Lecture 25: Anonymity and Circumventing Internet Censorshp

COMP 332, Spring 2018

Victoria Manfredi

WESLEYAN
UNIVERSITY

Acknowledgements: materials adapted from Computer Networking: A Top Down Approach 7th edition: ©1996-2016, J.F Kurose and K.W. Ross, All Rights Reserved as well as from Avi Kak's lecture 12 slides at https://engineering.purdue.edu/kak/compsec/

Today

1. Announcements
   - hw10 written due Wed. at 11:59p

2. Internet censorship
   - overview

3. Anonymity and Circumventing Internet Censorship
   - Tor
   - decoy routing

Internet Censorship

OVERVIEW

Censorship in the real world

Sections | The Washington Post — Democracy Dies in Darkness | Sign In | Subscribe

Asia & Pacific

The walls are closing in: China finds new ways to tighten Internet controls

2017

boingboing / CORY DOCTOROW / 12:23 PM WED

2018

European Parliament ambushed by doctored version of pending internet censorship rules that sneaks filtering into all online services

-1a. Online content sharing service providers *perform an act of communication to the public and shall conclude fair and appropriate licensing agreements rightholders. If the rightholder d agree to conclude a license, the c content sharing service providers obliged to the obligations set in p*

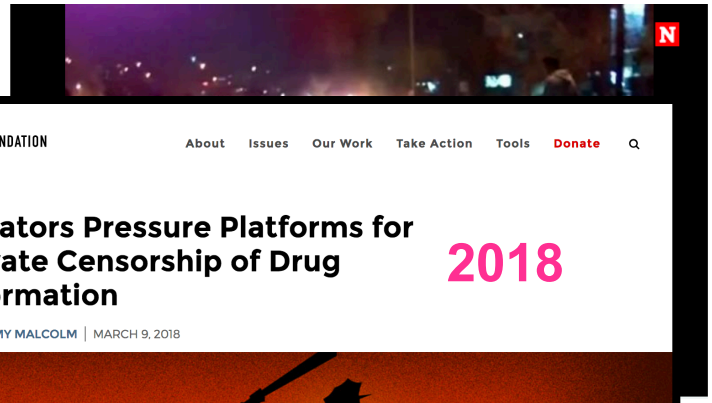
U.S. EDITION | Sun, Mar 25, 2018 | Newsweek

U.S. | World | Business | Tech & Science | Culture | Sports | Health

2018

IRAN INTERNET CENSORSHIP FORCES PROTESTERS TO TURN TO DARK WEB

BY ANTHONY CUTHBERTSON ON 1/5/18 AT 1:14 PM



ELECTRONIC FRONTIER FOUNDATION

About | Issues | Our Work | Take Action | Tools | Donate

Senators Pressure Platforms for Private Censorship of Drug Information

2018

BY JEREMY MALCOLM | MARCH 9, 2018

SECTIONS | HOME | SEARCH | The New York Times | SUBSCRIBE NOW | LOG IN

EUROPE

Erdogan's Next Target as He Restricts Turkey's Democracy: The Internet

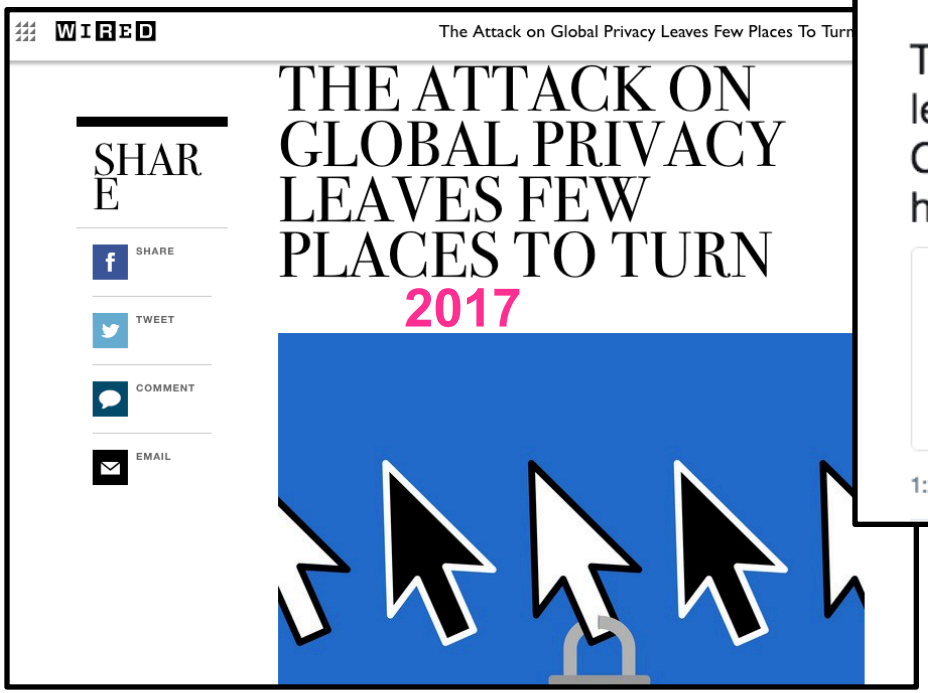
2018

By CARLOTTA GALL MARCH 4, 2018

China's state-of-the-art censorship

The Great Firewall of China

- searches in China give ``alternate'' results for certain words
- terminates connections if packets contain certain words
- … plus much more!



VPNs banned in China and Russia

Why aren't TLS/SSL and IPsec enough?

Packet headers are plaintext

- src and dst IP addresses visible to everyone

Traffic analysis attacks

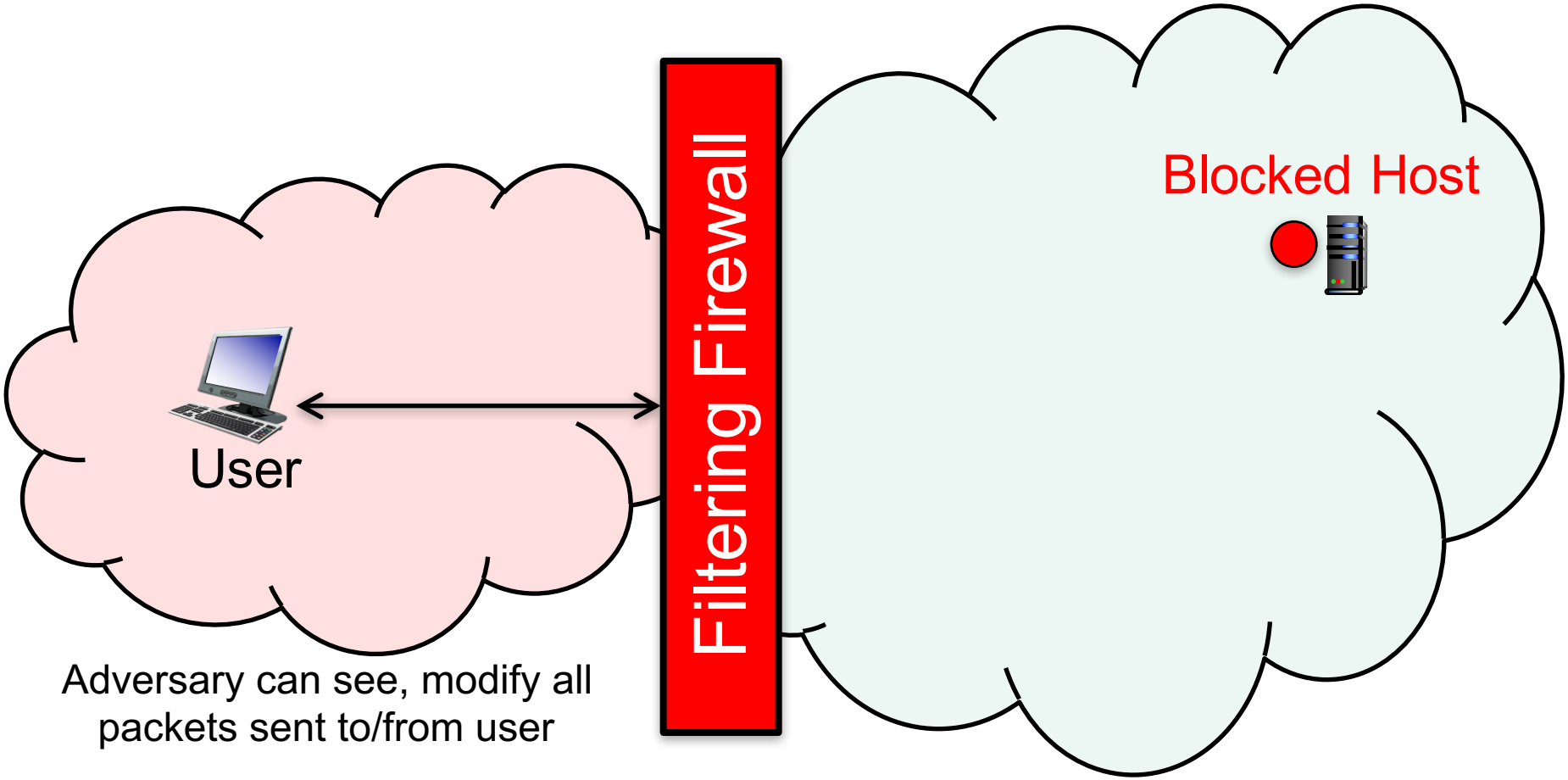
- obtain info about original src of packets and their ultimate dst

Even IPsec is vulnerable

- packet sniffer at any point before packets get to encapsulator used for Tunnel Mode knows both pkt src and dst

Many sites easily blocked or filtered

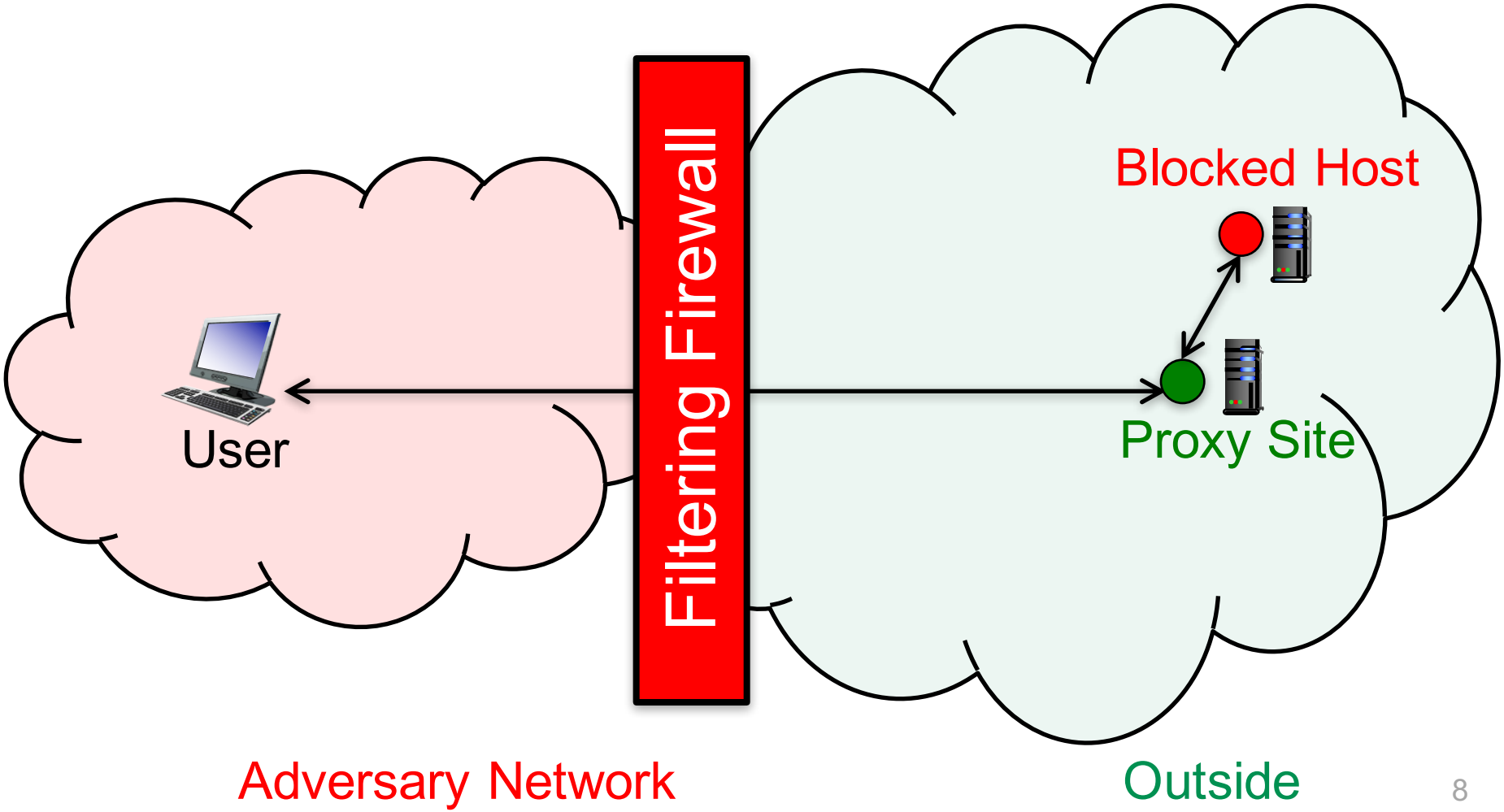
How? Don't forward packets with certain dst IP addresses



Current solutions

Some users use proxies or VPNs to bypass filters. How?

Connect first to unblocked site that accesses blocked site on your behalf. So your packets never have blocked destination IP address on them



Anonymity and Circumventing Internet Censorship

TOR

Acknowledgements

Most of this section based on Avi Kak's excellent slides

- lecture 20 slides at https://engineering.purdue.edu/kak/compsec

Also based on

- https://www.torproject.org/about/overview

The Onion Router (Tor)

Goals

- enable user to access blocked sites
- prevent adversary from sniffing traffic to analyze pkt headers to find out who is talking to whom

Uses onion routing idea to provide anonymized routing

- by Roger Dingledine, Nick Mathewson, Paul Syverson

Onion routing

- incrementally build path through Tor overlay network
  - each relay node in path knows only its predecessor and successor nodes
- layers of encryption placed on Tor messages
  - user negotiates separate set of encryption keys for each hop along path

Onion Proxies (OPs) and Onion Routers (ORs)

1. User's OP queries Tor directory
   - for IP addresses of ORs in Tor overlay
2. User selects 3 ORs
   - to construct path to dst
   - e.g., {B, C, D}

Q: How are packets forwarded over Internet to/between ORs?

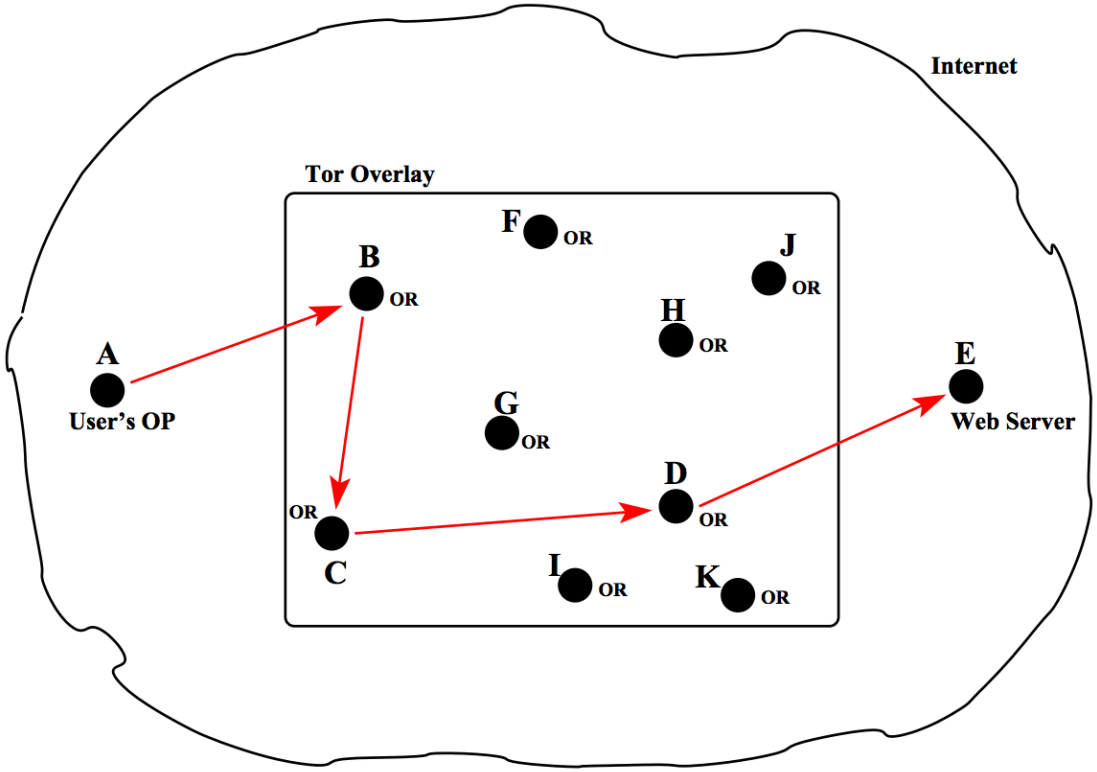


Figure 14: *B, C, and D are the ORs selected by user A for a path to the destination E.* (*This figure is from Lecture 20 of "Computer and Network Security" by Avi Kak*)

How user's OP builds path through Tor overlay

Control and relay torpackets used to create circuit

- data carried by relay torpackets during circuit construction
  - control torpackets at current terminal node on path to extend path

Every OR node has

- (static) public RSA key that it makes available to user's OP
- Diffie-Hellman (DH) key Y
  - created between user's OP and each OR on path chosen by user

When user's OP wants to send msg to OR on path,

- msg encrypted with session key
  - derived from OP's DH Y key and OR's DH Y key

How user's OP builds path through Tor overlay

A creates control torpacket sent from A to B

DATA field contains A's DH Y key $Y_{A \rightarrow B}$ encrypted with B's RSA public key

B responds back to A with control torpacket

DATA field contains B's DH Y key $Y_{B \rightarrow A}$. Now both A and B can calculate secret session key K_{AB} for their link

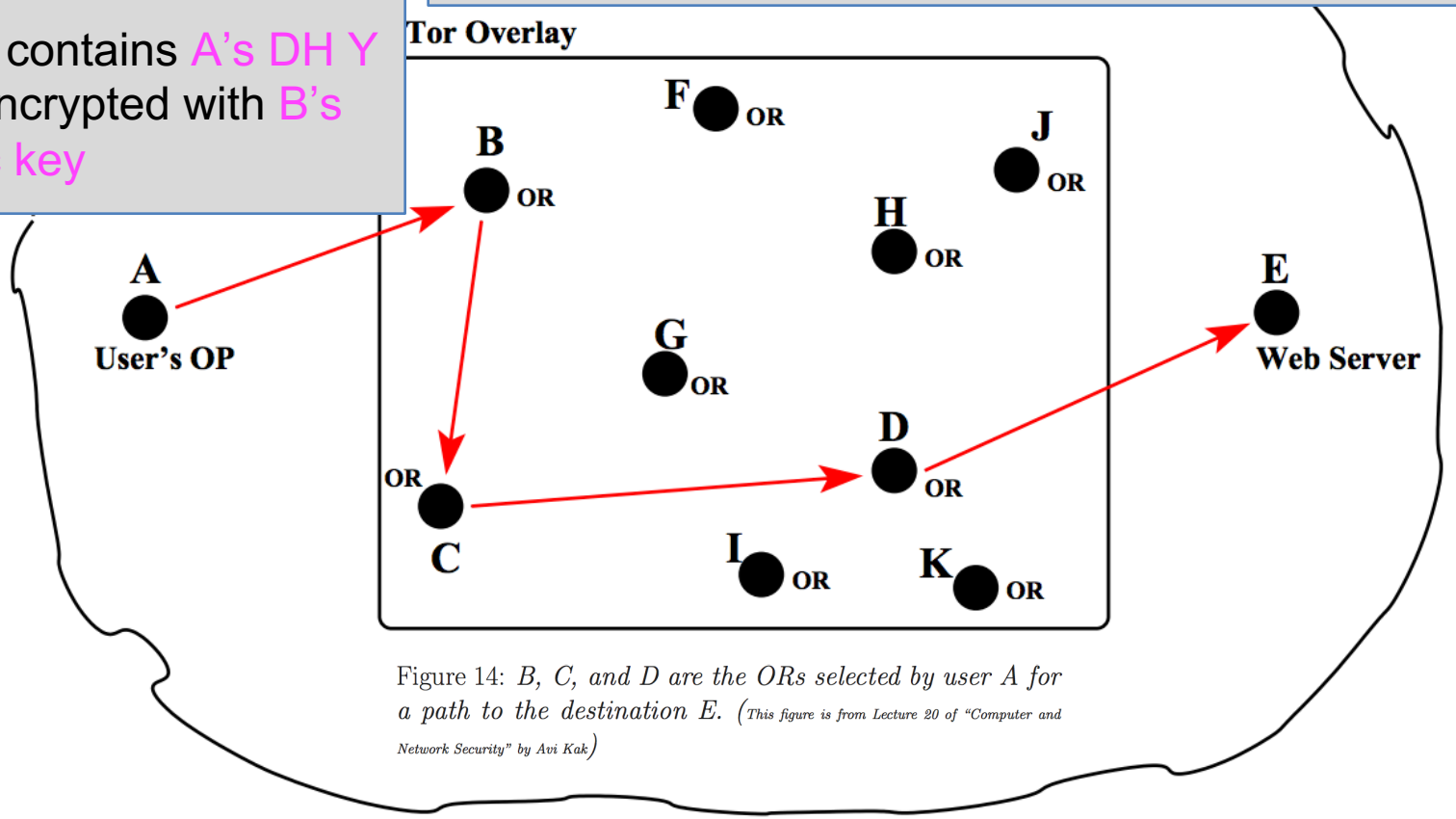


Figure 14: *B, C, and D are the ORs selected by user A for a path to the destination E.* (*This figure is from Lecture 20 of "Computer and Network Security" by Avi Kak*)

All communications between any pair of nodes in underlying network takes place using the TSL/SSL protocol for confidentiality

How user's OP builds path through Tor overlay

A sends B relay extend torpacket to extend circuit

DATA field includes DH Y key $Y_{A \rightarrow C}$ for new terminal node on path, C, and identity of new node. $Y_{A \rightarrow C}$ is encrypted with C's RSA public key to prevent B from seeing. DATA is then encrypted with the session key K_{AB}

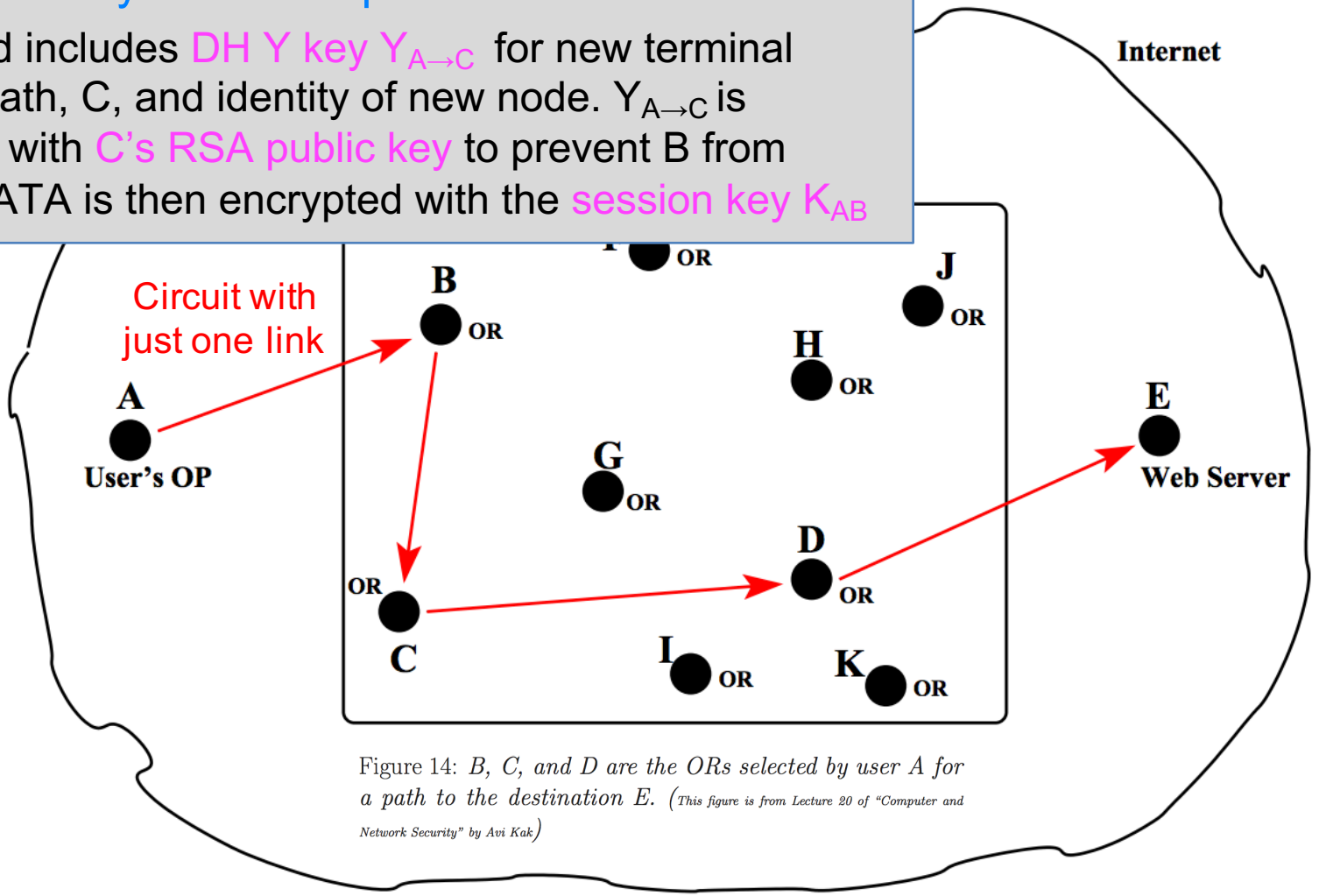


Figure 14: *B, C, and D are the ORs selected by user A for a path to the destination E.* (*This figure is from Lecture 20 of "Computer and Network Security" by Avi Kak*)

Nodes A and B can now start exchanging relay torpackets

How user's OP builds path through Tor overlay

B generates control torpacket

DATA field contains payload of A's relay extend

Node C responds to B with create control torpacket

DATA field contains C's DH Y key $Y_{C \rightarrow A}$. Node B sends this to A using relay extended torpacket. Now both A and C can calculate secret session key K_{AC} for any messages A may want to send to C that B should not see

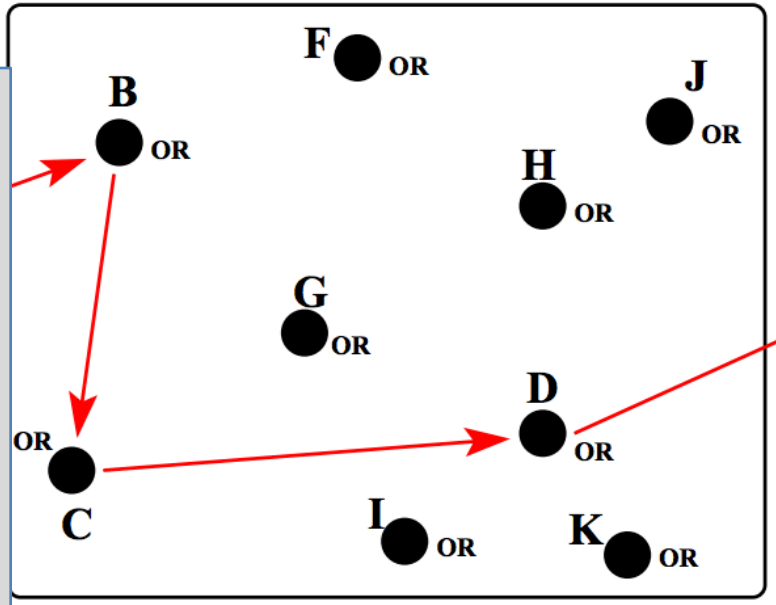


Figure 14: *B, C, and D are the ORs selected by user A for a path to the destination E.* (*This figure is from Lecture 20 of "Computer and Network Security" by Avi Kak*)

Each node on path has only local knowledge about path

Can D see source IP addr of A?

"One Bad Apple Spoils the Bunch" by Stevens Le Blond, Pere Manils, Abdelberi Chaabane, Mohamed Ali Kaafar, Claude Castelluccia, Arnaud Legout, and Walid Dabbous. Takes advantage of peculiarities of BitTorrent protocol to reveal src IP addr of 10,000 hosts using Tor for BitTorrent downloads during 23 day period in 2011

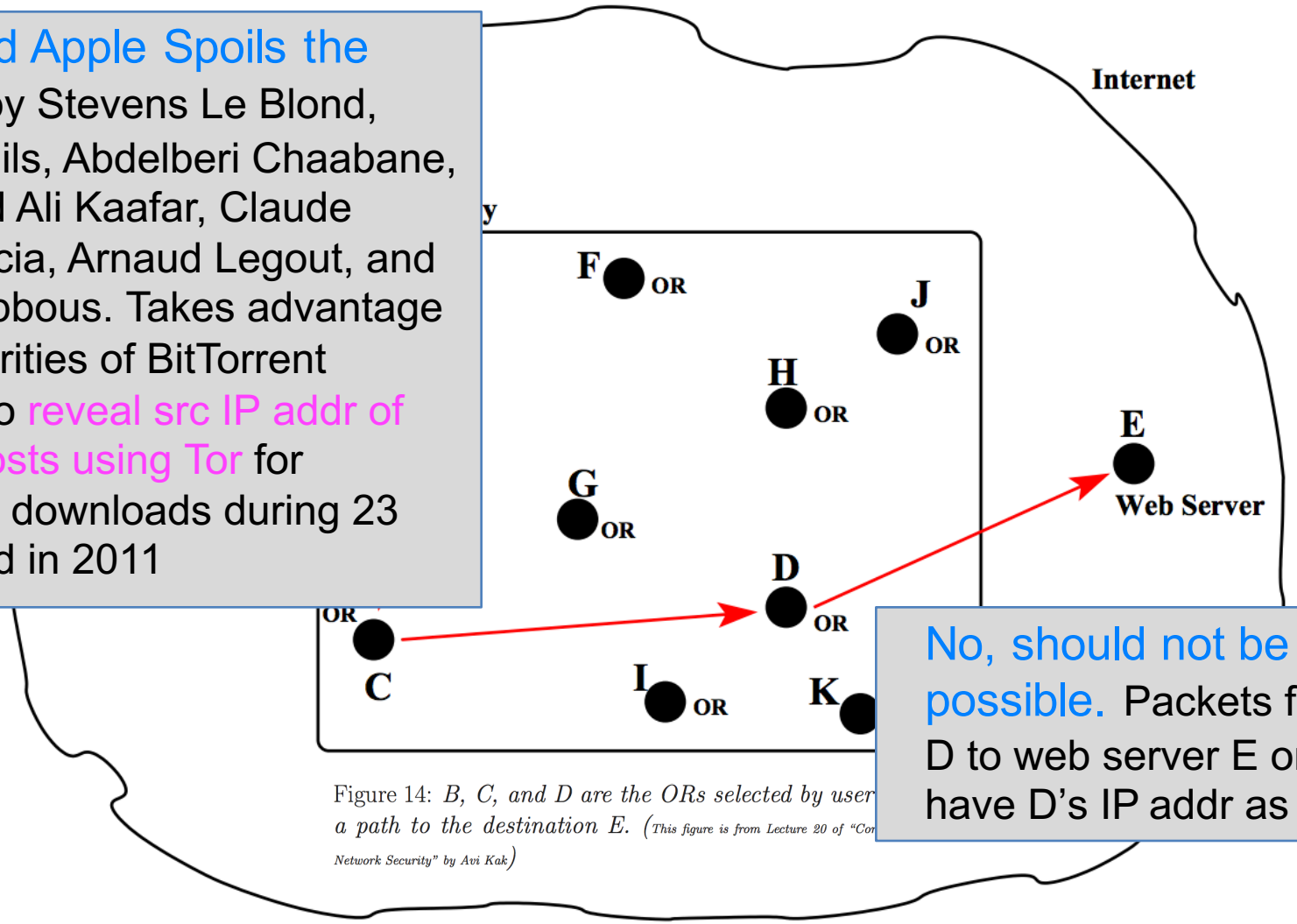


Figure 14: *B, C, and D are the ORs selected by user a path to the destination E.* (*This figure is from Lecture 20 of "Co Network Security" by Avi Kak*)

No, should not be possible. Packets from D to web server E only have D's IP addr as src

Can D see data of sent to/from A/E?

Not if node A is trying to reach HTTPS site

End-to-end encryption of pkt payload

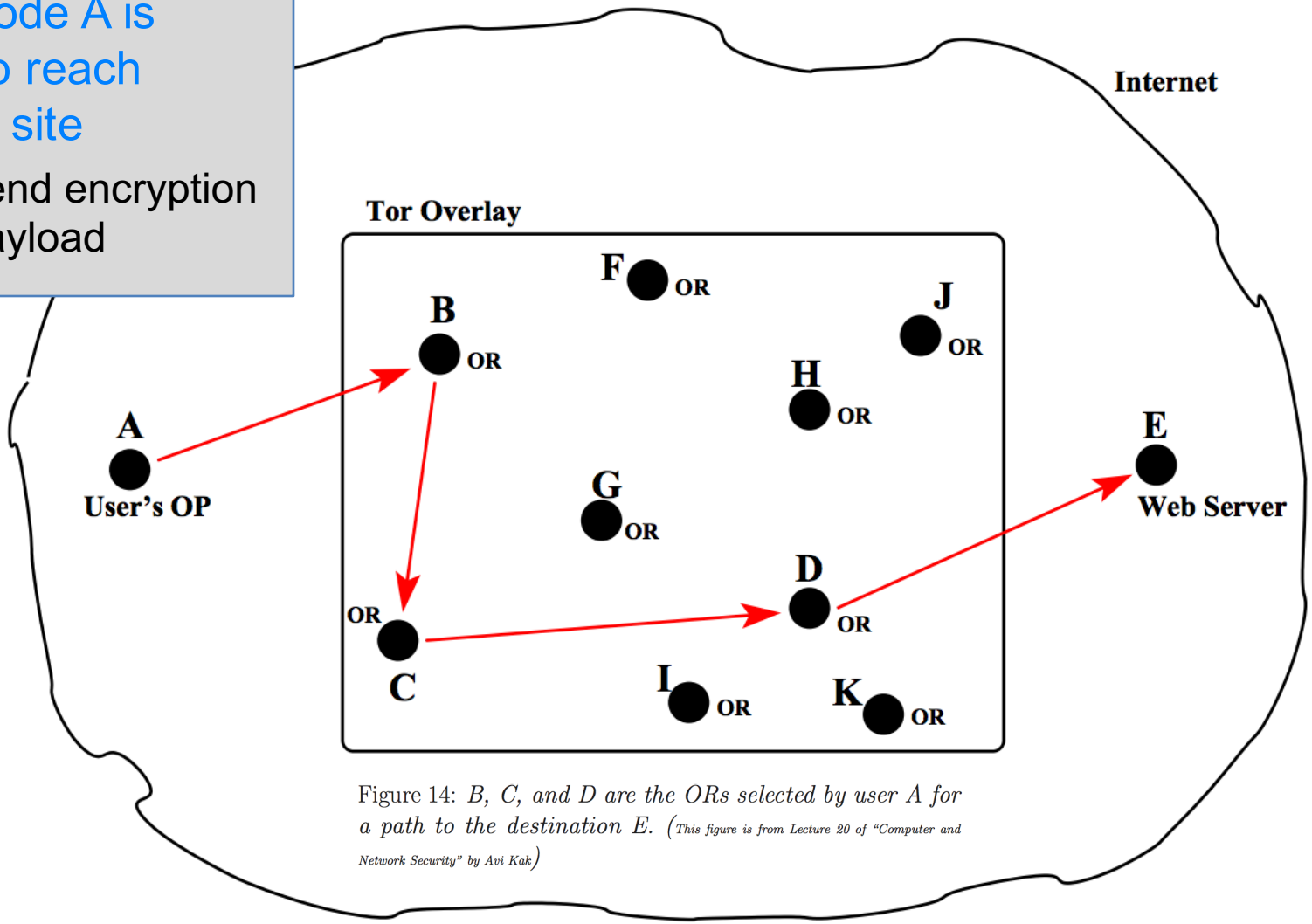
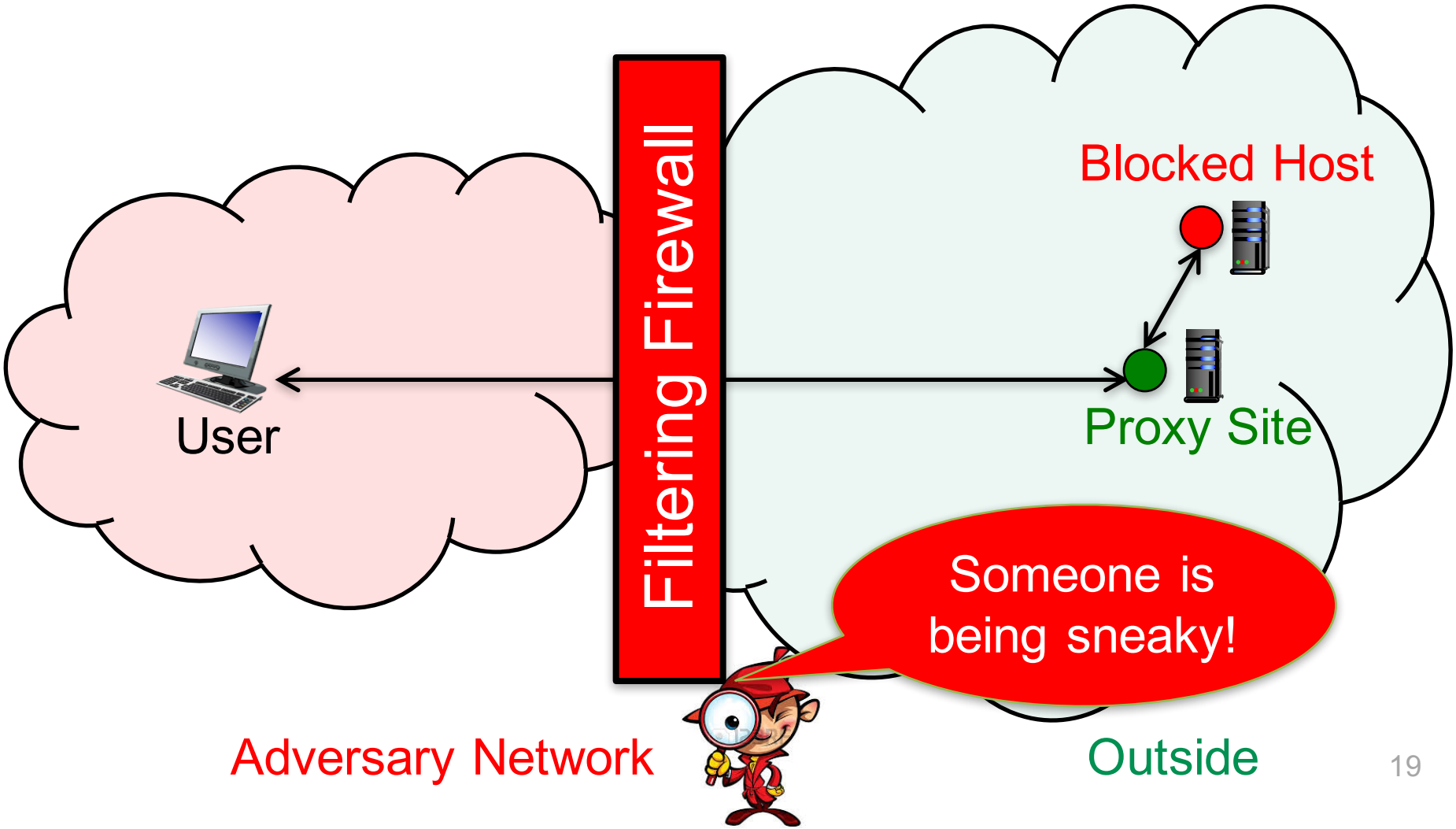


Figure 14: *B, C, and D are the ORs selected by user A for a path to the destination E.* (*This figure is from Lecture 20 of "Computer and Network Security" by Avi Kak*)

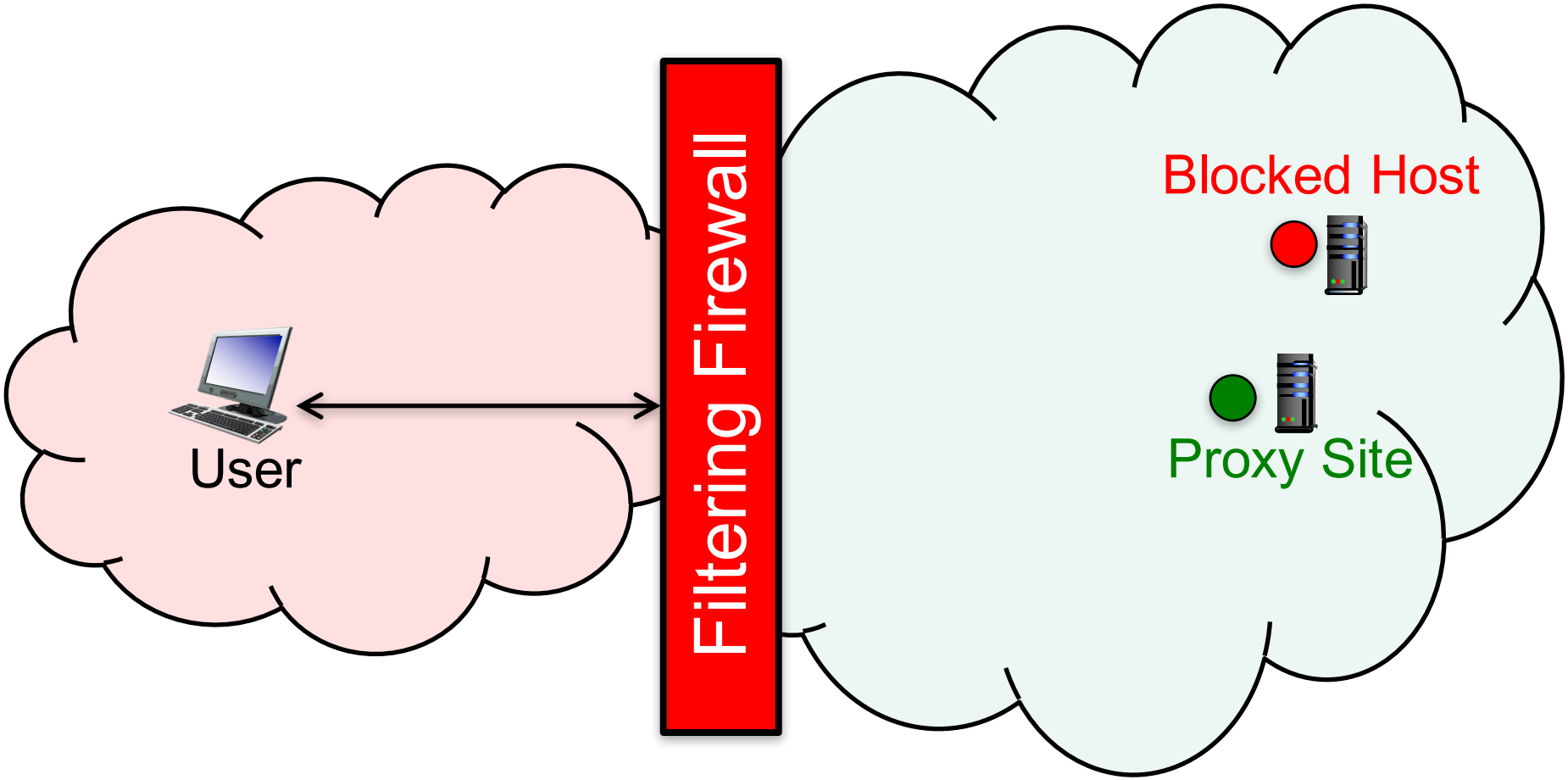
What information does adversary see?

That user is connecting to Tor OR. Just using a proxy can attract unwanted attention … I may not know packet's true destination, but by wanting to hide destination, I deem you suspicious



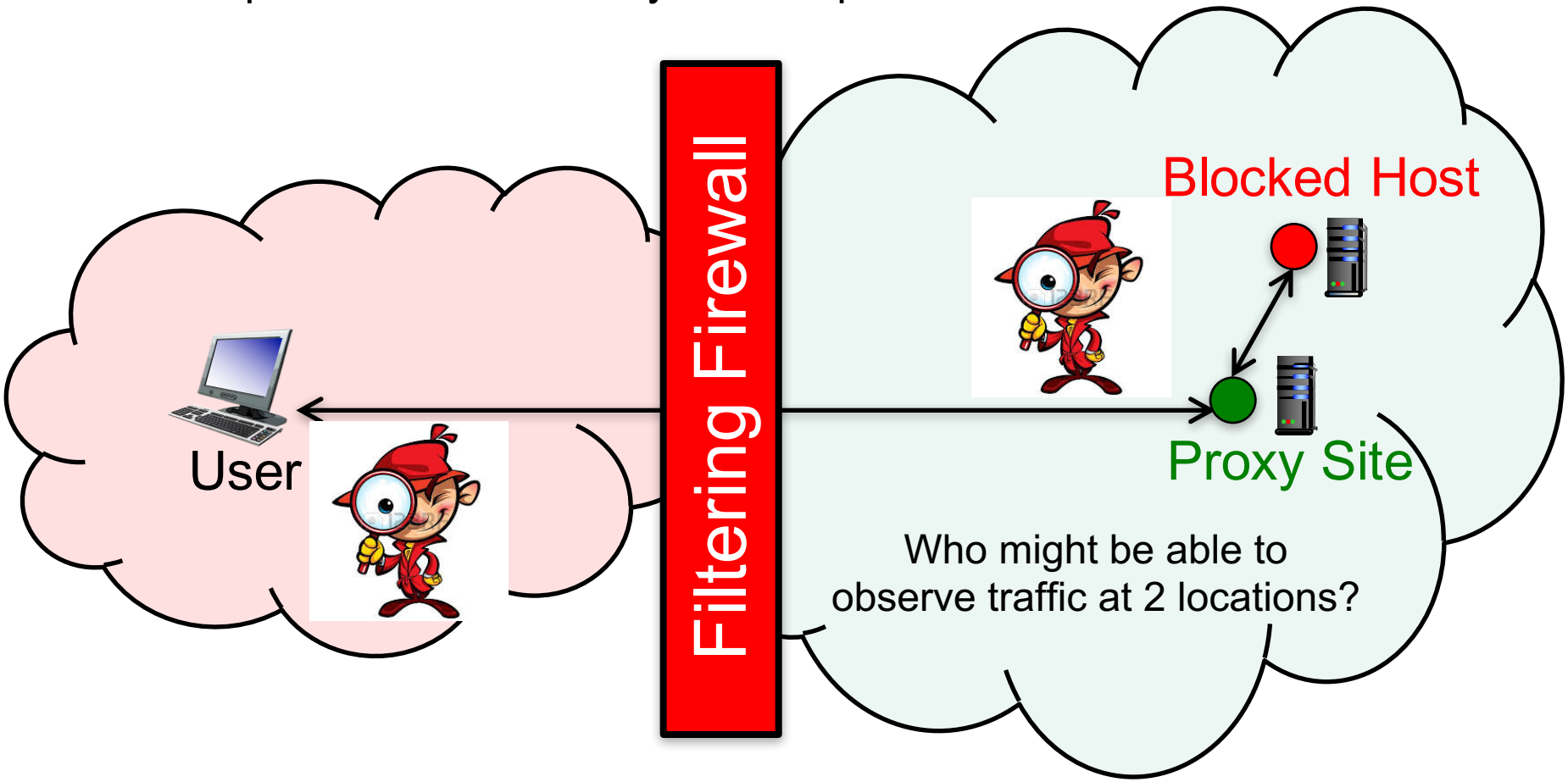
What can adversary do?

IP addresses are easily blocked. Adversary drops pkts with dest IP addresses associated with Tor ORs or proxies



Tor assumes adversary observes at 1 location

If adversary observes at 2 locations, can break Tor. Why? timing attacks, because onion routers don't reorder packets, adversary can correlate input traffic at Tor entry with output traffic at Tor exit

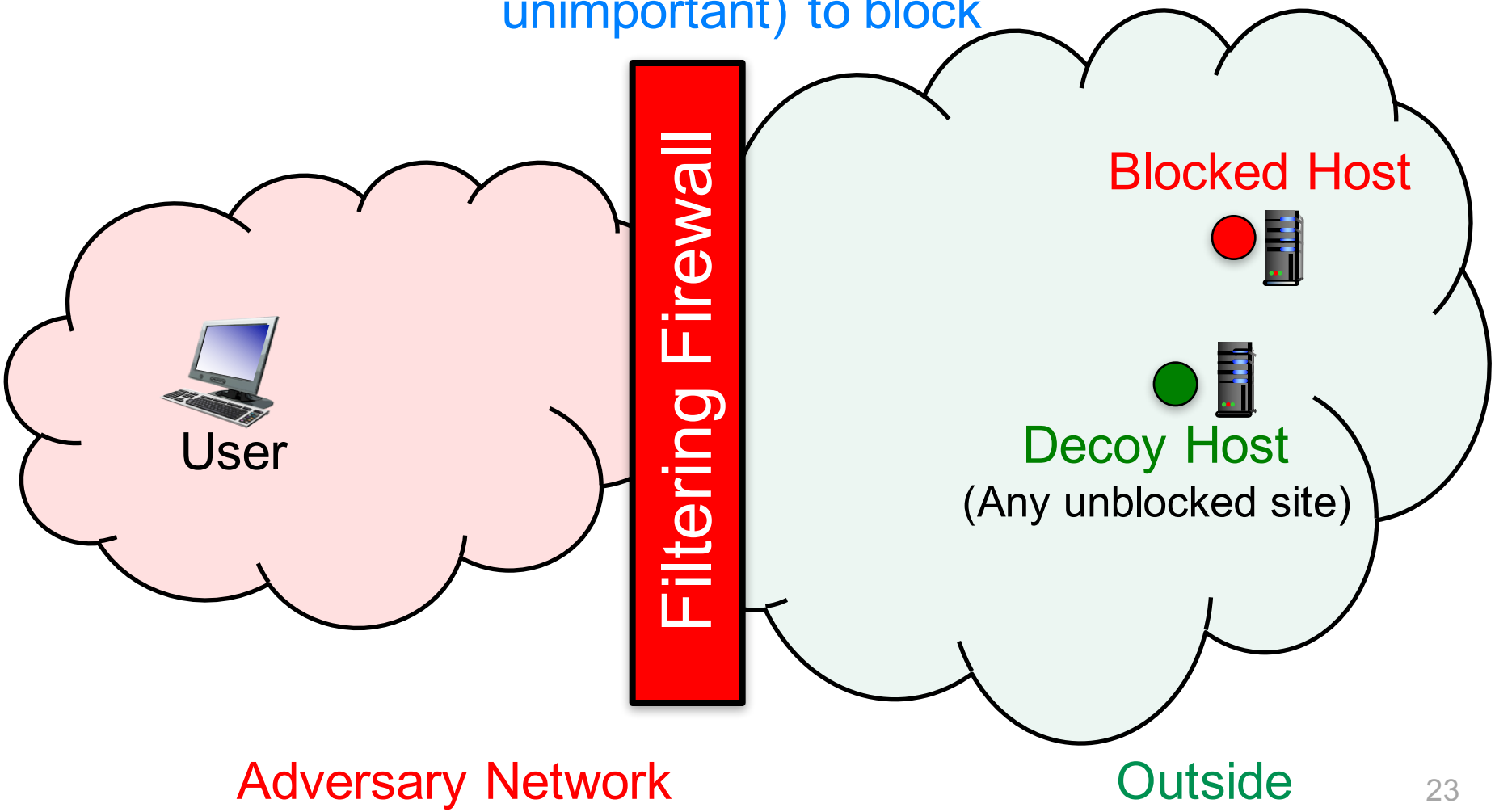


Anonymity and Circumventing Internet Censorship

DECOY ROUTING

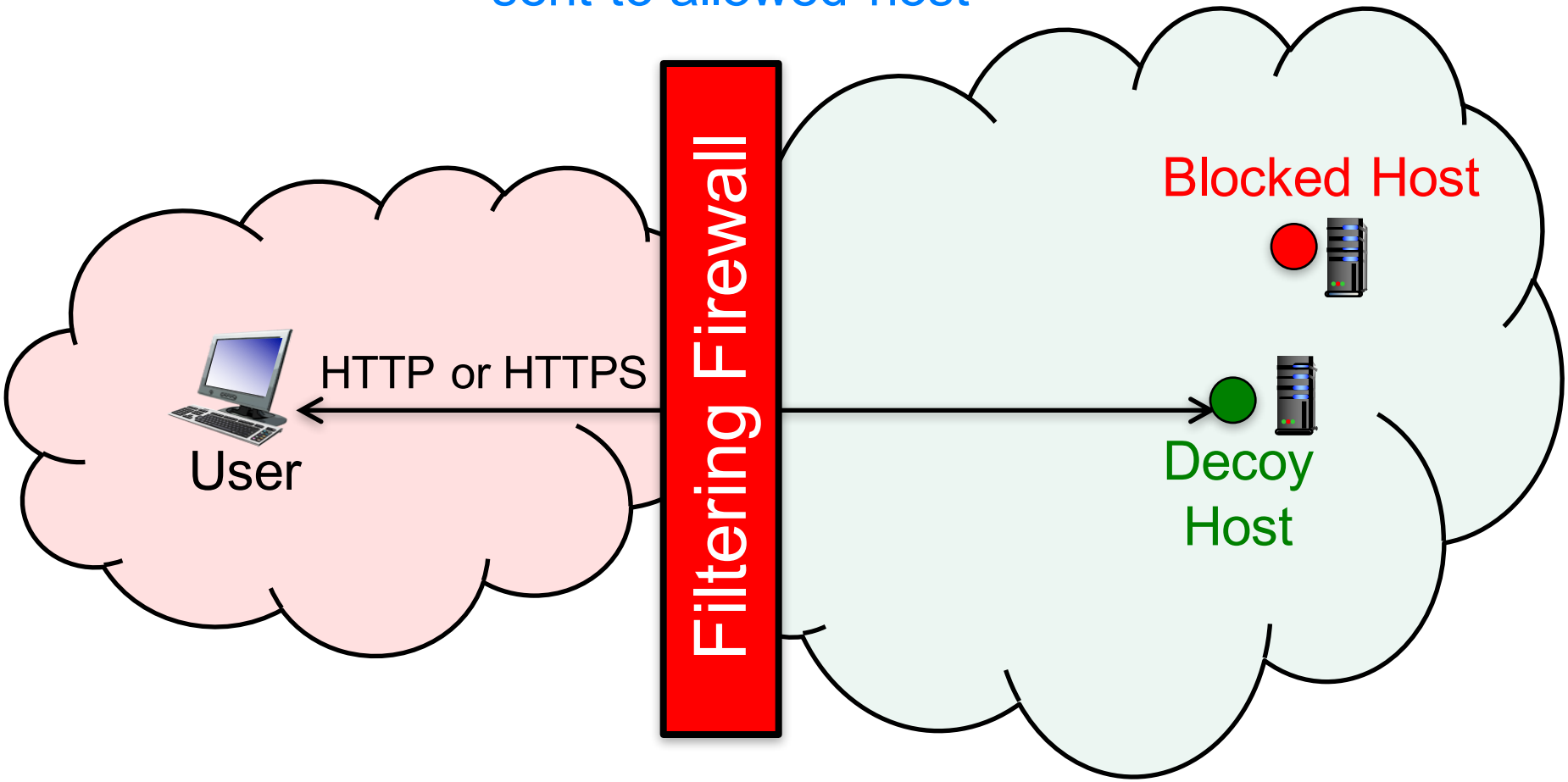
Decoy Routing

Takes advantage of (1) routers being much harder to block than hosts and (2) the existence of sites that are too important (or unimportant) to block



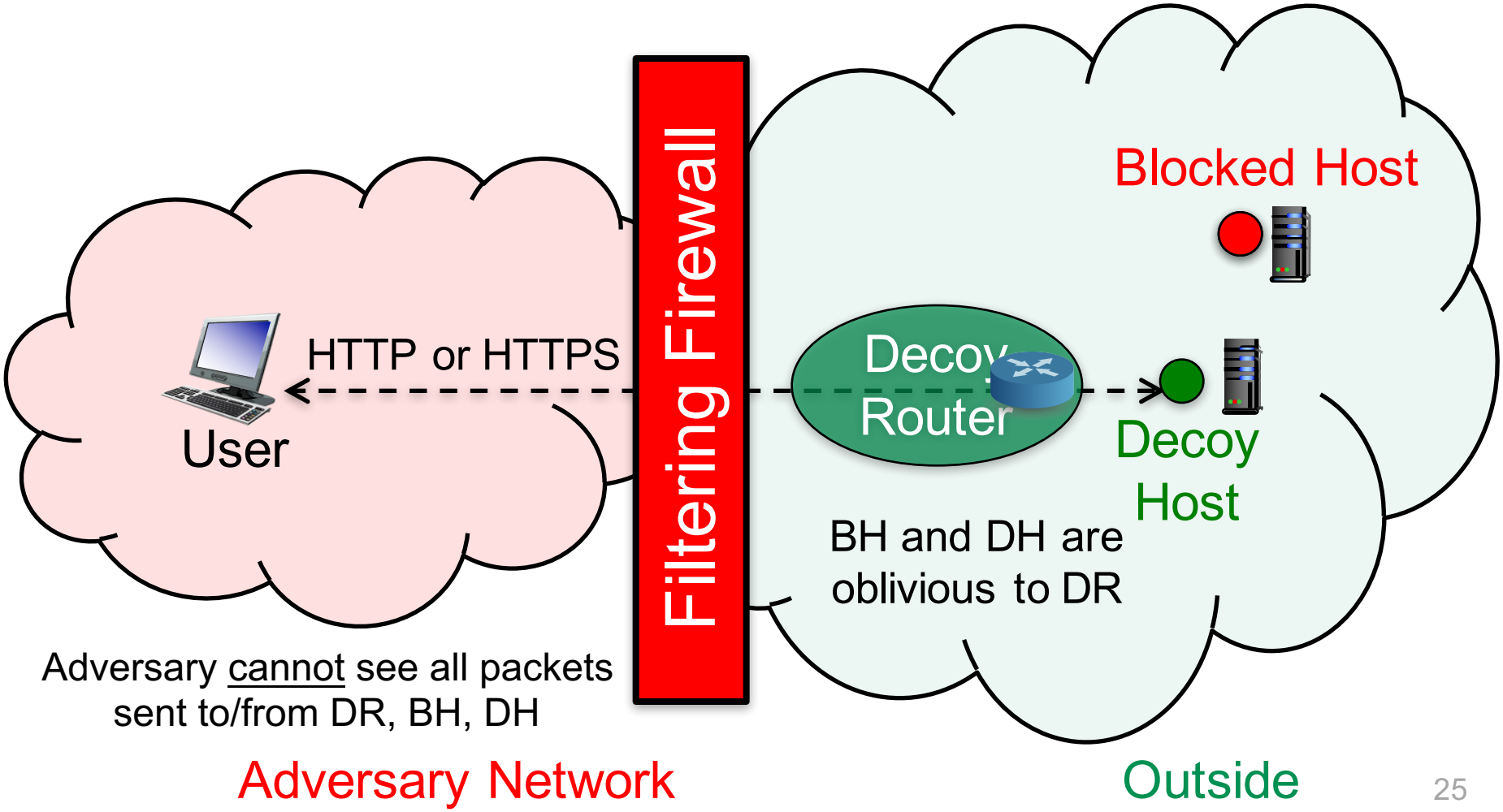
Decoy Routing

A cryptographic signal (a string of random numbers and letters), generated from User's secret key, is hidden in packets sent to allowed host



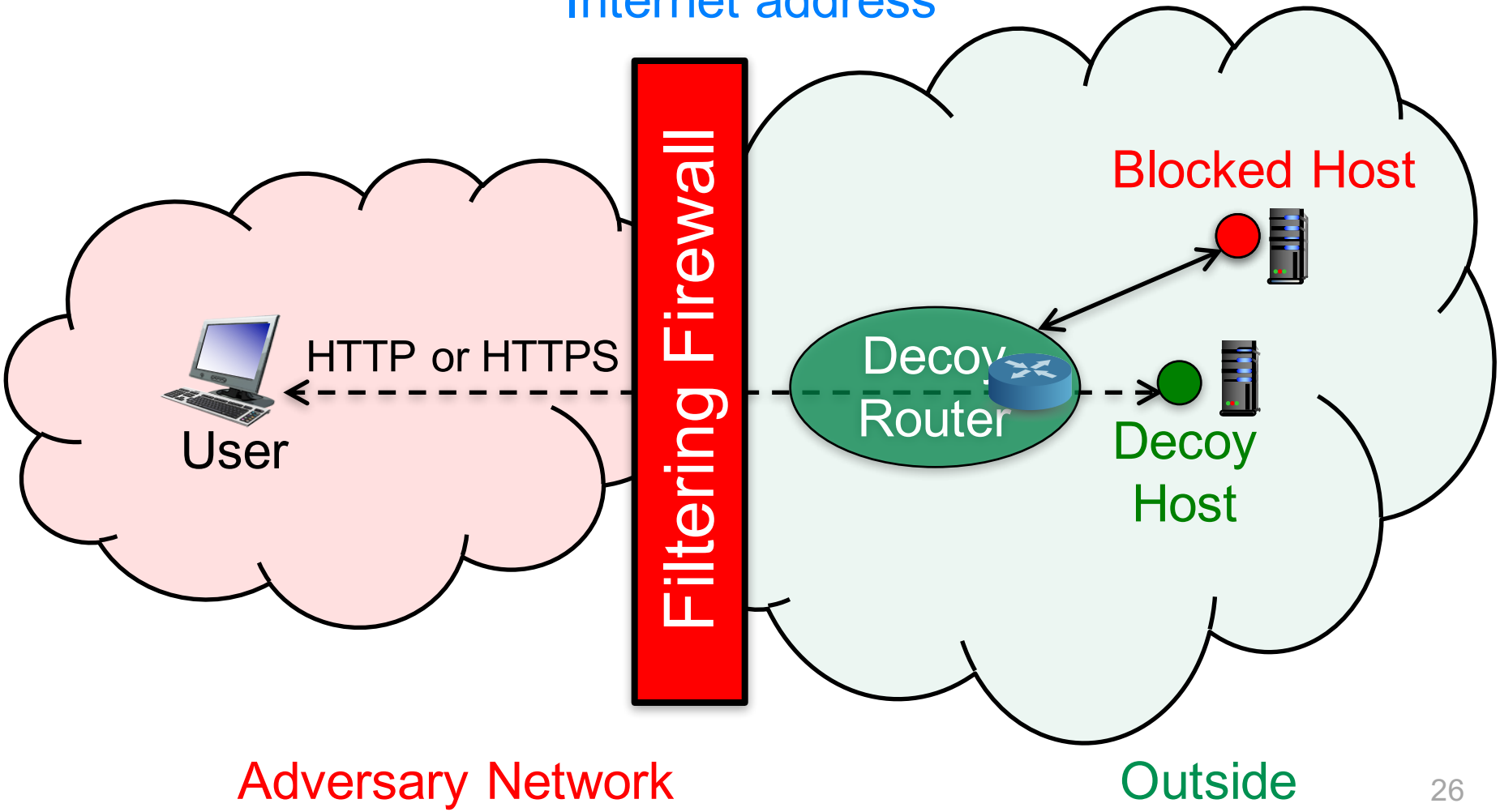
Decoy Routing

If a Decoy Router detects the hidden signal, it responds to User with its own hidden signal



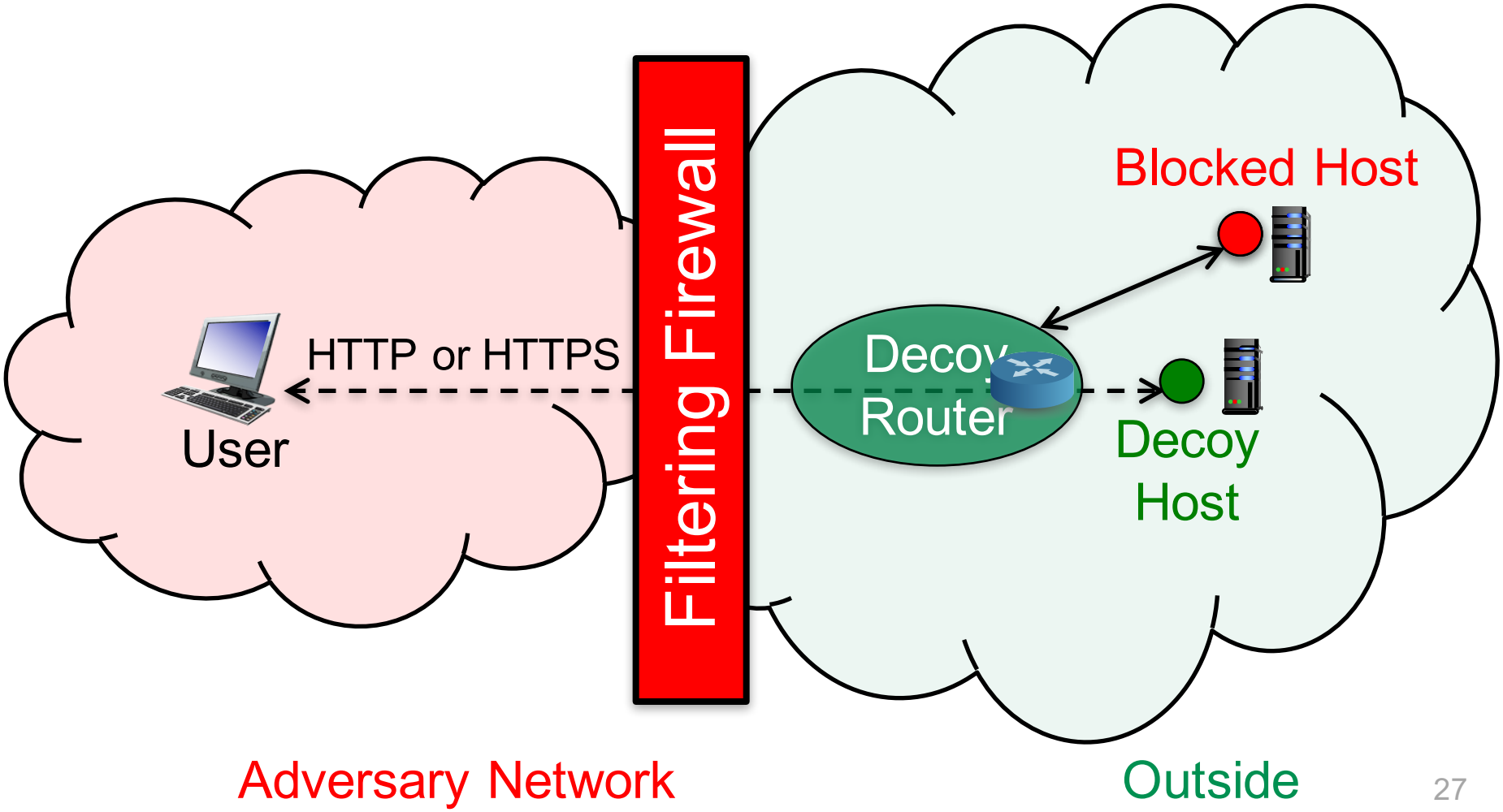
Decoy Routing

The Decoy Router and User establish a secure connection. User can now securely connect via Decoy Router to any Internet address



What does Decoy Router know about User?

Everything! Decoy Router is impersonating user to DH. User is only anonymous to adversary, not to Decoy Router



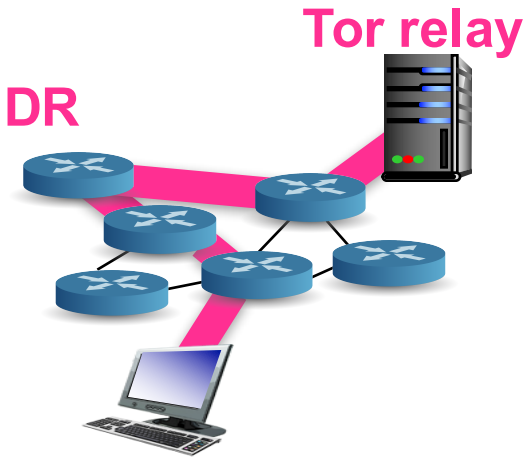
Providing user anonymity

Tor

- provides anonymity
- (also circumvents Internet censorship but Tor relays are easily blocked once identified)

Decoy Routing

- circumvents Internet censorship
- (Decoy) Routers are not easily blocked
  - sources choose dst, not route of pkts



Decoy Routing + Tor

- use Decoy Routing to reach Tor entry node
  - i.e., Tor node is blocked host that user wants to access

Implementing Decoy Routing

1. Handshake
   - User and Decoy Router authenticate each other and connection
   - has been embedded within HTTP, HTTPS, TCP protocols
   - both public-key and private-key crypto systems have been used

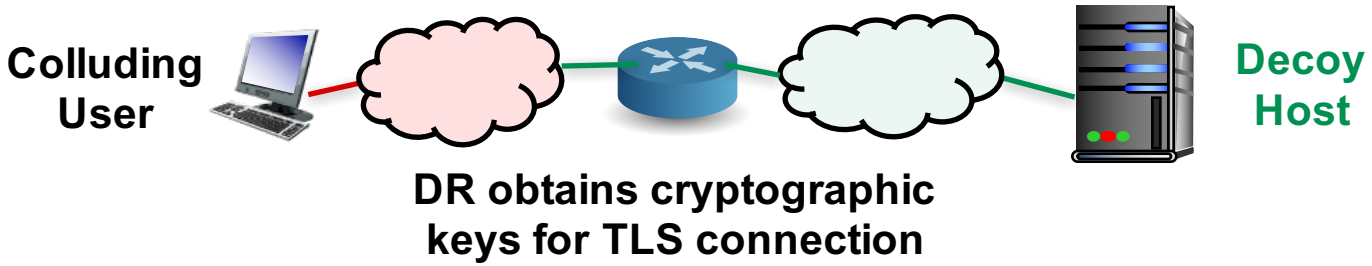
2. Tunnel
   - User → DR communication
     - tunneled through User → Decoy Host packets
   - DR → User communication
     - tunneled through Decoy Host → User packets

Different Decoy Routing implementations take different approaches to how handshake and tunnel are implemented

Rebound protocol

Rebound: Decoy Routing on Asymmetric Routes Via Error Messages. D. Ellard, C. Jones, V. Manfredi, T. Strayer, B. Thapa, M. Van Welie, A. Jackson. In LCN 2015.

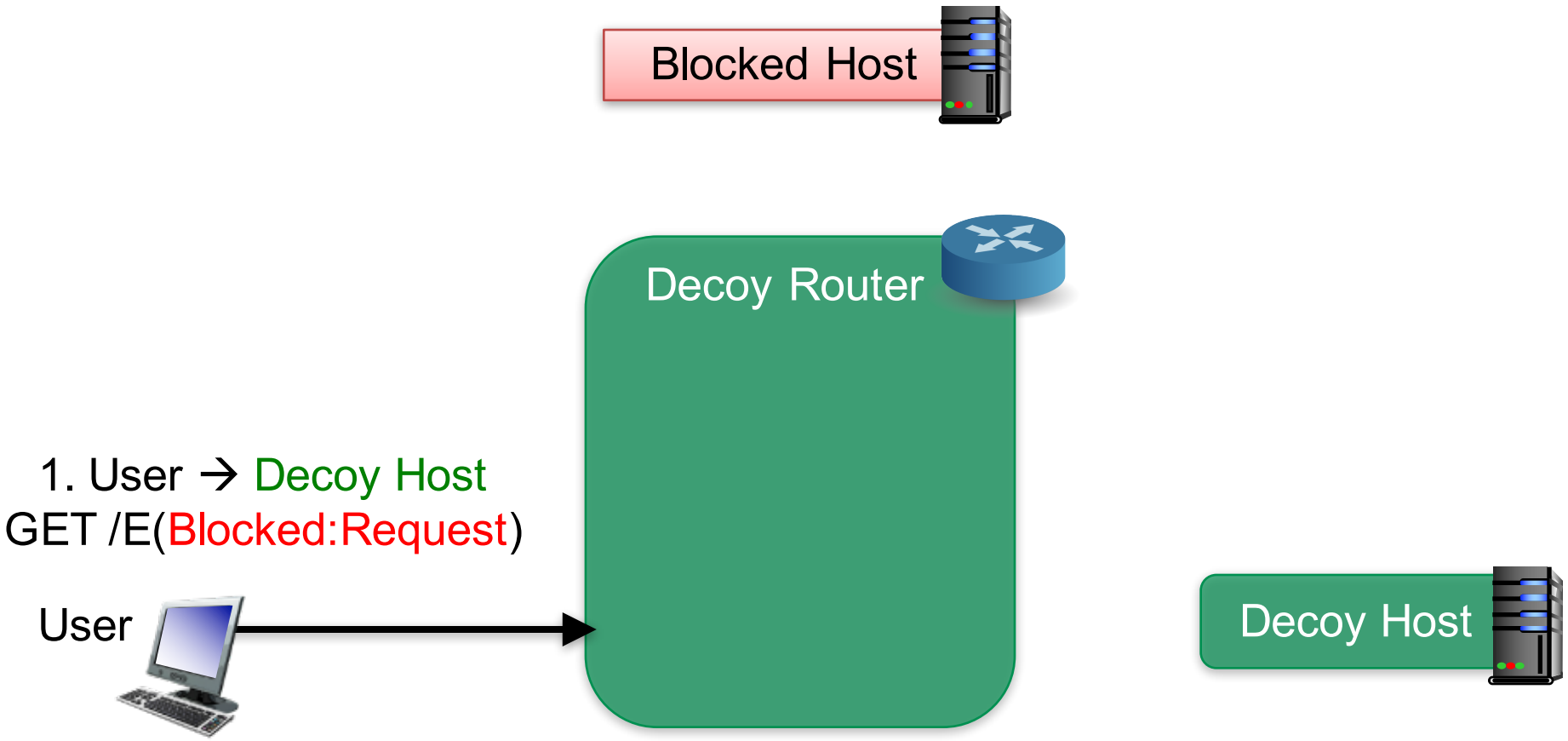
1. Handshake
   - Decoy Router man-in-the-middles TLS session for colluding User on possibly asymmetric route



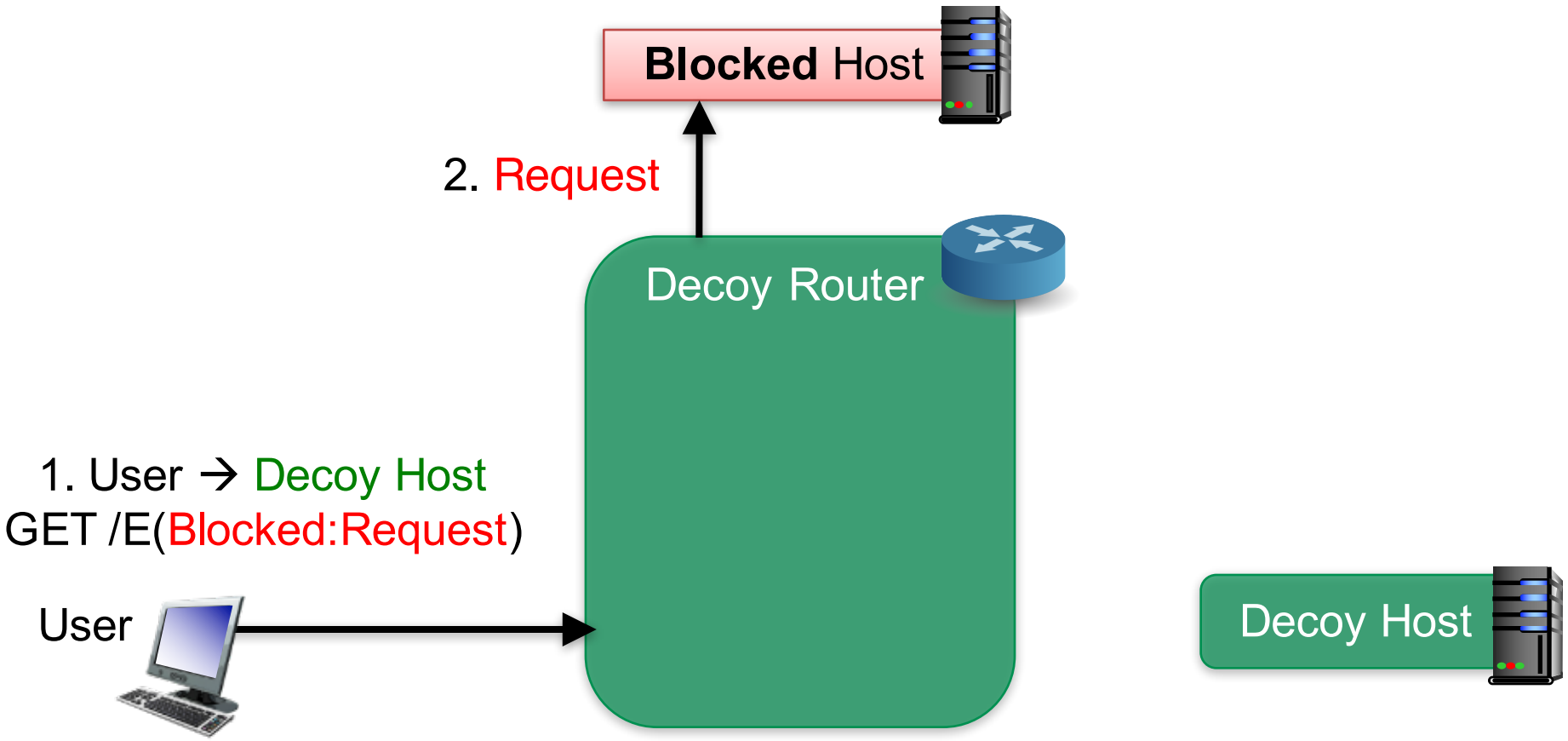
2. Tunnel
   - takes advantage of how HTTP error messages work
     - user requests unknown URL, error response contains URL

DR rewrites only User packets, does not impersonate server

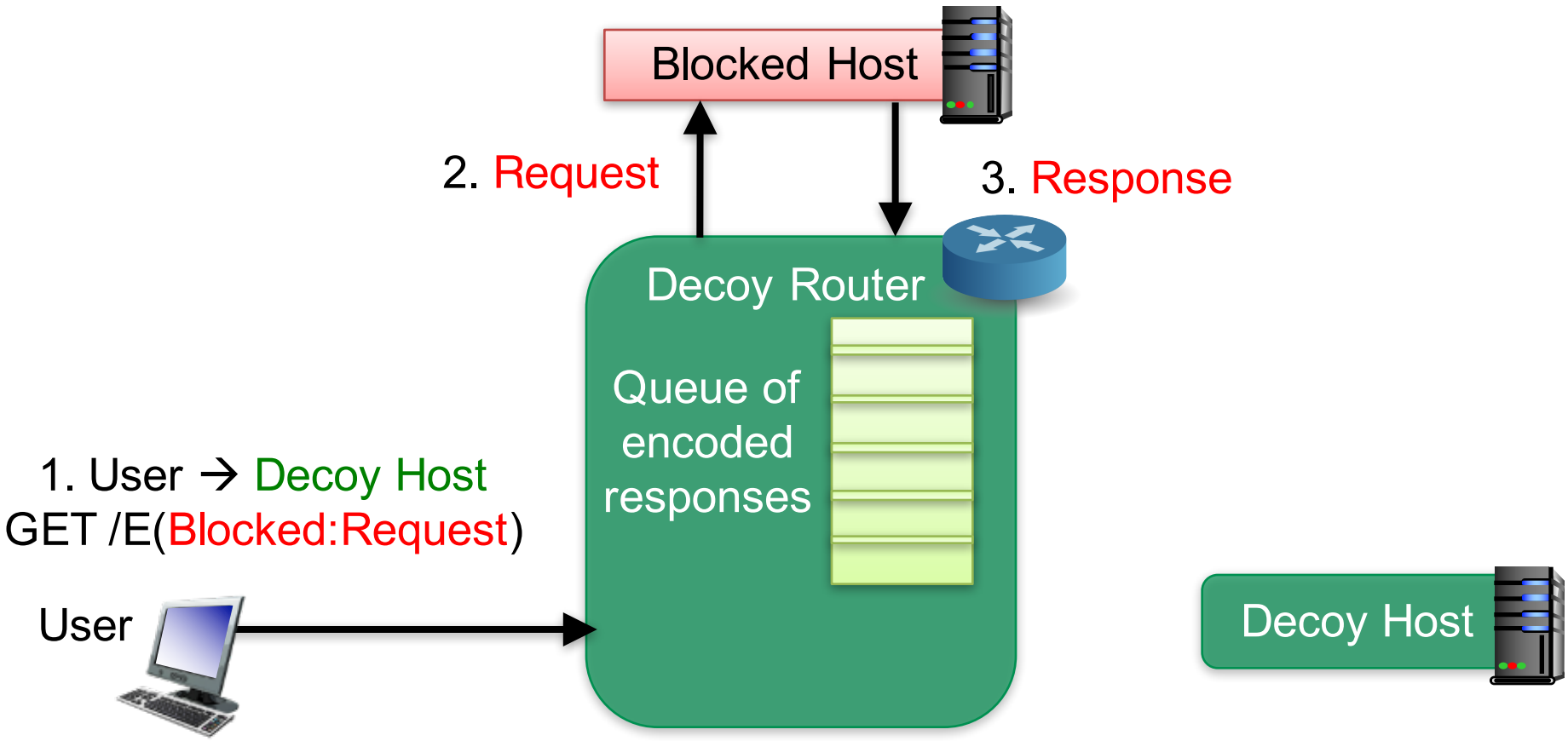
Rebound Tunnel



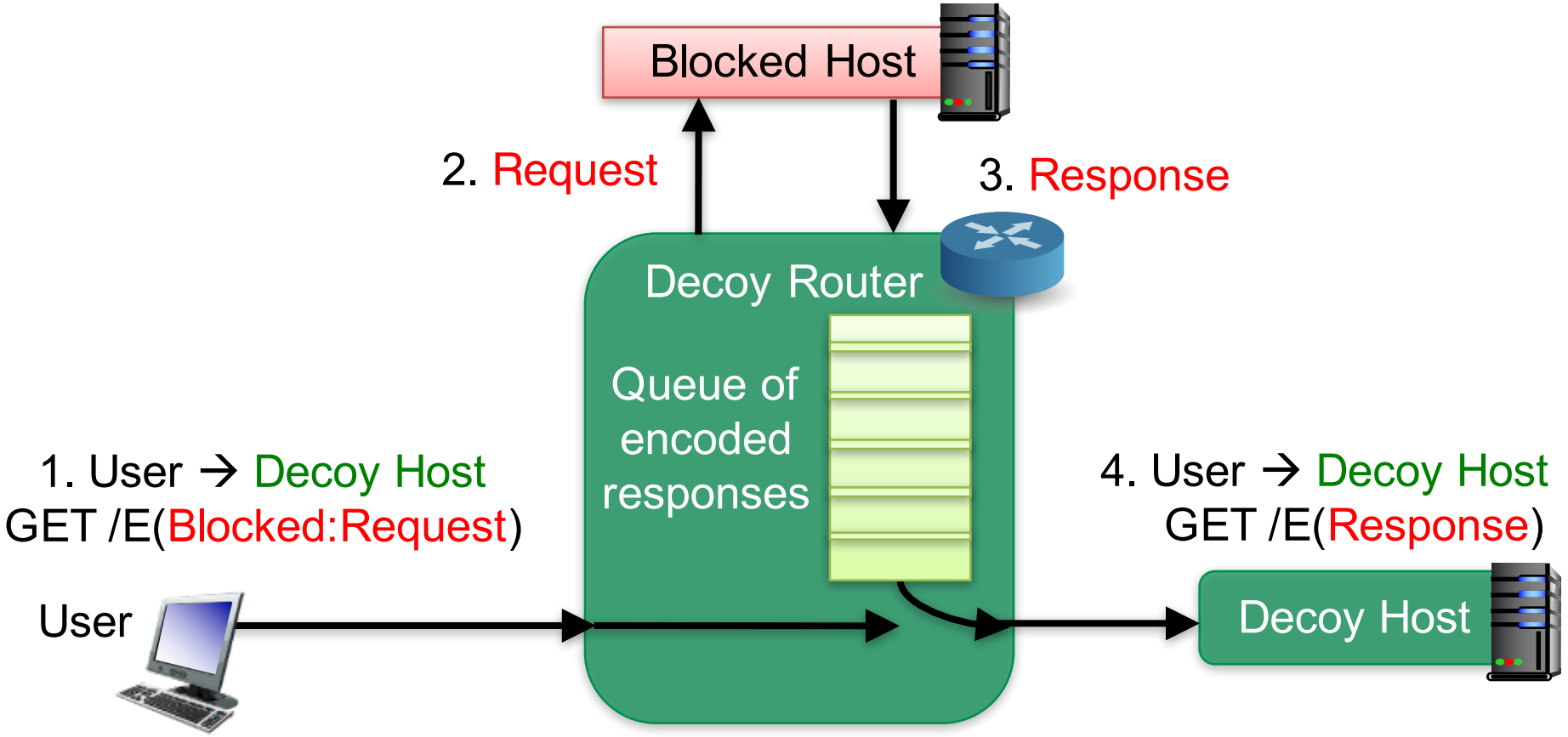
Rebound Tunnel



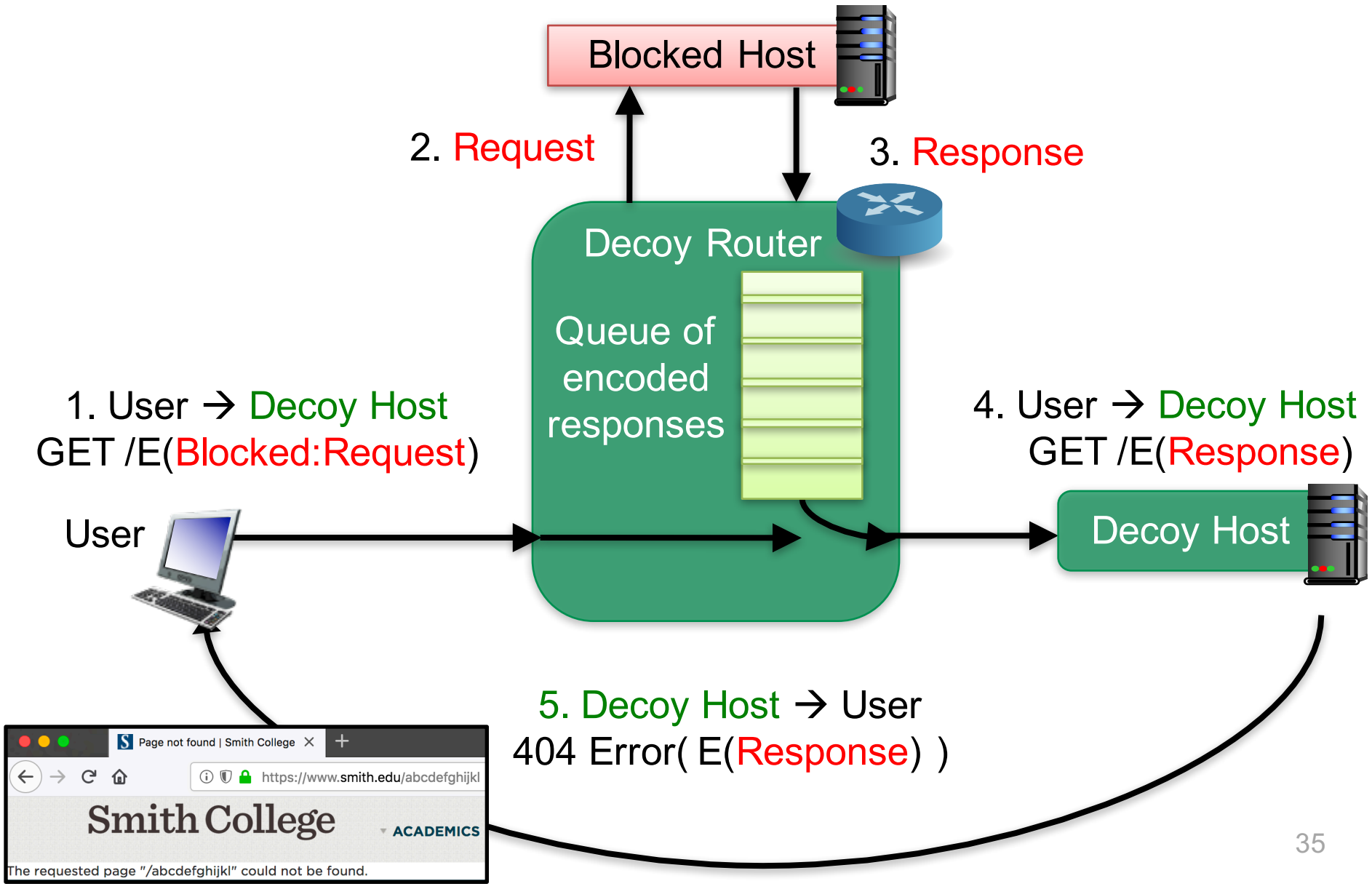
Rebound Tunnel



Rebound Tunnel



Rebound Tunnel



Rebound performance over Internet

User app on laptop

– connected via wifi to Decoy Router that is 12 hops away

Protocol	Bytes/s	Std Dev
HTTP	1,174,240	83,812
Rebound	129,398	9,655

Transfer rates for 1 MB transfers from Blocked Host to User

Blocked Site	Rebound Load Time	Ordinary Load Time
cnn.com	38.7 s (4.24 s stdev)	7.68 s (6.28 s stdev)
nytimes.com	17.5 s (8.55 s)	3.31 s (0.85 s)
en.wikipedia.org	2.1 s (0.89 s)	0.38 s (0.05 s)
slashdot.org	23.9 s (6.04 s)	4.32 s (0.66 s)
twitter search	9.44 s (1.39 s)	0.91 s (0.09 s)
google search	4.96 s (1.30 s)	0.27 s (0.09 s)

Our code is open source: https://curveball.nct.bbn.com