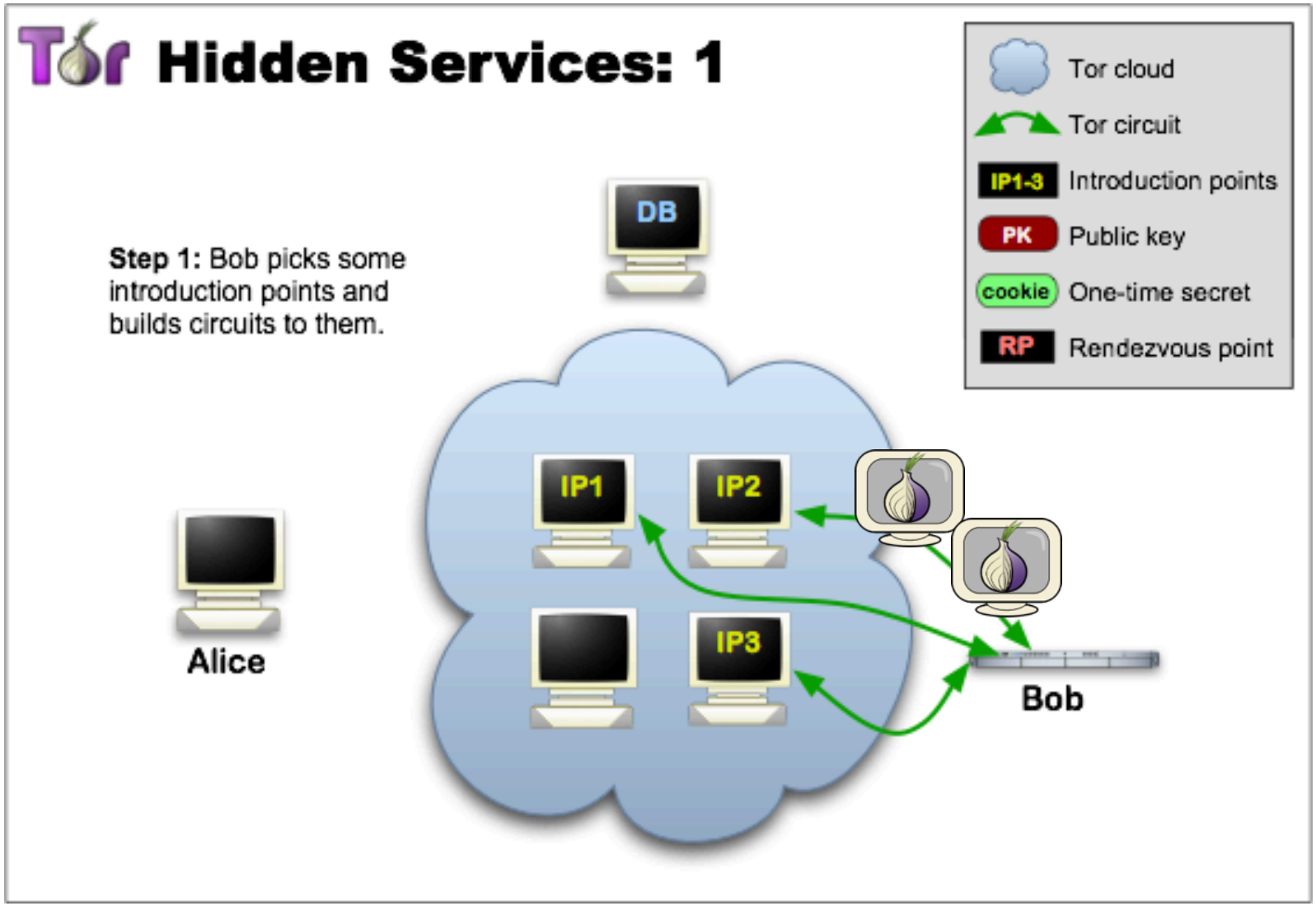
- Self-authenticating URL: the hash of the site's public key

uj3wazyk5u4hnvtk.onion

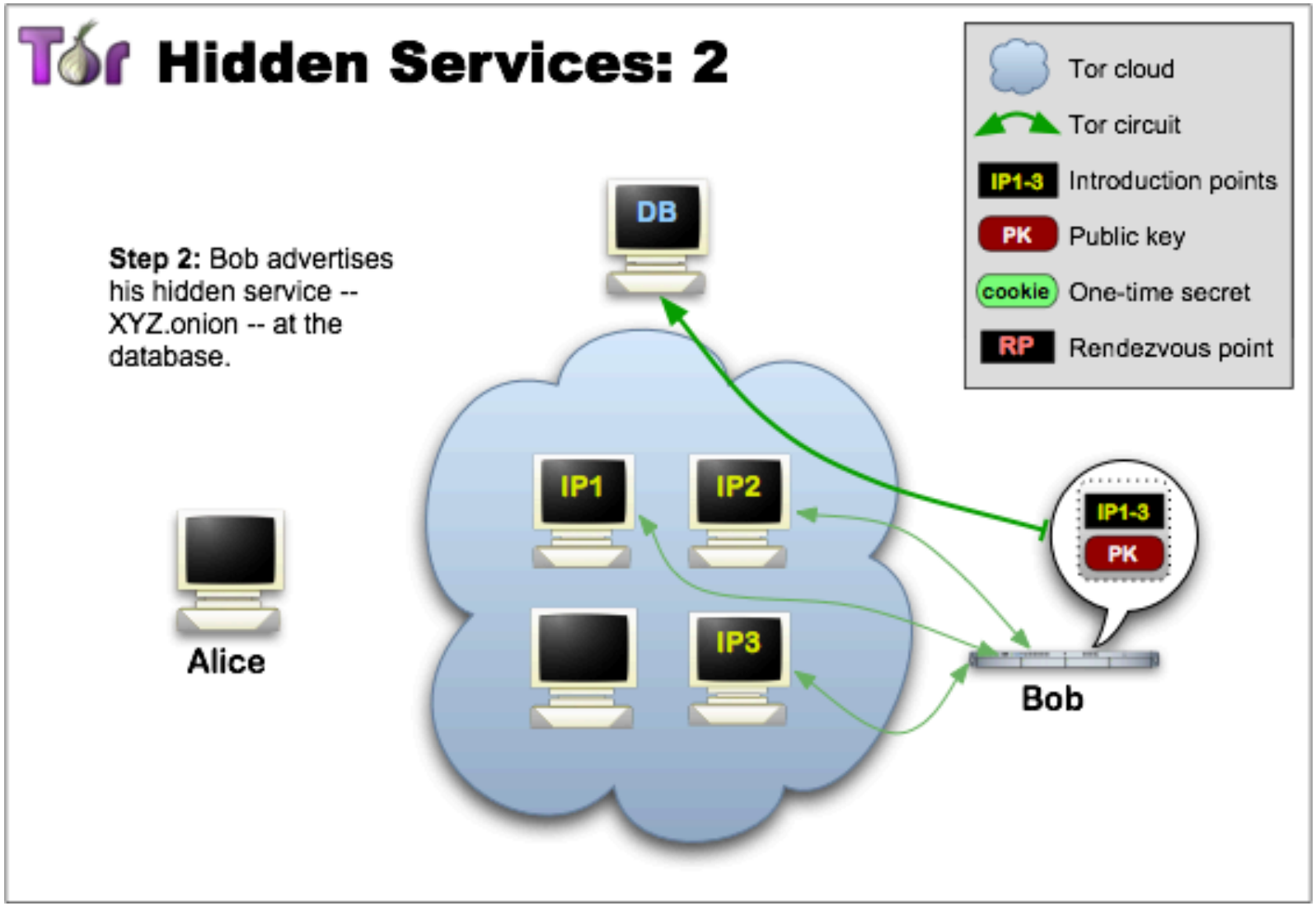
suw74isz7wqzpmgu.onion

facebookcorewwwi.onion

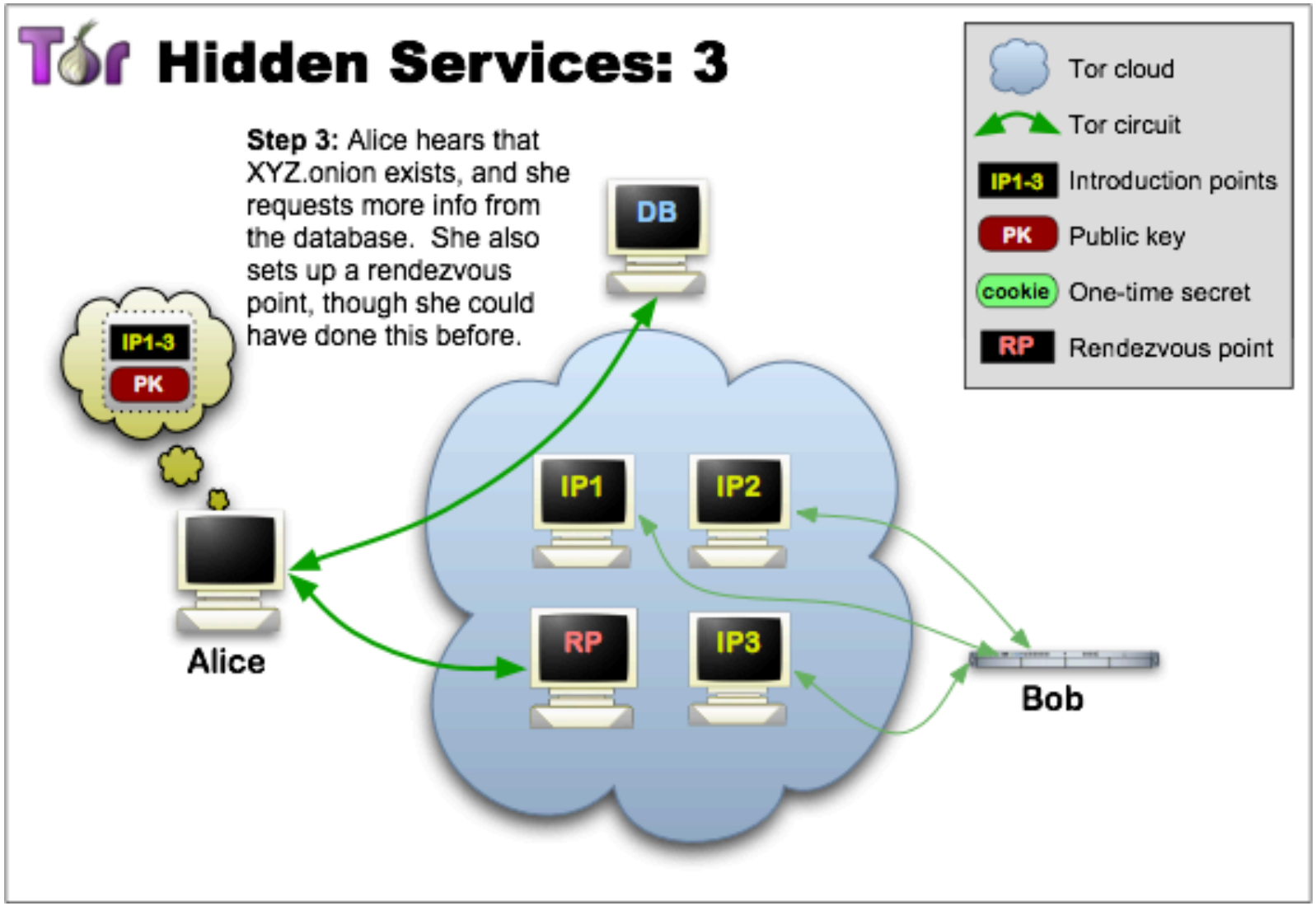
Hidden services



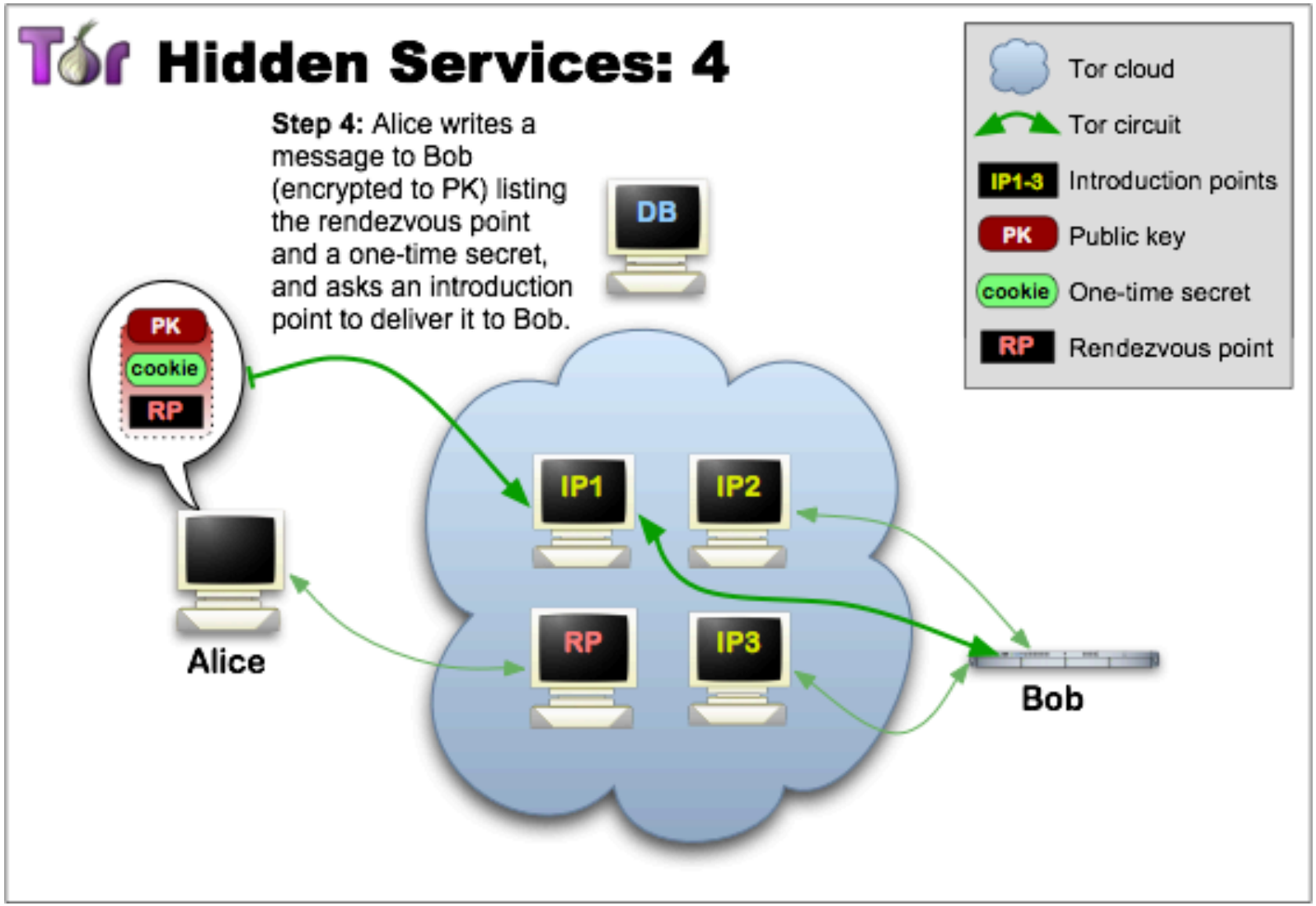
Hidden services



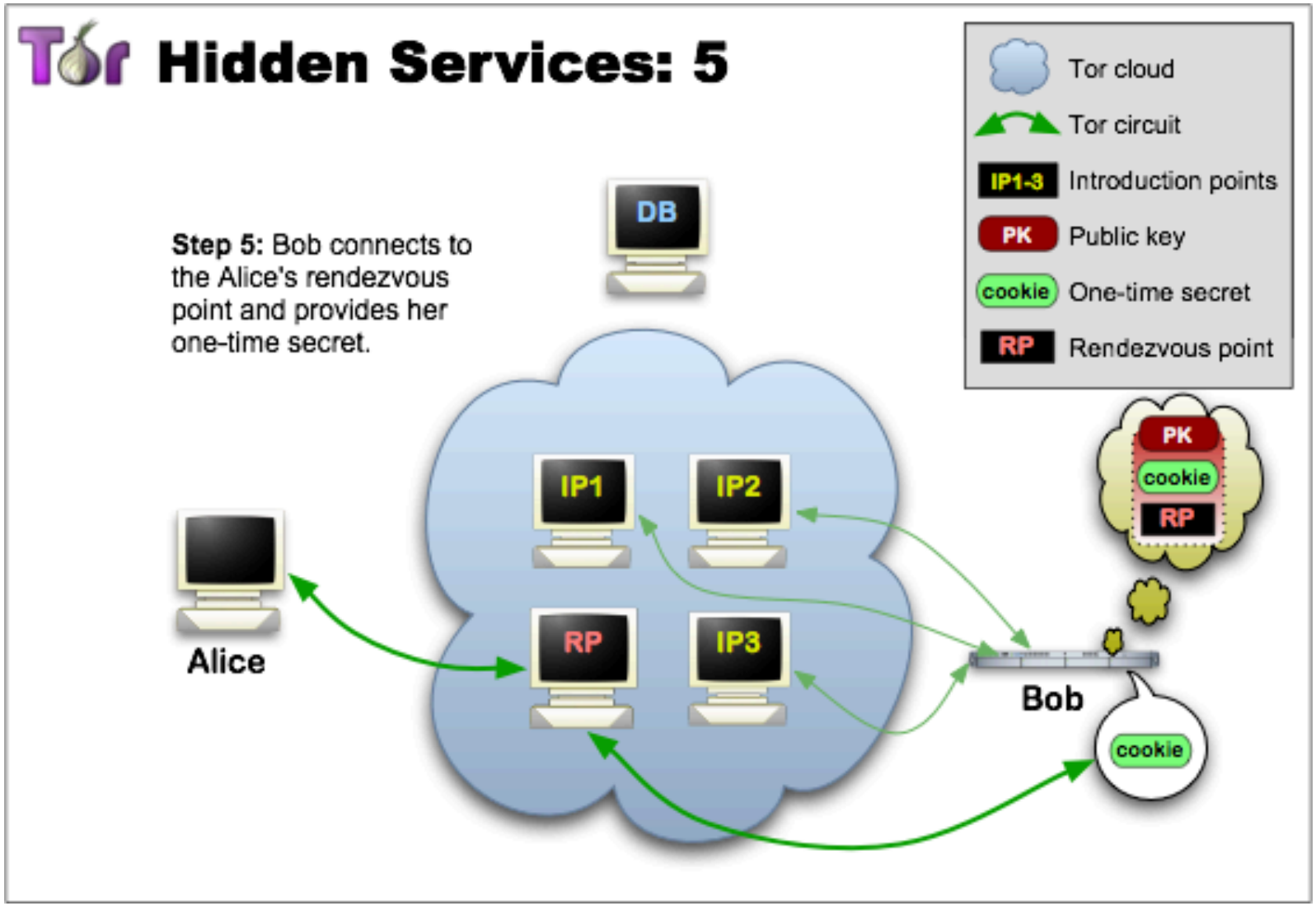
Hidden services



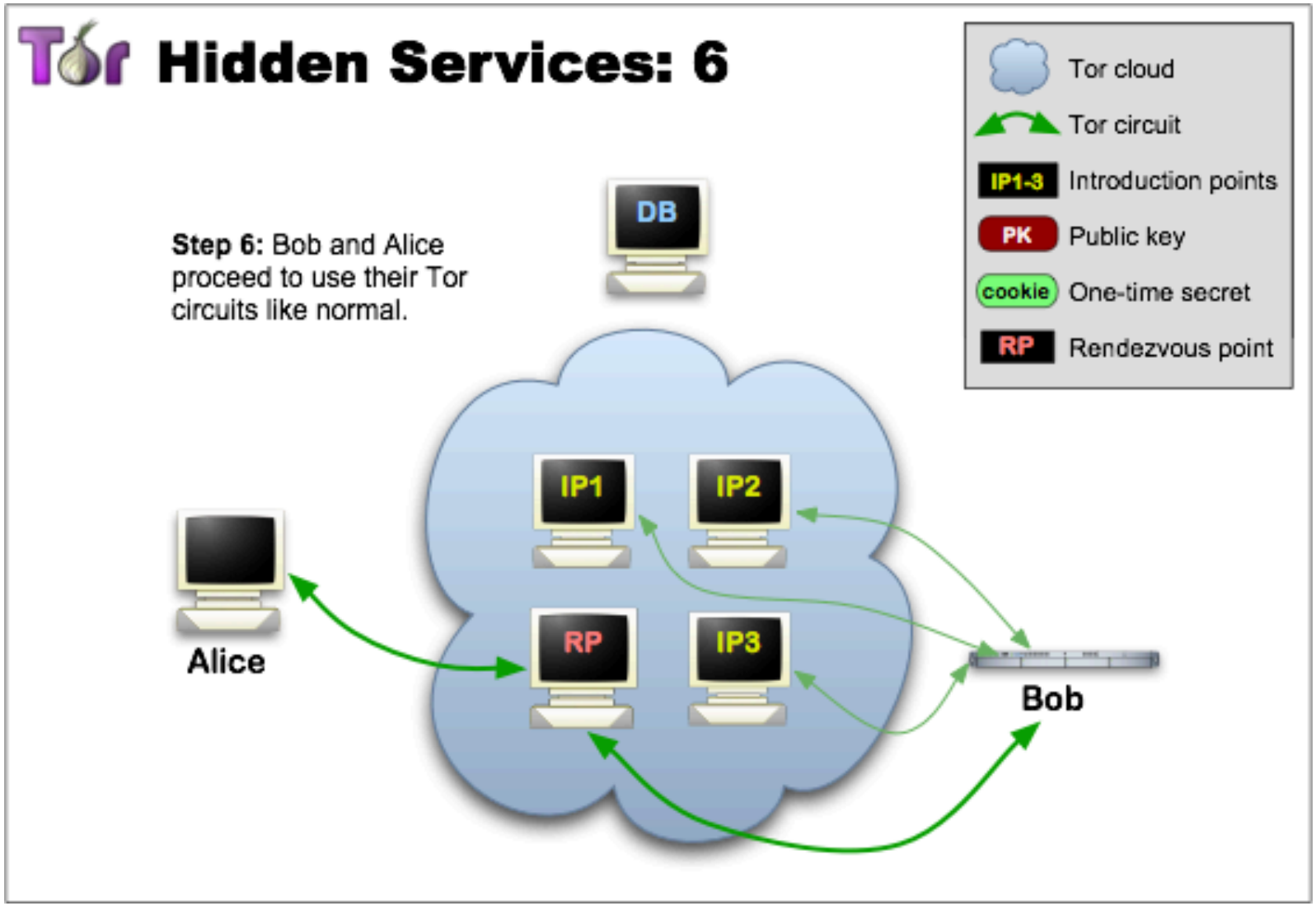
Hidden services



Hidden services

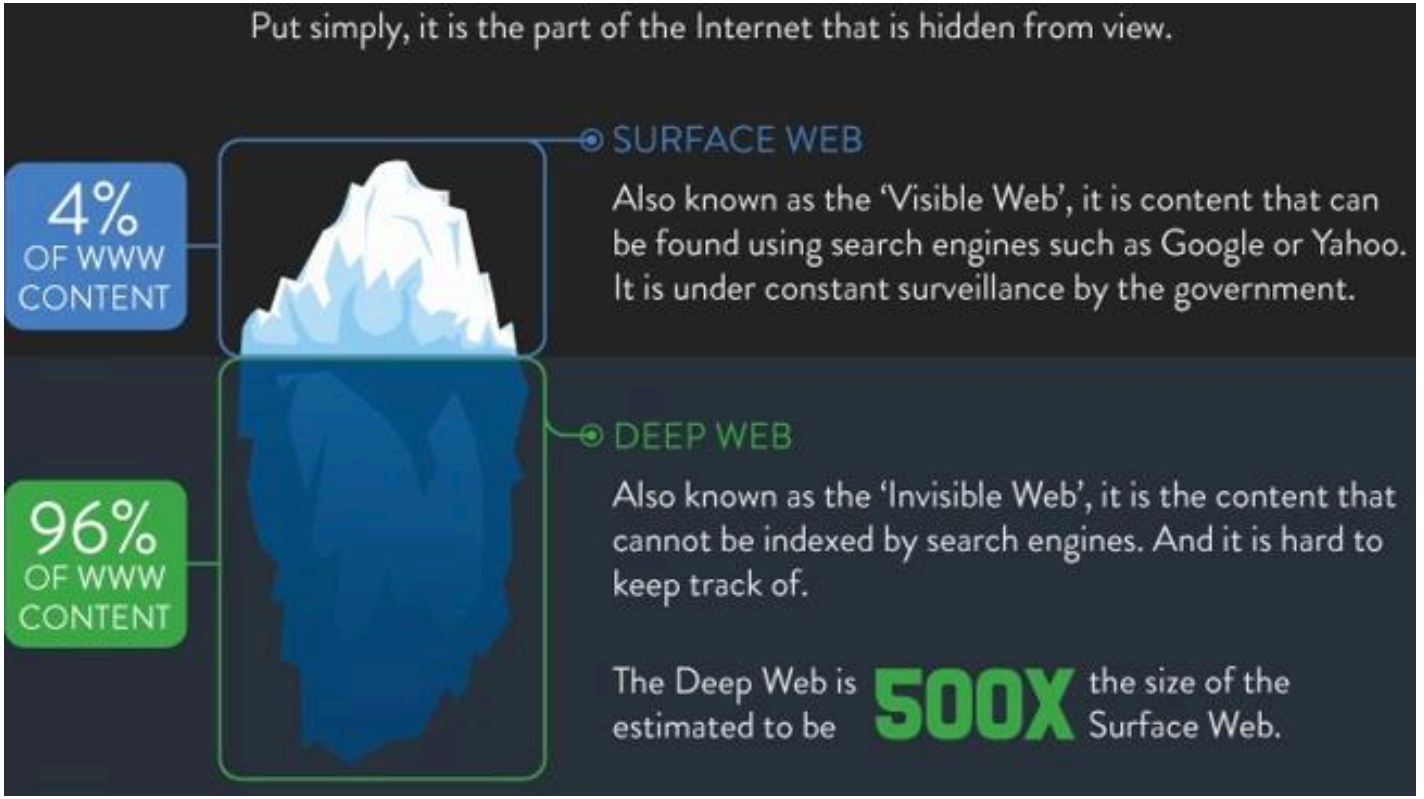


Hidden services



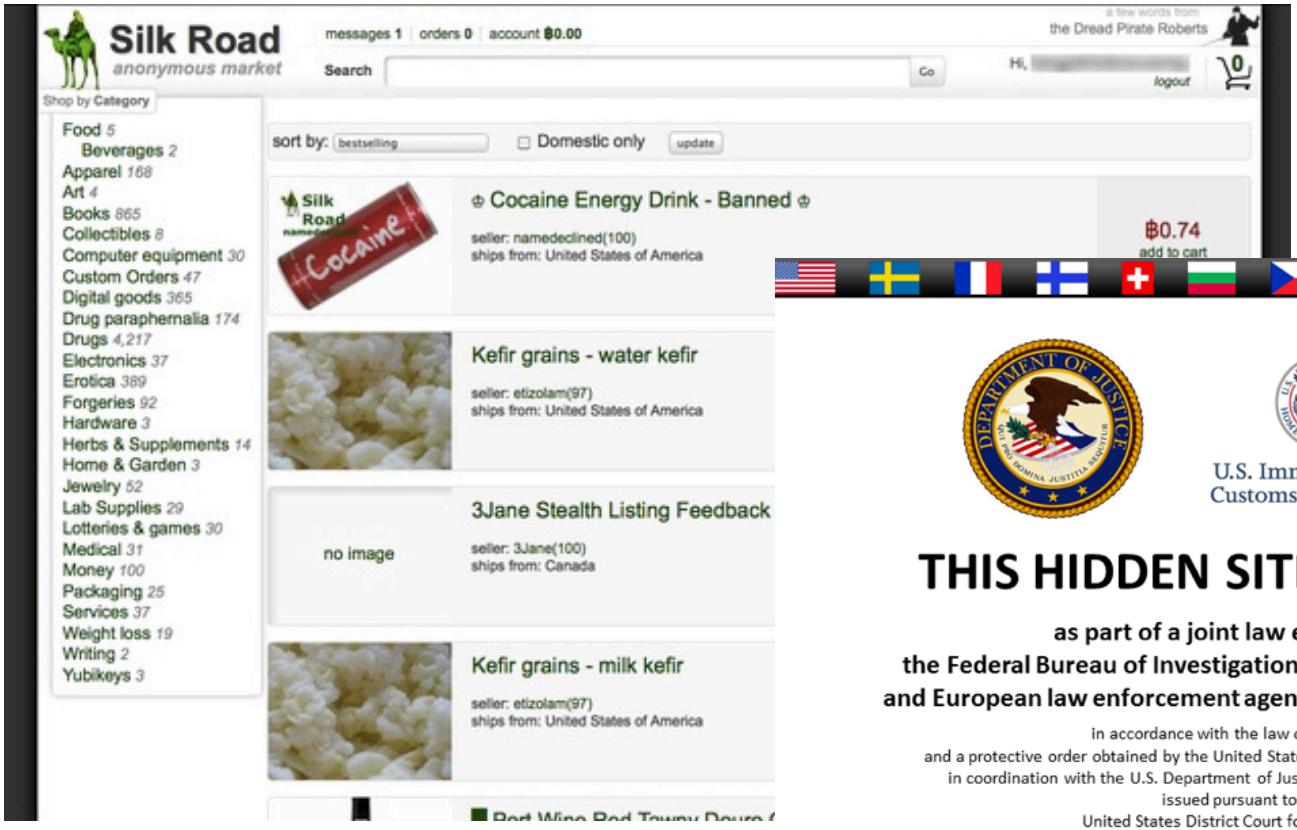
Deep web (or dark web)

Part of Internet not indexed by standard search engines



Silk Road

anonymous marketplace



Path selection in Tor

Challenge: balance performance and anonymity

Originally relays were selected uniformly at random

- -> Performance issues as Tor got popular

Relays are now weighted based on their available bandwidth to balance traffic load

- Bandwidth Authorities actively probe available BW to prevent relays from reporting false BW info

More selection criteria & suggested algorithms exist

- E.g., the use of Entry Guard (will discuss this later)
- E.g., AS-aware selection

Attacks on Tor

Many attacks proposed, indicating that Tor is successful and a high-value target

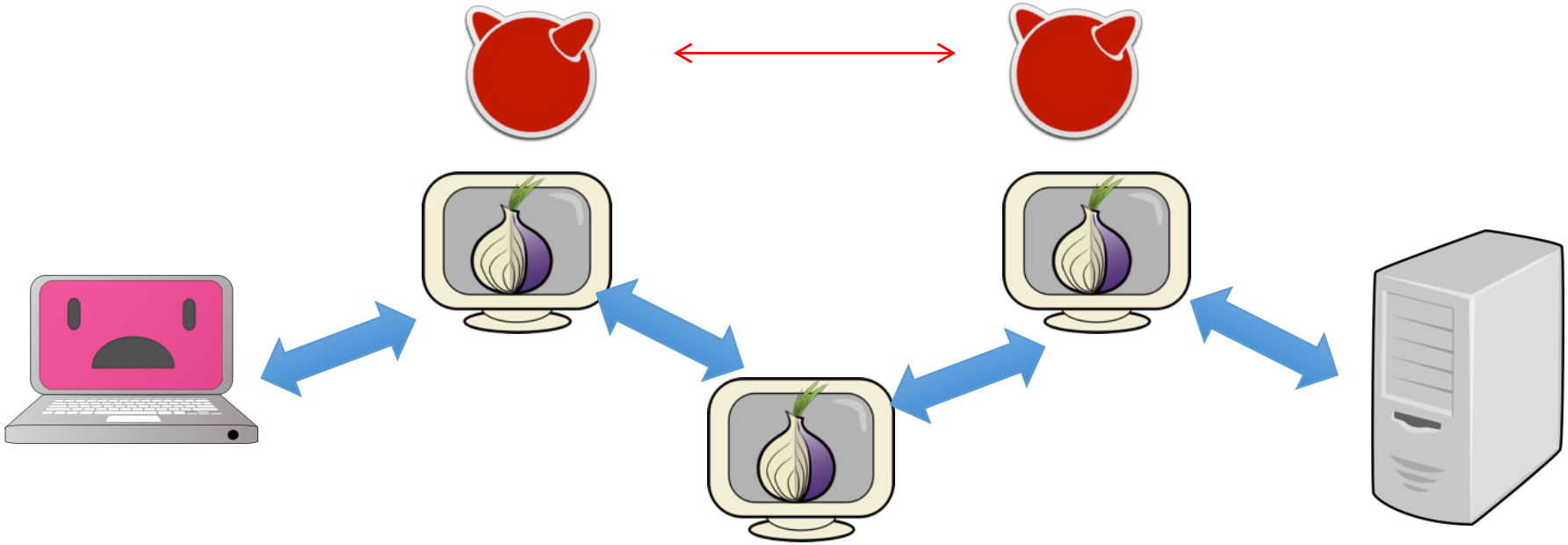
Common attack types

- Passive traffic analysis
  - "What if the attacker can watch part of the Internet?"
- Active traffic analysis
  - "What if the attacker can actively send traffic and monitor the effect?"
- Controlling some Tor relays
  - "What if the attacker runs some relays?"

Note: Tor does not protect every aspect of anonymity.

Attack by controlling the first & last relays

If the attacker controls both the first and last relays, then the attacker can easily link the sender and receiver via end-to-end correlation



Attack by controlling the first & last relays

Suppose the attacker controls *c* relays

Suppose there are *n* relays in total

first 和 last 都 g

For each circuit, attack succeeds with probability $(c/n)^2$

Worse yet, if the client randomly builds new circuits for every new connection, the probability that the client is profiled at least once increases over time.

Attack by controlling the first & last relays

Countermeasure: Entry Guards

The client uses a fixed set of relays (called entry guards) as her first hop

- Only stable and reliable relays can be used as guards
- Clients rotate their guards nodes every 4-8 weeks

If these relays are benign, then the client is secure

If an evil relay is selected as an entry guard, then the probably of being attacked is *c/n*…

Challenge: a better guard selection algorithm

Attack by manipulating guard selection

The attacker can try to run a large number of relays (i.e., a *Sybil attack*), increasing the chance of being selected as entry guards

- Sybil attack: a single entity forges multiple identities
- In 2014, 115 relays (6.4% of the Guard capacity) were linked to a several months of attack against Tor.

"If you're doing an experiment without the knowledge or consent of the people you're experimenting on, you might be doing something questionable—and if you're doing it without their informed consent because you know they wouldn't give it to you, then you're almost certainly doing something wrong. Whatever you're doing, it isn't science." Nick Mathewson, co-founder of the Tor Project.

Attack by manipulating guard selection

Countermeasure: Sybilhunter

Detects Sybil relays based on appearance and behavior, such as configuration and uptime sequences.

- P. Winter, R. Ensafi, K. Loesing, and N. Feamster, "Identifying and characterizing Sybils in the Tor network," 2016.
- https://www.nymity.ch/sybilhunting/

Challenge: accurate detection

Attack against Tor's availability

Tor relays are listed on the public directory

Adversarial ISPs can deny access by filtering connections to/from all publicly listed Tor relays

- You're "censored"

Countermeasure: Tor Bridges

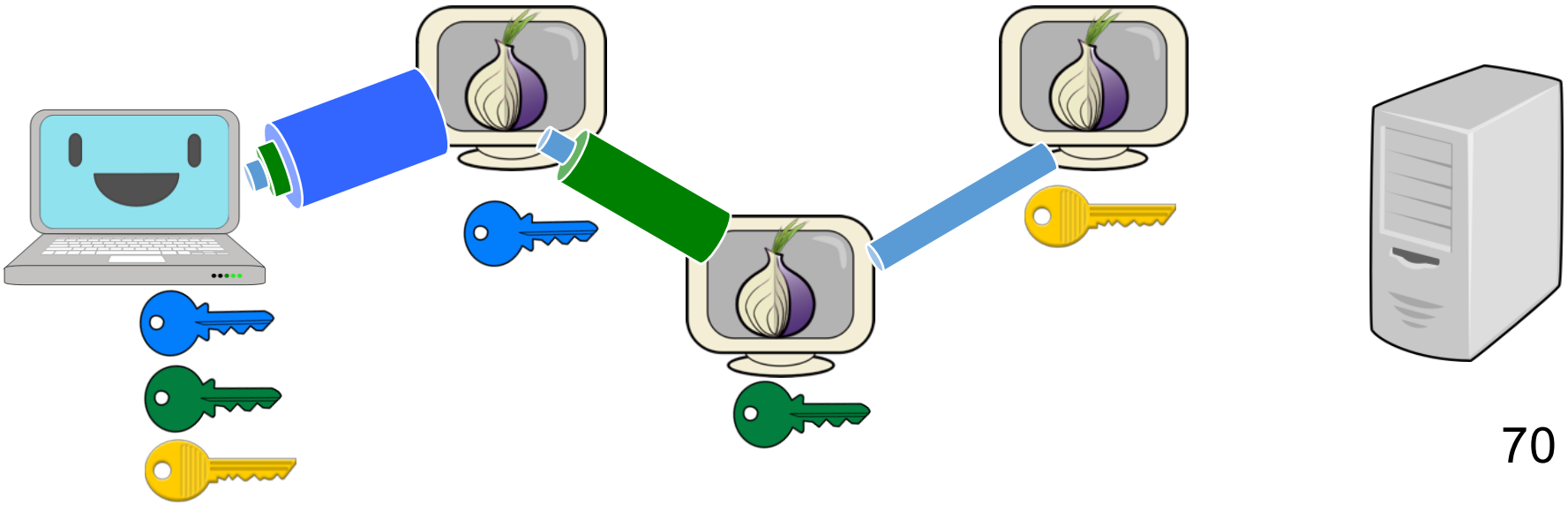
- Tor bridges are relays that are not listed publicly

Challenge: how to locate bridges in the first place?

Tor vs. Mix networks

Similar to mix networks, but no mixing and less public key crypto to reduce latency

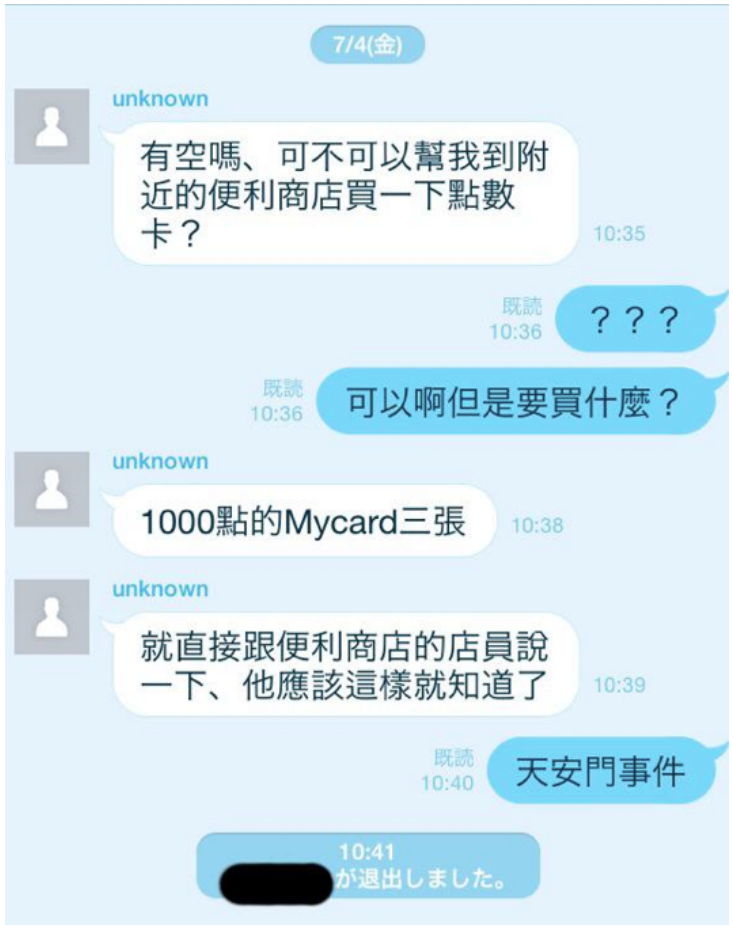
Suspect to timing attacks (like all low-latency anonymity systems)



Censorship Circumvention

Internet censorship

“Internet censorship is the control or suppression of what can be accessed, published, or viewed on the Internet.”




DNS: 8.8.8.8 kuşun ötsün!
Alternatif: 8.8.4.4

Censor's capabilities

Censor's capabilities	Seen
DNS injection	China 2007 [104], 2011 [88], China 2014 [91]; Pakistan 2010 [106], 2013 [80]; Iran 2013 [79]
HTTP injection	Pakistan 2013 [80]
TCP RST injection	China 2006 [82], China 2010 [89]
Packet dropping	Iran 2013 [79], China 2015 [76],
Stateless	China 2002 [77], 2006 [82]
Stateful	China 2007 [84], China 2012 [87], China 2013 [78]
Packet reassembly	China 2013 [78]
Using Netsweeper	Pakistan 2013 [100], Qatar 2013 [101], UAE 2013 [101], Yemen 2013 [101]
Using Blue Coat	Syria 2011 [95, 107]; Burma 2011 [101]; UAE 2013 [101], Qatar 2013 [101]
Using SmartFilter	Iran 2004 [108], Qatar 2013 [101], Saudi Arabia 2012 [101], UAE 2013 [101]

M. C. Tschantz, S. Afroz, D. Fifield, and V. Paxson, "SoK: Towards Grounding Censorship Circumvention in Empiricism," in Proceedings of IEEE Symposium on Security and Privacy, 2016.

Why circumvention is hard?

Powerful censors

- State-level censors see and control countrywide network

An arms race between censorship and circumvention

- Recall the use of bridges in Tor: How can we distribute the list of bridges to **censored individuals only** (without letting the censors know)?

M. C. Tschantz, S. Afroz, D. Fifield, and V. Paxson, "SoK: Towards Grounding Censorship Circumvention in Empiricism," in *Proceedings of IEEE Symposium on Security and Privacy*, 2016.

vs. The Great Firewall (GFW)

Began in operation **2003**

Deployed mirrors & email-based distribution

Introduced *bridges* (secret relays)

2008 Blocked www.torproject.org

2009 Blocked public relays

Enumerated bridges from centralized bridge databases

2011 Deployed *Deep Packet Inspection* (DPI) to identify Tor's distinctive protocol features (E.g., list of TLS client cipher suites)

made Tor's use of TLS less distinctive

Introduced *pluggable transports*, an additional layer encapsulating Tor TLS

The latest pluggable transports are designed to resist active probing

Supported domain fronting (meek) **2014**

Employed *active probing*, posing as a user to test suspected forwarders

2018 Google, Amazon disabled support of domain fronting

M. C. Tschantz, S. Afroz, D. Fifield, and V. Paxson, "SoK: Towards Grounding Censorship Circumvention in Empiricism," in Proceedings of IEEE Symposium on Security and Privacy, 2016.

Censorship resistance strategies for non-blocking communication

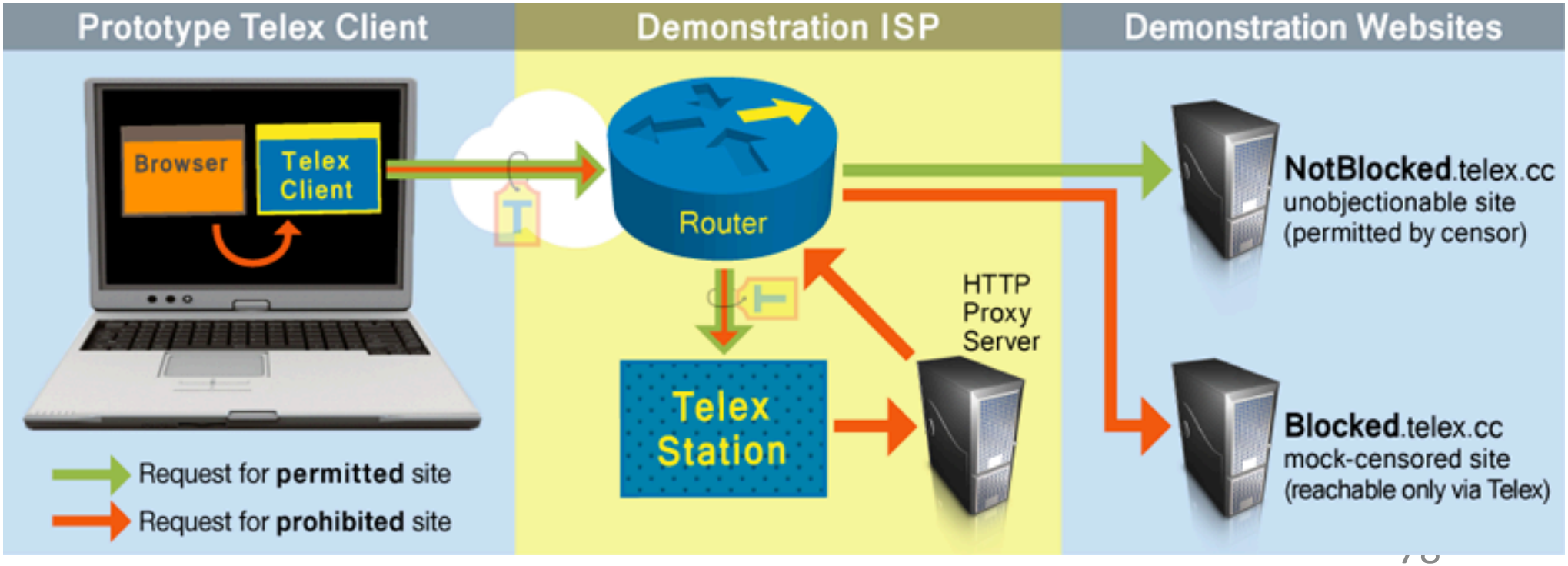
1. Use encrypted channels or proxies, and try keeping proxies available from being completely blocked
   - E.g., Tor with bridges
2. Leverage steganography and covert channels
   - Encode messages in seemingly innocuous pictures or emails
3. Cause collateral damage to the censor
   - Hide among innocents, mingle with important applications, leverages censors' unwillingness to completely block day-to-day Internet access
   - E.g., decoy routing, domain fronting
   - Don't work for irrational adversaries (e.g., the Russian government vs. Telegram)

Decoy Routing

https://www.decoyrouting.com/

Can be seen as locating proxy functionality in the core of the network

Blocking such functionality would cause expensive collateral damage



Domain Fronting

Different domain names are used at different layers of communication

Blocking the front domain entirely will cause expensive **collateral damage**

Google and Amazon disabled support of domain fronting in 2018

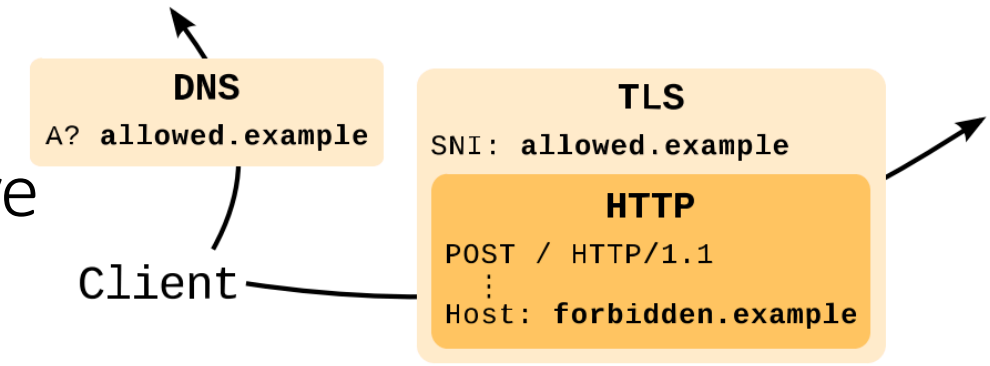


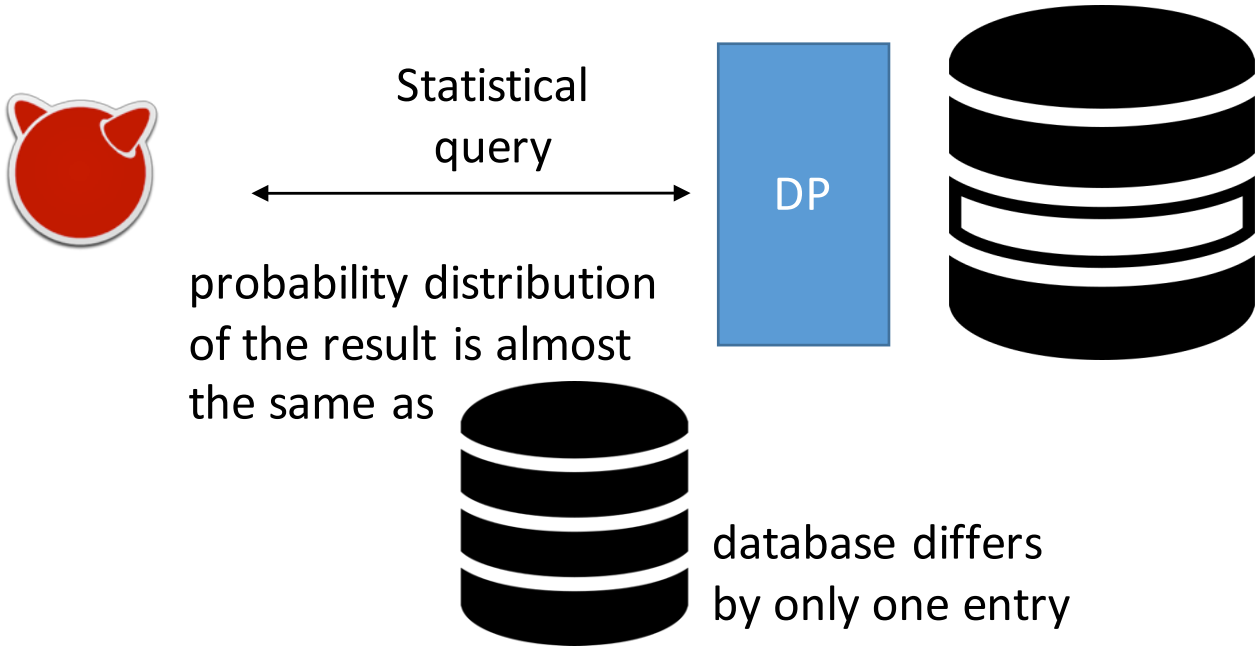
Fig. 1. Domain fronting uses different domain names at different layers. At the plaintext layers visible to the censor—the DNS request and the TLS Server Name Indication—appears the front domain **allowed.example**. At the HTTP layer, unreadable to the censor, is the actual, covert destination **forbidden.example**.

D. Fifield, C. Lan, R. Hynes, P. Wegmann, and V. Paxson, "Blocking-resistant communication through domain fronting," in Proceedings on Privacy Enhancing Technologies, 2015.

More on Privacy Definitions

Differential Privacy

A kind of privacy definition

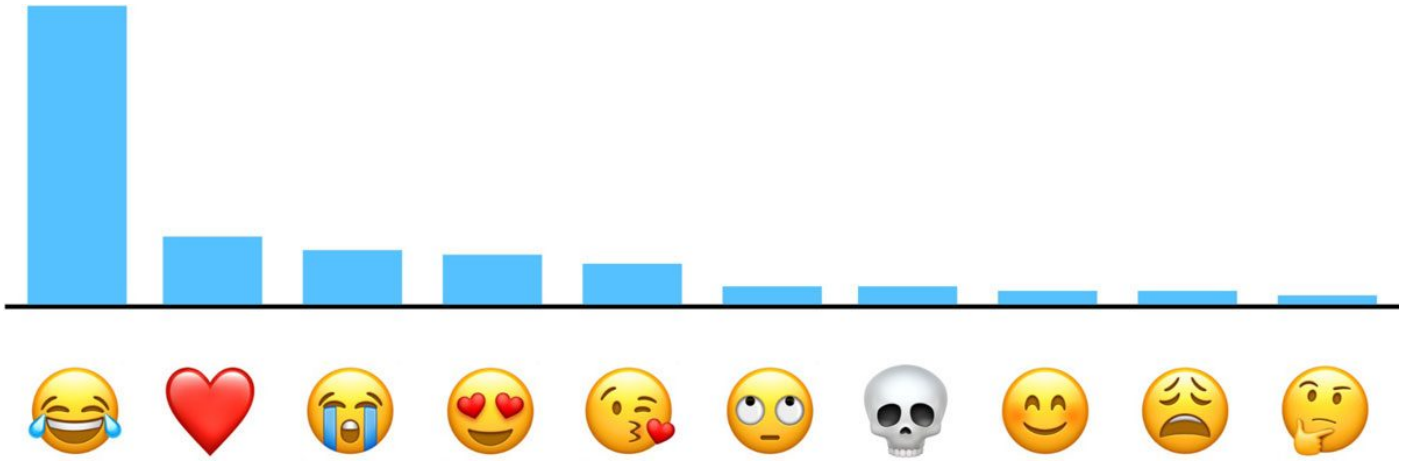


C. Dwork, "Differential privacy." *Encyclopedia of Cryptography and Security* (2011): 338-340.
https://blog.cryptographyengineering.com/2016/06/15/what-is-differential-privacy/

Differential Privacy

Example: Apple wants to know the most popular emoji among iPhone users

Naïve solution (no privacy): each user reports daily usage as an array, e.g., [2, 0, 0, 0, 1, 0, 0, 3, 0, 1]



https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf

Differential Privacy

Example: Apple wants to know the most popular emoji among iPhone users

A DP approach: adds *noise* to the report in order to obfuscate individual preferences but allows the aggregate count to be close to the real count, e.g., [2, 0, 0, 0, 1, 0, 0, 3, 0, 1] -> [1, 4, 2, 0, 3, 0, 1, 3, 5, 2]

A tradeoff between privacy and accuracy

https://www.apple.com/privacy/docs/Differential_Privacy_Overview.pdf

Differential Privacy


How to add noise?

A DP approach based on *randomized response*:

- Say Google wants to know whether users use more Google or Apple style of emoji daily
- To report, each user flips a coin:
  - If coin = 1, return a random answer (50% “Yes” and 50% “No”)
  - Otherwise, return the true answer
- Compute the answer by compensating the noise
- Q: Suppose 30% responded “Yes”, what should be the actual percentage approximately?

Erlingsson, Úlfar, Vasyl Pihur, and Aleksandra Korolova. "Rappor: Randomized aggregatable privacy-preserving ordinal response." *ACM CCS*, 2014.

Review:
Terminology checklist

Terminology checklist -I

Security requirements

- ~~Confidentiality~~
- ~~Integrity~~
- ~~Availability~~
- Anonymity
- Authenticity
- ~~Non-repudiation~~ Digital signature schema

Terminology checklist -II

Attacks

- ~~Brute-force search~~
- ~~Dictionary attack~~
- Man-in-the-middle attack
- Replay attack
- Reflection attack
- Length extension attack against Merkle-Damgård construction
- Birthday paradox

Unconditionally secure (information-theoretically secure) vs. computationally secure

Terminology checklist -III

Public-key cryptography

- ~~Merkle puzzles~~
- Diffie-Hellman key exchange
- Public-key encryption (RSA, ElGamal)
- Digital signature (RSA)

Key distribution and management

- Public Key Infrastructures (PKI)
- Digital certificates
- Certificate authorities
- Secret sharing

Terminology checklist -IV

Symmetric-key cryptography

- Symmetric-key encryption
  - One-time pad
  - Block ciphers
  - Mode of operation (ECB, CBC, …)
- Message authentication code (MAC)
- MAC-then-encrypt, encrypt-then-MAC

Terminology checklist -V

Cryptographic hash function (unkeyed symmetric)

- Secure hash function properties
  - Pre-image resistance
  - Weak collision resistance
  - Strong collision resistance
- Hash function applications
  - Merkle hash trees
  - One-way hash chains
  - Password hashing
  - Commitment

Terminology checklist -VI

Entity authentication

- Two-factor authentication
- Something you have
- Something you know
- Something you are
- Challenge-response protocols
- Nonce

Terminology checklist -VII

Anonymity

- Anonymity set
- Sender anonymity and receiver anonymity
- DC-nets
- Mix networks
- Tor
- Domain fronting
- Differential privacy