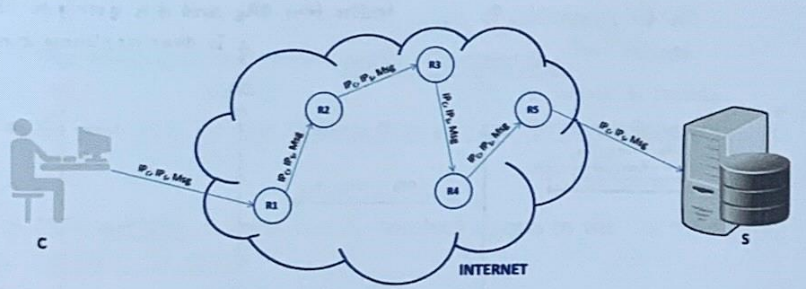
12

ONION ROUTING & TOR

Routing and privacy



- Internet routing exposes user's privacy (meta-data like IPs)
- All routers on the path between source and destination, know the origin and destination of forwarded packets
- Core internet routers are managed by governments and big corporations (so they can observe a large fraction of Internet activity) ↳ can be used to profile users

Today's lecture

Tor

Stand up for privacy and freedom online

[R. Dingledine, N. Mathewson, and P. F. Syverson: "Tor: The Second-Generation Onion Router", USENIX Security Symposium 2004]

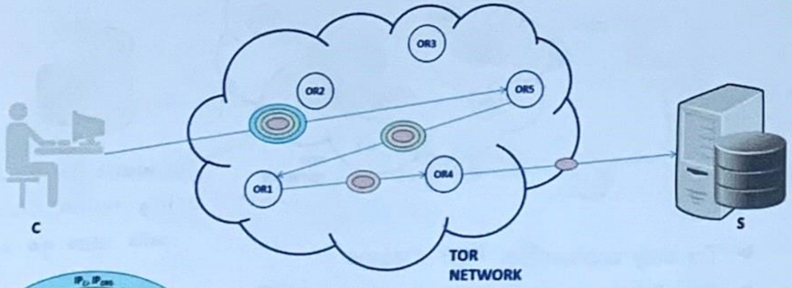
Idea: combine advantages of mixes and proxies

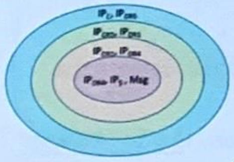
- use public-key crypto only to establish circuit (bc. pk crypto is expensive)
- use symmetric-key crypto to exchange data
- distribute trust like mixes
- do not delay or batch like mixes (low-latency)

But does not defend against adversary that observes the whole network (unlike mixes)

Tor's main ingredient: the onion

OR – Onion Router

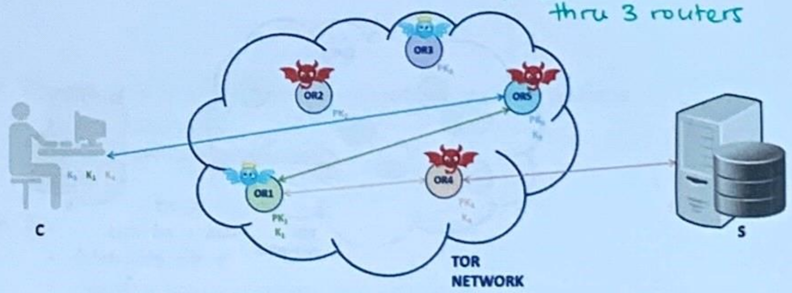




The idea of this onion is this encryption in layers; and it can be seen how the encrypted layers are getting stripped as it traverses the TOR network until it reaches the destination.

Tor circuit setup

Typically goes thru 3 routers

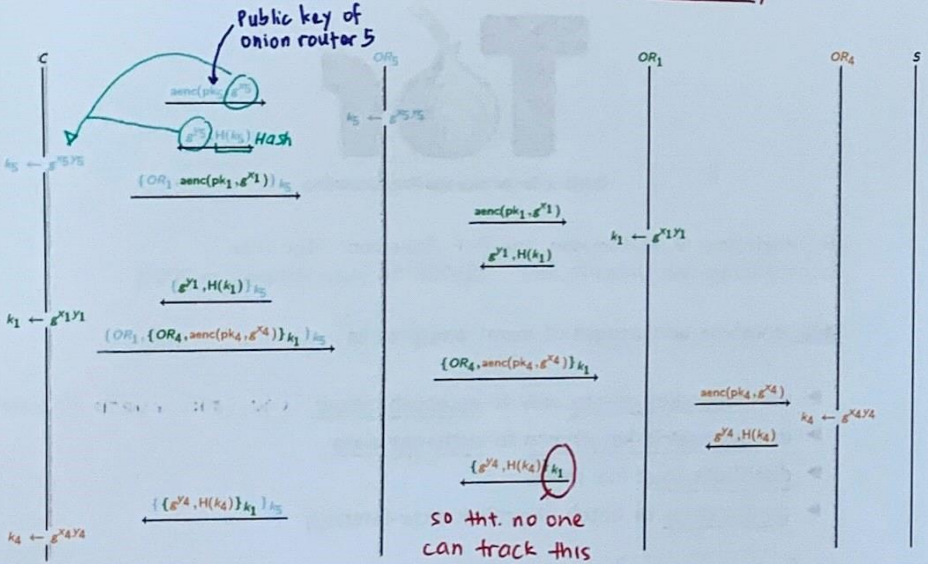


- C establishes session key K_5 and circuit with Onion Router OR_5
- C tunnels through that circuit to extend to Onion Router OR_1 → C will also agree on session key K_1 w/ OR_1
- C tunnels through that extended circuit to extend to Onion Router OR_4 → K_4; Note OR_5 has no idea which relay OR_1 is going to extend the circuit to
- Client applications connect and communicate of established Tor circuit
- A single honest Onion Router on the Tor circuit guarantees anonymity against an attacker controlling some Onion Routers

aenc - asymmetric encryption.

The (simplified) Tor message flow - circuit setup

session key is now set up

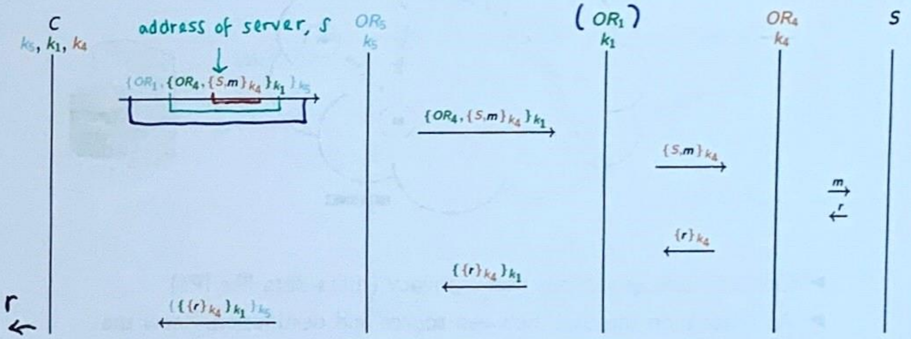


Public key of onion router 5
C
OR_5
OR_1
OR_4
S
$\mathrm{aenc}(pk_5, g^{x_5})$
$k_5 \leftarrow g^{x_5 y_5}$
$g^{y_5}, H(k_5)$
Hash
$k_5 \leftarrow g^{x_5 y_5}$
$\{OR_1, \mathrm{aenc}(pk_1, g^{x_1})\}_{k_5}$
$\mathrm{aenc}(pk_1, g^{x_1})$
$k_1 \leftarrow g^{x_1 y_1}$
$g^{y_1}, H(k_1)$
$\{g^{y_1}, H(k_1)\}_{k_5}$
$k_1 \leftarrow g^{x_1 y_1}$
$\{OR_1, \{OR_4, \mathrm{aenc}(pk_4, g^{x_4})\}_{k_1}\}_{k_5}$
$\{OR_4, \mathrm{aenc}(pk_4, g^{x_4})\}_{k_1}$
$\mathrm{aenc}(pk_4, g^{x_4})$
$k_4 \leftarrow g^{x_4 y_4}$
$g^{y_4}, H(k_4)$
$\{g^{y_4}, H(k_4)\}_{k_1}$
$\{\{g^{y_4}, H(k_4)\}_{k_1}\}_{k_5}$
$k_4 \leftarrow g^{x_4 y_4}$

so tht. no one can track this message as it goes thru. the network

The (simplified) Tor message flow - actual communication

only knows tht it is getting traffic frm OR_5 and it is going to OR_4. It does not know c or s.

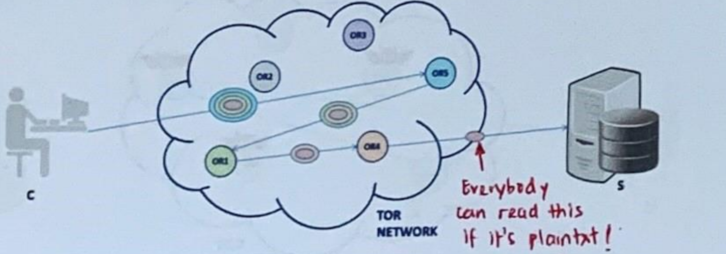


Tor provides privacy of not being able to link the client & servers and know which websites people are visiting. It breaks this link by jumping over multiple proxies and using onion encryption.

Tor only provides privacy - not confidentiality

✓ PRIVACY

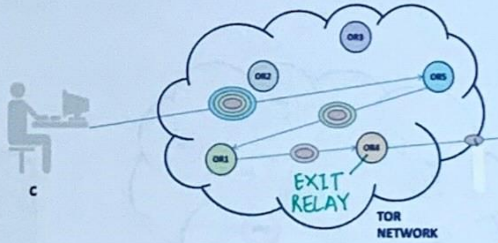
X CONFIDENTIALITY



Everybody can read this if it's plaintxt!

- Tor anonymises the origin of the traffic
- Tor encrypts everything inside the Tor network
- but Tor DOES NOT encrypt all traffic through the Internet
- for confidentiality you still need to use end-to-end encryption such as SSL/TLS

Tor takes care of DNS resolution



We want to pack our DNS query using onion encryption and it will also go over the circuit

- Tor only anonymises TCP streams
- But, DNS resolution is executed over UDP
- So, DNS resolution if handled by the client browser defeats the purpose of using Tor
- To avoid privacy breaches due to DNS resolution, the Tor browser delegates DNS resolution to the exit node

In the e.g. above, OR4 will do the DNS lookup for us and the result is sent back to client

Avoiding censorship

PROBLEM

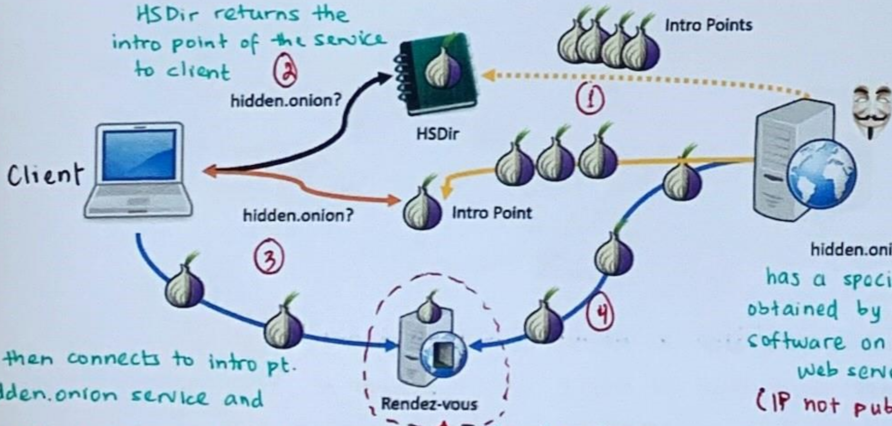
- Tor relays are listed on the public Tor directory
  has IP addresses of all Tor relays
- So your local ISP can observe that you are communicating with Tor nodes
- ISPs and governments can try to block access to the Tor network by blocking Tor relays

SOLN

- Tor bridge relays are relays not listed on the public Tor directory
- Entering the Tor network through a Tor bridge relay can prevent ISPs and governments blocking access to the Tor network

Tor can also provide anonymity to websites & servers, through onion services. (For whistleblowers, etc.)

How do Onion Services work?



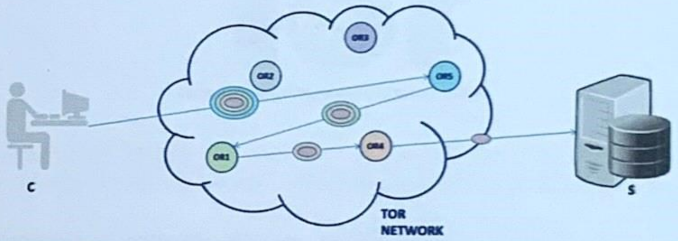
HSDir returns the intro point of the service to client

hidden.onion has a special onion URL obtained by running a TOR software on top of the web server (IP not public)

Client then connects to intro pt. of the hidden.onion service and asks it to pass a message to hidden.onion that he is waiting at this other TOR relay

So, the client does not know the IP address of hidden.onion and hidden.onion does not know who the client is (Anonymity in both ways!)

Limitations of Tor



- Tor does not provide protection against end-to-end timing attacks
- If the attacker can see both ends of the communication channel, he can correlate volume and timing information on the two sides

↳ If the entry and exit nodes are colluding, then they can do end-to-end timing attacks!

Conclusions

- Presented a brief overview of several anonymity systems
  - How they work
  - Their privacy guarantees
- Tor
  - How it works
  - Tradeoff between privacy and efficiency
- There is much more to anonymous communications
  - Tarzan, Bluemoon, *etc*