

# PRACTICAL NETWORK SECURITY

**Kunwadee Sripanidkulchai, Ph.D.**

kunwadee (AT) cp.eng.chula.ac.th

# Disclaimer:

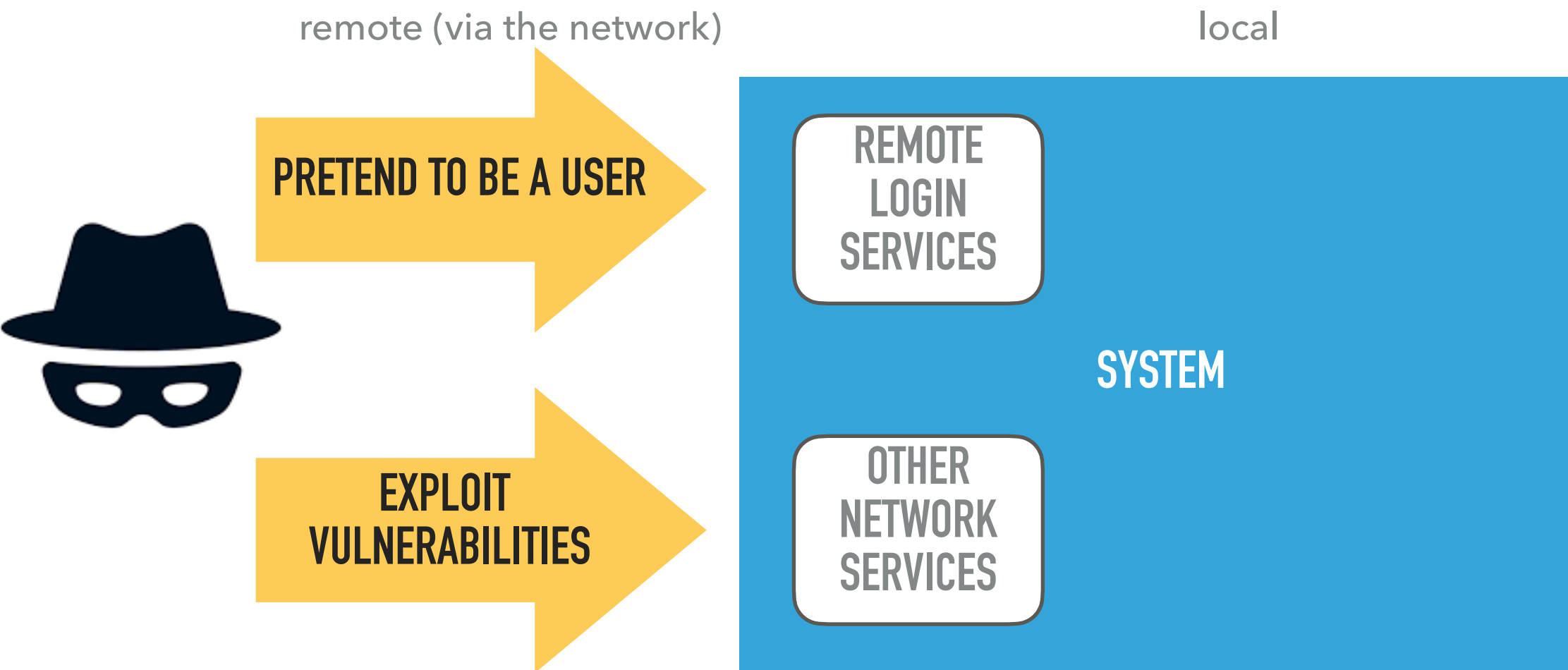
The information in these slides and the accompanying activity is for educational purposes so that you can harden your systems and make ethical choices. Use responsibly.

"Attack is the secret of defense; defense is the planning of an attack."

Sun Tzu, The Art of War

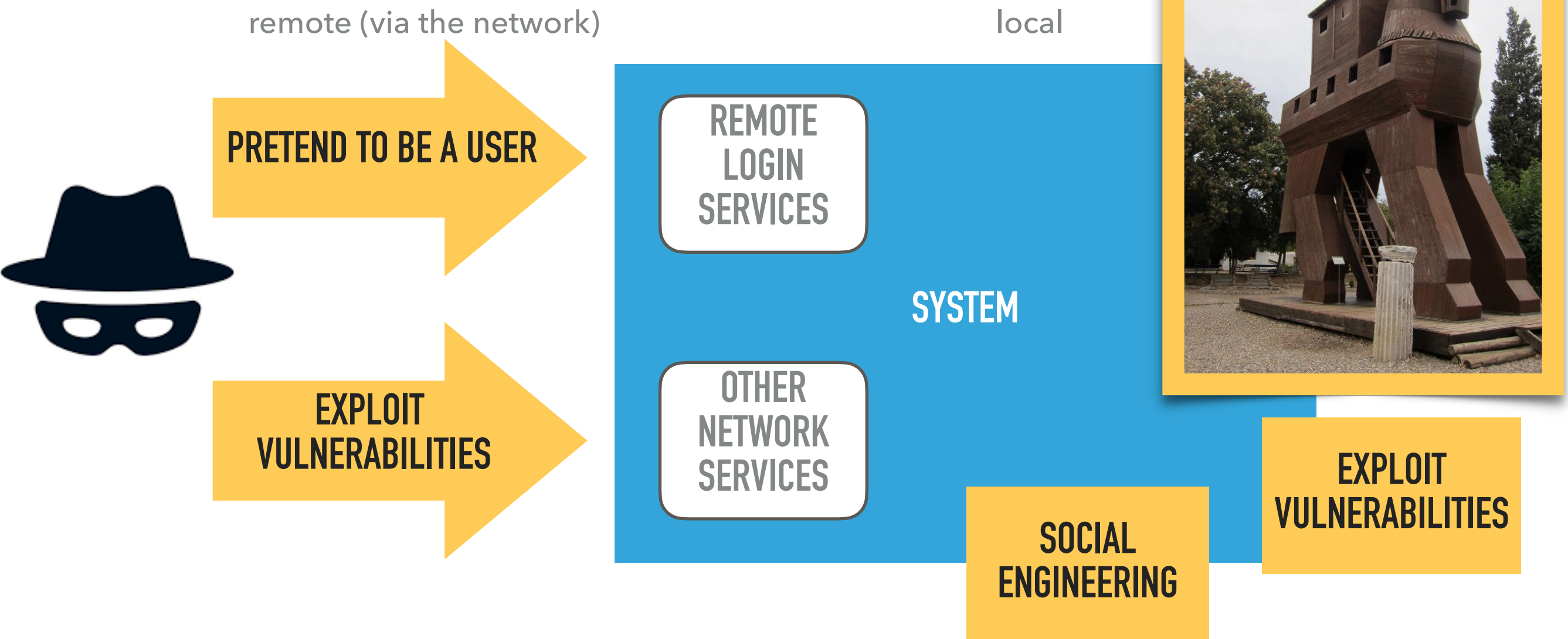


# HOW SYSTEMS ARE HACKED





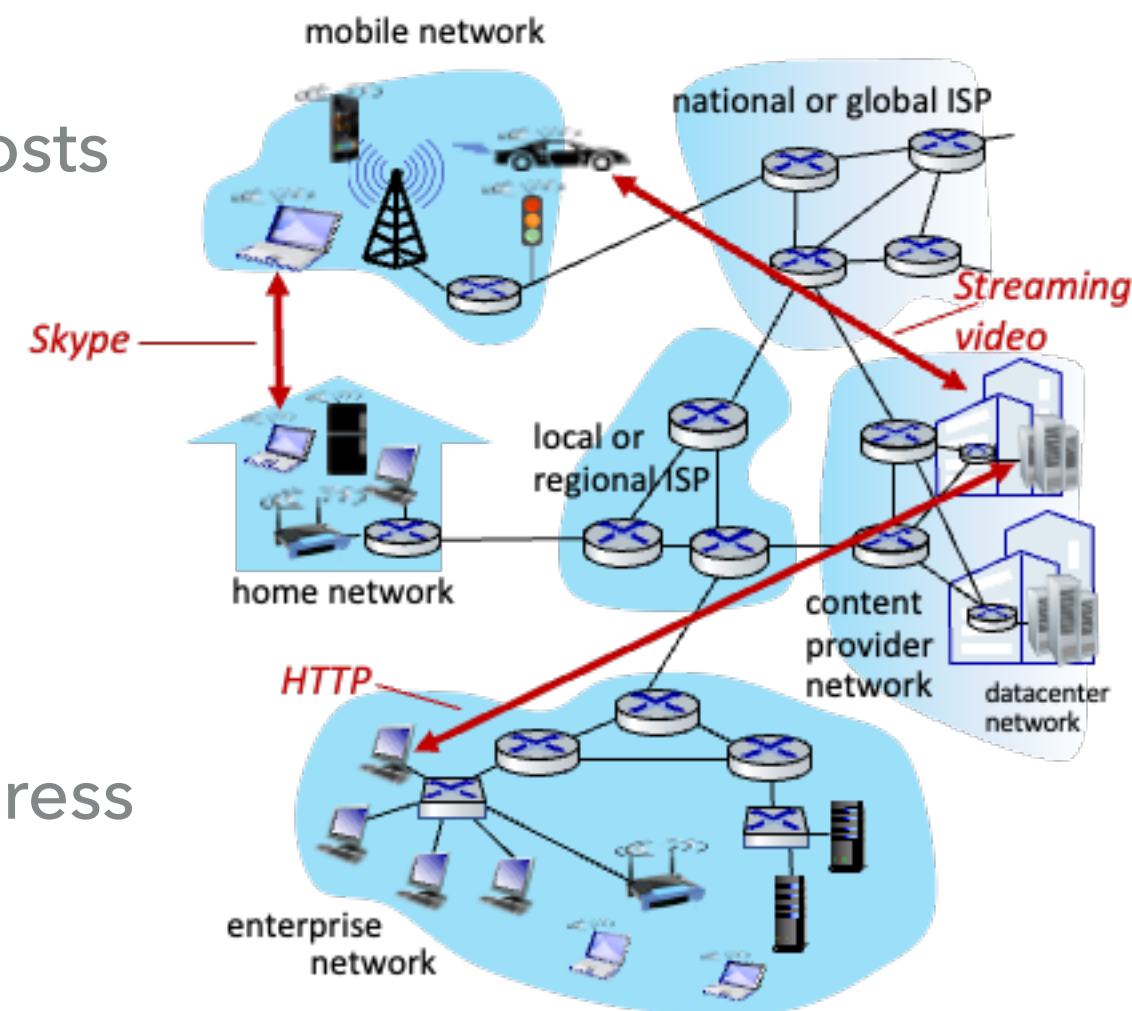
# HOW SYSTEMS ARE HACKED



# SO, WHAT IS A NETWORK?

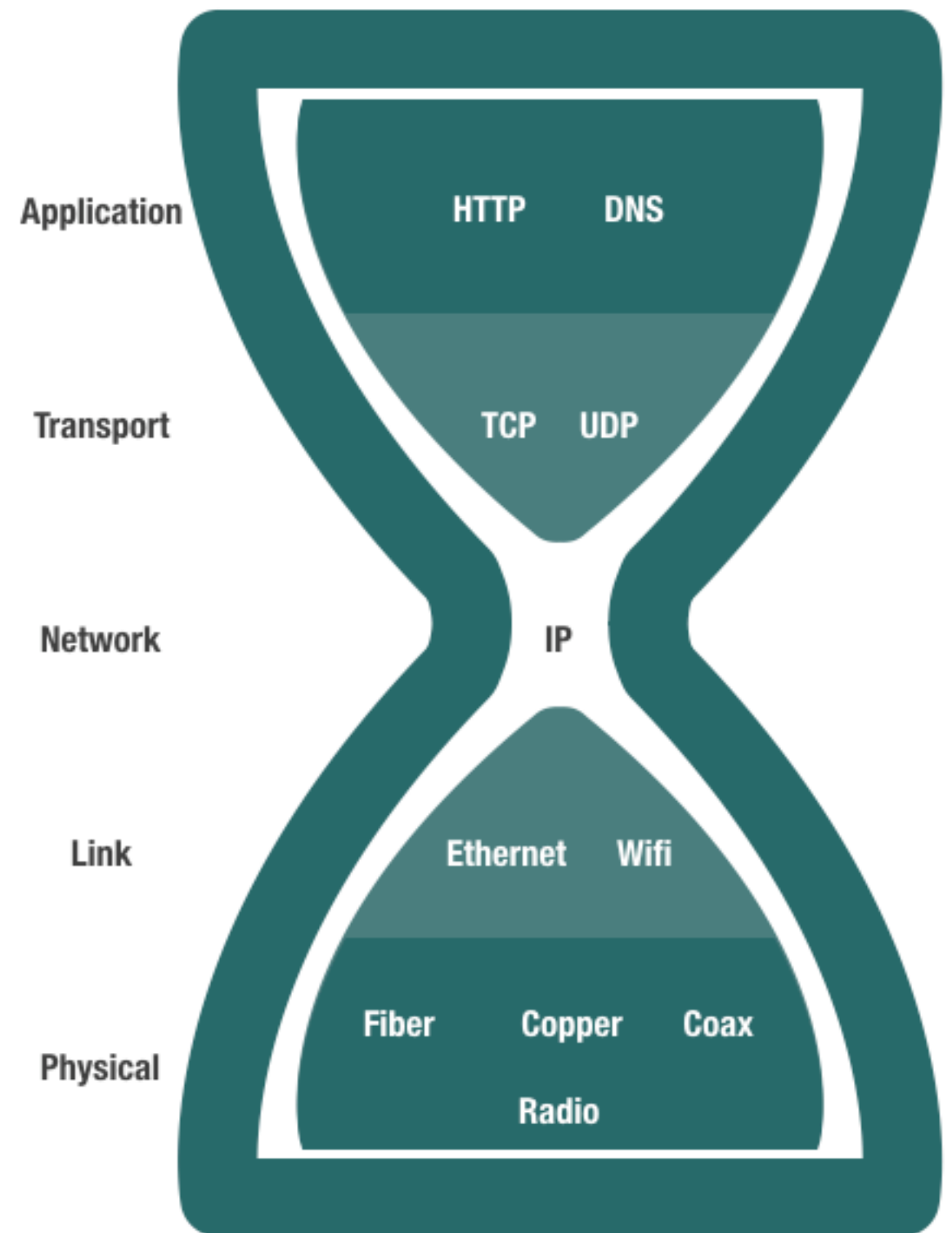
# THE INTERNET PROVIDES BEST-EFFORT SERVICE

- ▶ Global network that provides best-effort delivery of packets between connected hosts
- ▶ Packet: a structured sequence of bytes
  - ▶ Header: metadata used by network
  - ▶ Payload: user data to be transported
- ▶ Every host has a unique identifier – IP address
- ▶ Series of routers receive packets, look at destination address on the header and send it one hop towards the destination IP address

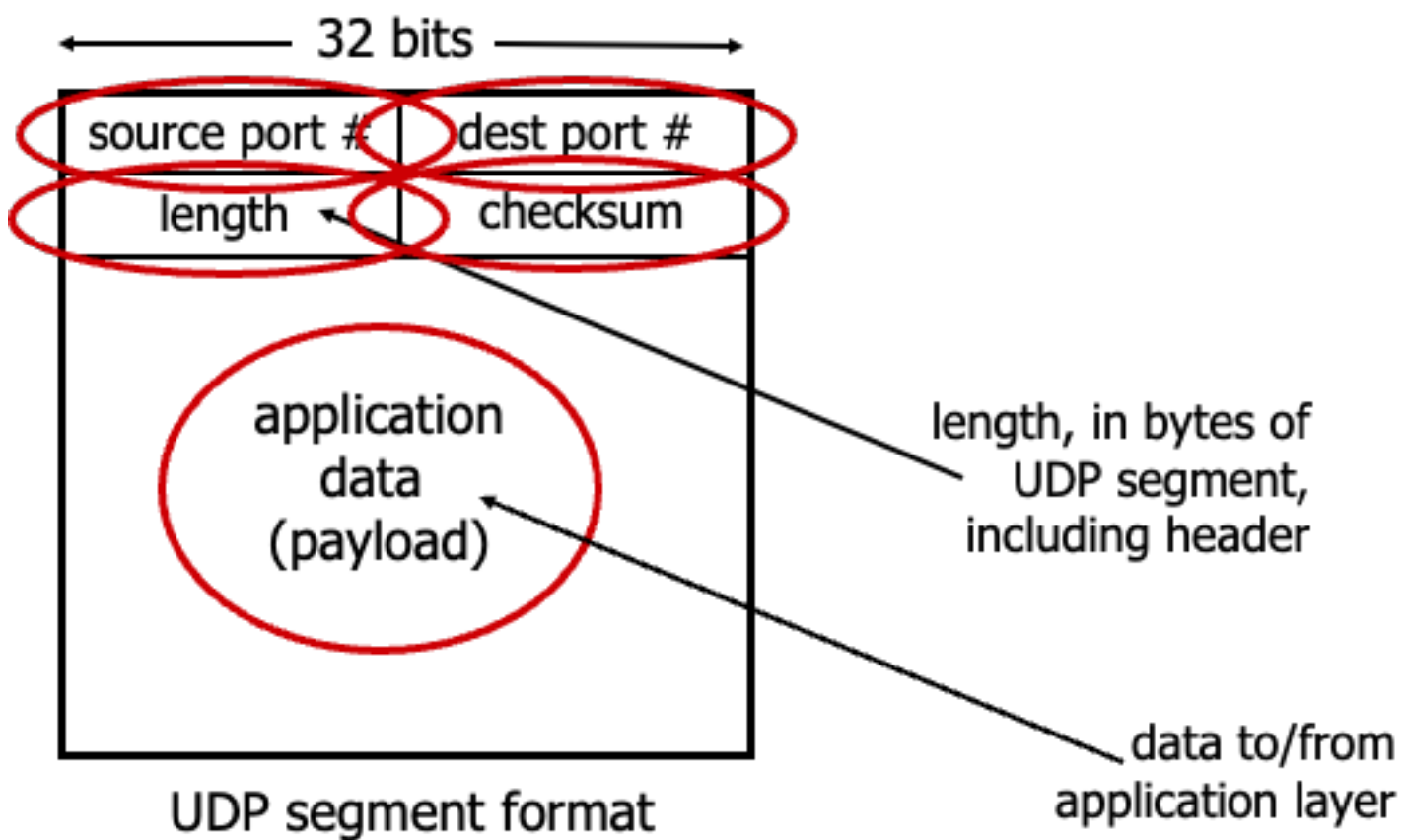


# NETWORK DESIGN PRINCIPLES

- ▶ The network is dumb and minimalistic
- ▶ Shift complexity to the end-points
- ▶ The focus has been on simple and robust connectivity, not security



# WHAT IS A PROTOCOL?

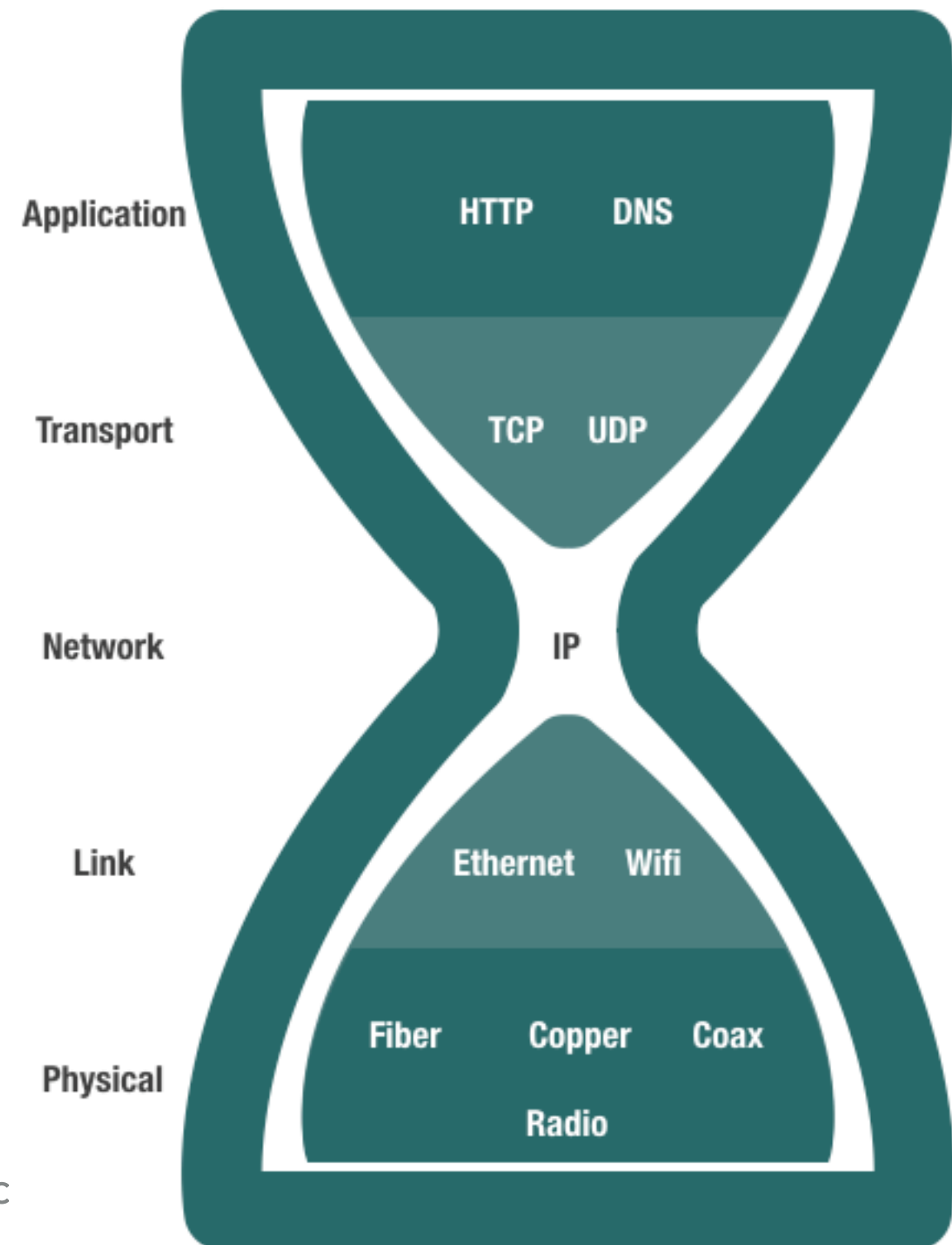


- ▶ We define how hosts communicate in published network protocols
- ▶ Syntax: How communication is structured (e.g., format and order of messages)
- ▶ Semantics: What communication means. Actions taken on transmit or receipt of message, or when a timer expires. What assumptions can be made.



# PROTOCOL LAYERING AND THE INTERNET HOURGLASS

- ▶ Networks use a stack of protocol layers
  - ▶ Each layer has different responsibilities.
  - ▶ Layers define abstraction boundaries
- ▶ Lower layers provide services to layers above
  - ▶ Don't care what higher layers do
- ▶ Higher layers use services of layers below
  - ▶ Don't worry about how it works



**SO MANY THINGS THAT  
A HACKER MAY DO**

# THREAT MODEL FOR NETWORK SECURITY

- ▶ Adversary can intercept / modify network traffic.
- ▶ Adversary can send packets.
- ▶ Adversary has full control of their own machines.
- ▶ Adversary can participate in protocols (usually).
  - ▶ Often not feasible to keep bad guys out of a large systems.

## SNIFF THE ENVIRONMENT



typically using a  
packet sniffer, i.e.,  
Wireshark,  
tcpdump, snort, etc.

**DEFENSES?**



## LEARN MORE ABOUT THE TARGET: SOCIAL SIDE



HMM...WHO OWNS  
CLASSDEEDEE

HMM...WHO OWNS  
MYCOURSEVILLE

**DEFENSES?**

For .th, we go to THNIC,  
the Thailand registry

[https://www.thnic.co.th/  
whois/](https://www.thnic.co.th/whois/)

For .com, we go to ICANN.

<https://lookup.icann.org/en>

Showing results for: MYCOURSEVILLE.COM

Original Query: mycourseville.com

# Contact Information

## Registrant Contact

Name: atiwong  
Organization: atiwong  
Mailing Address: atiwong atiwong,  
atiwong Krung Thep Maha Nakhon  
Bangkok 999999 TH  
Phone: +66.999999  
Ext:  
Fax:  
Fax Ext:  
Email:atiwong@gmail.com

## Admin Contact

Name: atiwong  
Organization: atiwong  
Mailing Address: atiwong atiwong,  
atiwong Krung Thep Maha Nakhon  
Bangkok 999999 TH  
Phone: +66.999999  
Ext:  
Fax:  
Fax Ext:  
Email:atiwong@gmail.com

## Tech Contact

Name: atiwong  
Organization: atiwong  
Mailing Address: atiwong atiwong,  
atiwong Krung Thep Maha Nakhon  
Bangkok 999999 TH  
Phone: +66.999999  
Ext:  
Fax:  
Fax Ext:  
Email:atiwong@gmail.com

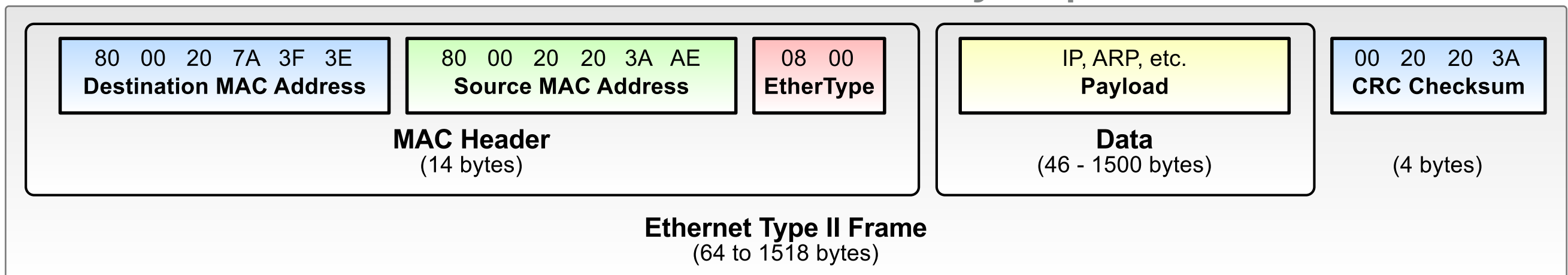
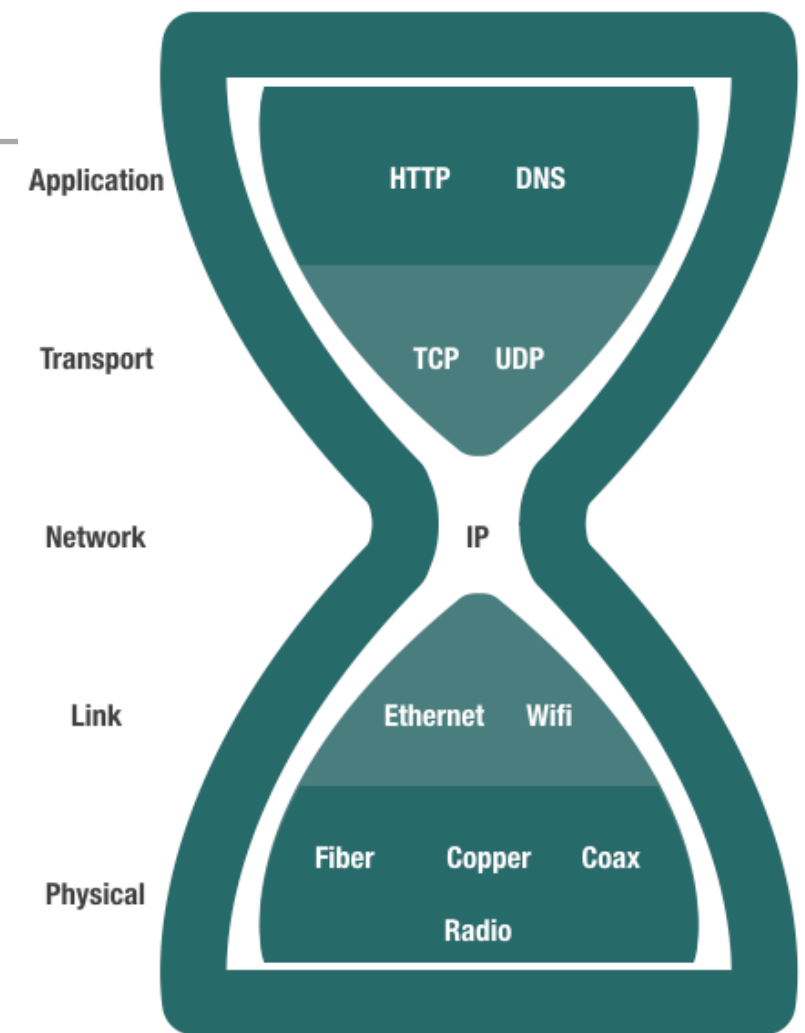
|   |
|---|
| Domain : VWIN.CO.TH   |
| ACE : VWIN.CO.TH  |
| Registrar : T.H.NIC Co., Ltd.   |
| Name Server : NS0.WISEWAYS.CO.TH<br>NS1.WISEWAYS.CO.TH  |
| Status : <span>Active</span>  |
| Updated Date : 2014-04-24 09:52:25  |
| Created Date : 1999-01-17 00:00:00  |
| Expiry Date : 2018-01-16 00:00:00   |
| Domain Type : Business  |
| Domain Holder : V.WIN Group Training Center Co.,Ltd.<br>13/111 Navamin Road Klongkum Buengkum, BANGKOK<br>10240<br>TH<br>Phone: +66-2-734-7411<br>Fax: +66-2-375-0237<br>Domain Contact: Kiart Piromsopa<br>Email Contact: kiart@wiseways.co.th |
| Technical Contact : V.WIN GROUP CO.,LTD.<br>13/110-111 NAWAMIN RD.KLONGKUM BUNGKUM, BANGKOK<br>10240<br>TH<br>Phone: +66-2-734-7411<br>Fax: +66-2-3750237<br>Domain Contact: KRERK PIROMSOPA<br>Email Contact: krerk@vwin.co.th                 |

**SO, LET'S GO  
LAYER BY LAYER**



# LINK LAYER

- ▶ Assumes: Local nodes are physically connected
- ▶ Task: Transfer bytes between two hosts on the physically connected network
- ▶ Ethernet is the most common link layer protocol.



## LINK LAYER HAS NO SECURITY: NO CIA

- ▶ Provides connectivity between hosts on a single Local Area Network
- ▶ Data is split into ~1500 byte Frames, which are addressed to a device's 48-bit physical (MAC) address – assigned by manufacturer
- ▶ Switches forward frames based on learning where different MACs are located. **No guarantees that frames are not sent to other hosts**
- ▶ Confidentiality?
- ▶ Authenticity?
- ▶ Integrity?

# MAC addresses

- 32-bit IP address:
  - *network-layer* address for interface
  - used for layer 3 (network layer) forwarding
  - e.g.: 128.119.40.136
- MAC (or LAN or physical or Ethernet) address:
  - function: used “locally” to get frame from one interface to another physically-connected interface (same subnet, in IP-addressing sense)
  - 48-bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
  - e.g.: 1A-2F-BB-76-09-AD

*hexadecimal (base 16) notation  
(each “numeral” represents 4 bits)*

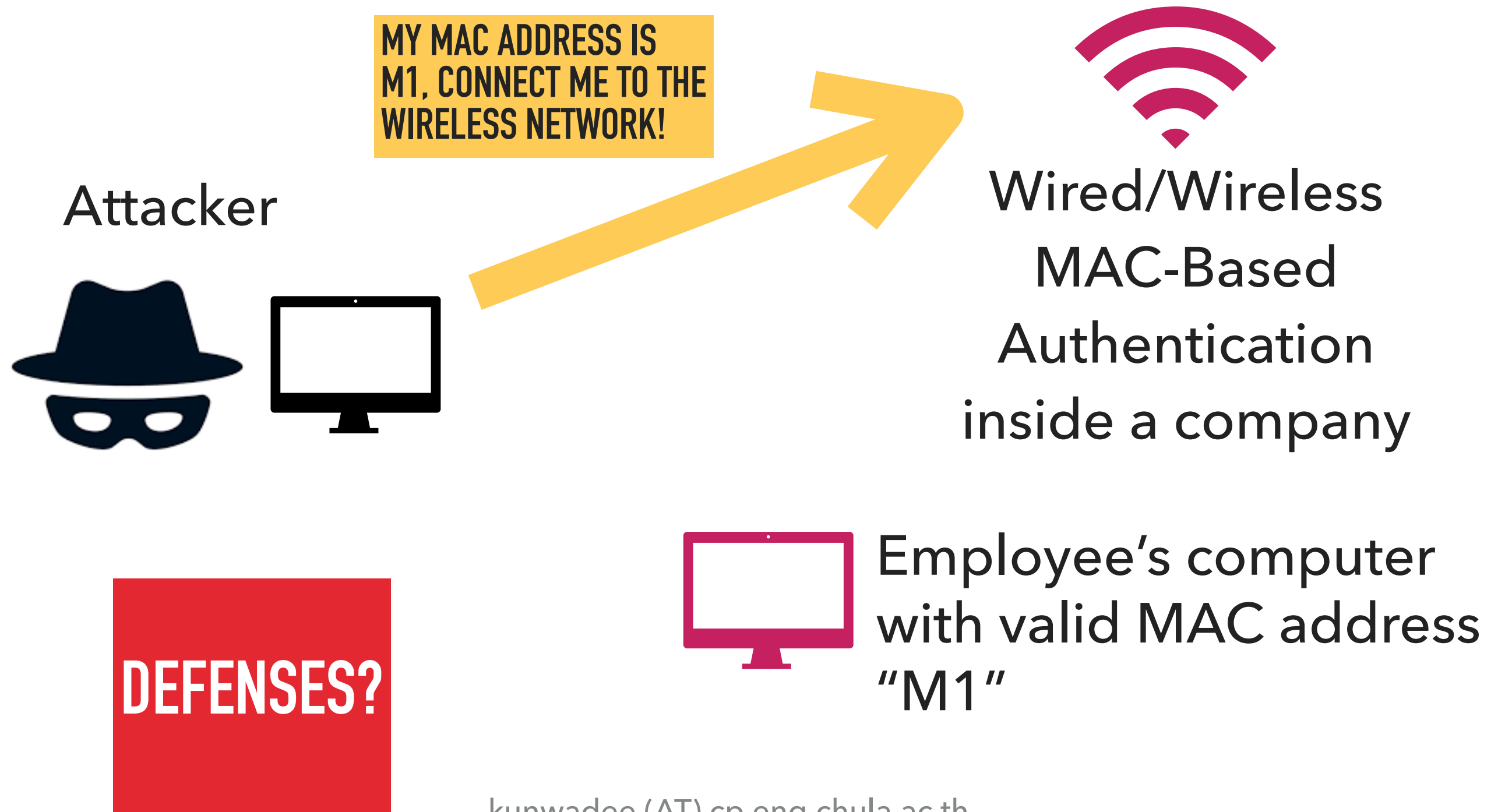
- 
1. SNIFFING
  2. MAC SPOOFING
  3. ARP SPOOFING/  
POISONING
  4. MAC ADDRESS PRIVACY

---

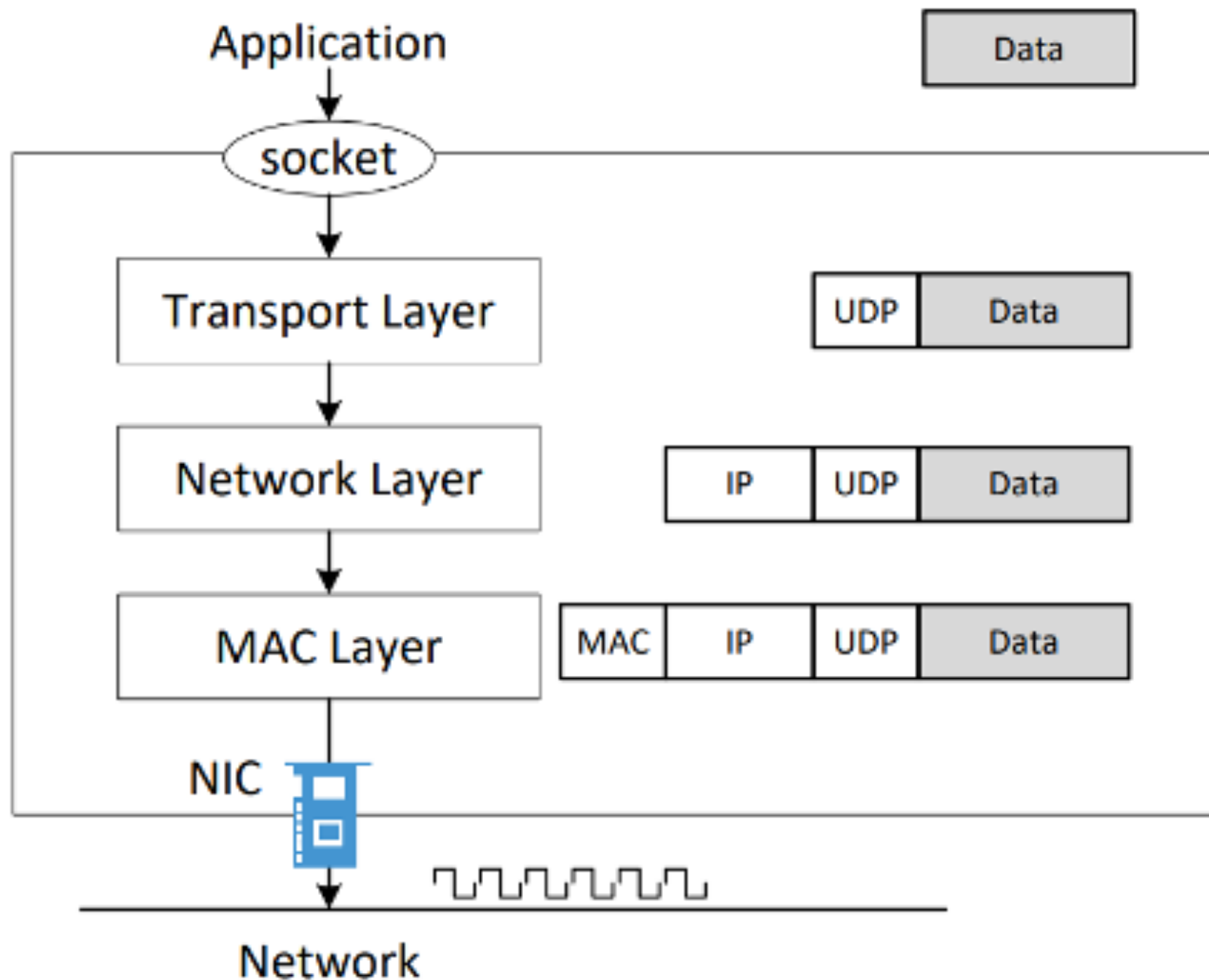
## KNOWN ISSUES



# ATTACK IN STEALTH: SPOOF MAC ADDRESSES



# HOW TO SPOOF



In normal packet construction

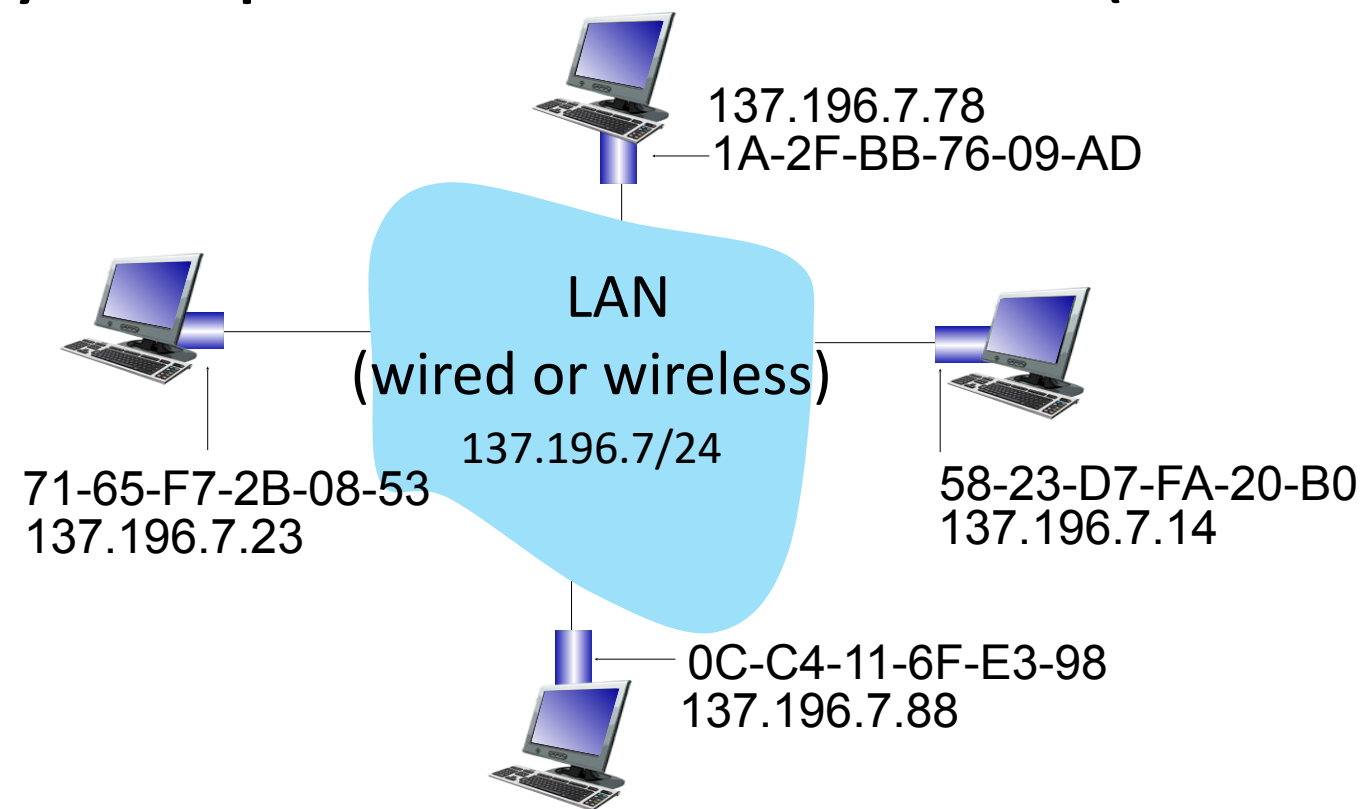
- Only some selected header fields can be set by users

- OS set the other fields

# MAC addresses

each interface on LAN

- has unique 48-bit **MAC** address
- has a locally unique 32-bit IP address (as we've seen)



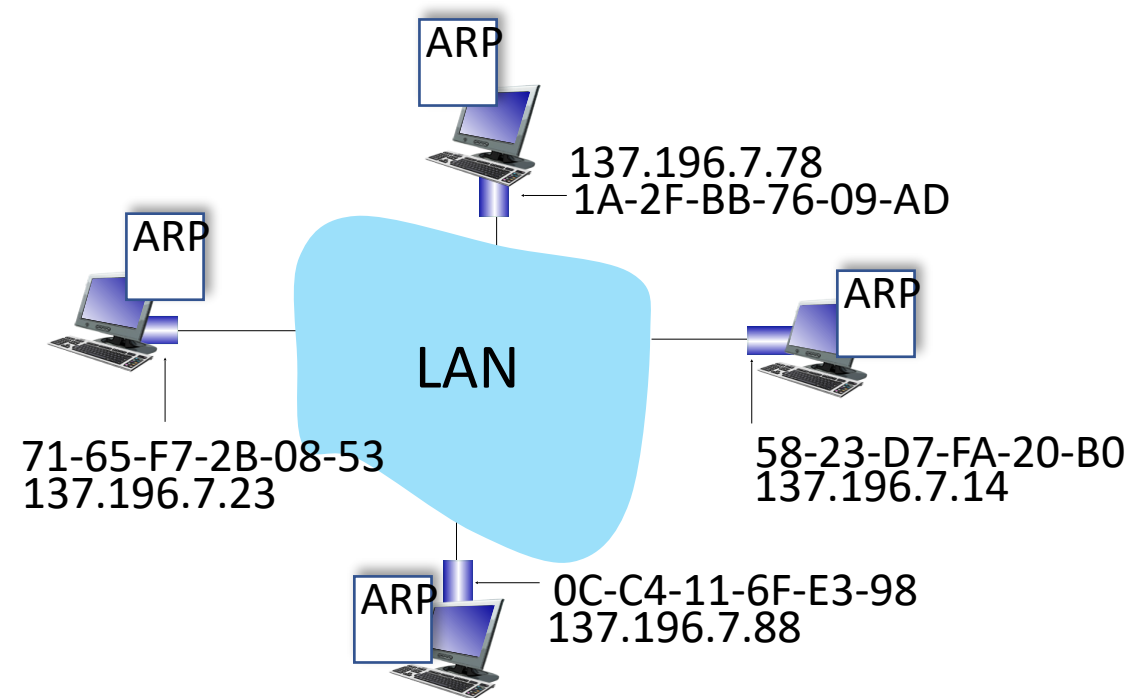
# MAC addresses

- MAC address allocation administered by IEEE
- manufacturer buys portion of MAC address space (to assure uniqueness)
- analogy:
  - MAC address: like Social Security Number
  - IP address: like postal address
- MAC flat address: portability
  - can move interface from one LAN to another
  - recall IP address *not* portable: depends on IP subnet to which node is attached



# ARP: address resolution protocol

*Question:* how to determine interface's MAC address, knowing its IP address?



**ARP table:** each IP node (host, router) on LAN has table

- IP/MAC address mappings for some LAN nodes:  
< IP address; MAC address; TTL >
- TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

# ARP protocol in action

example: A wants to send datagram to B

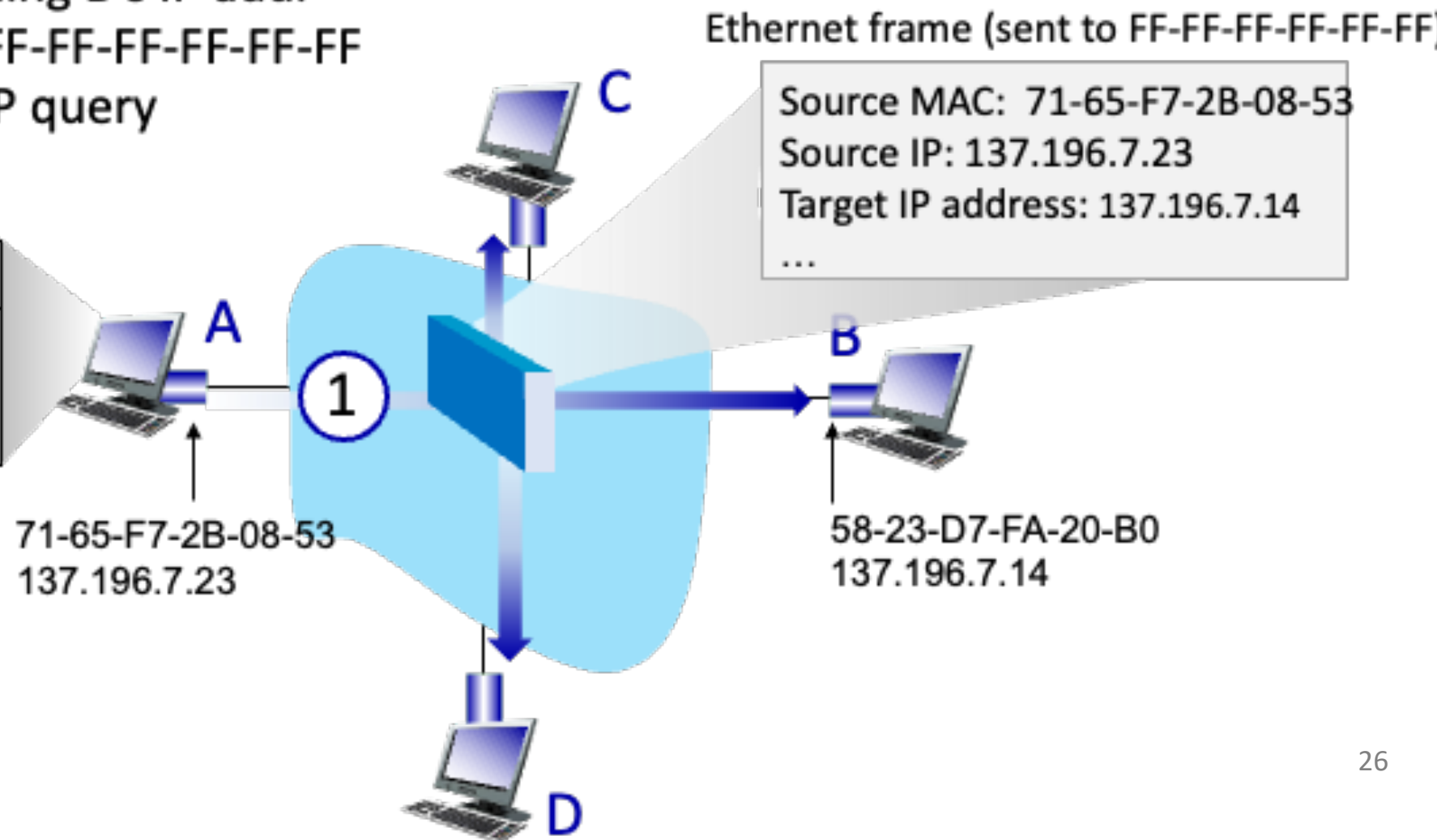
- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address

A broadcasts ARP query, containing B's IP addr

- ①
- destination MAC address = FF-FF-FF-FF-FF-FF
  - all nodes on LAN receive ARP query

ARP table in A

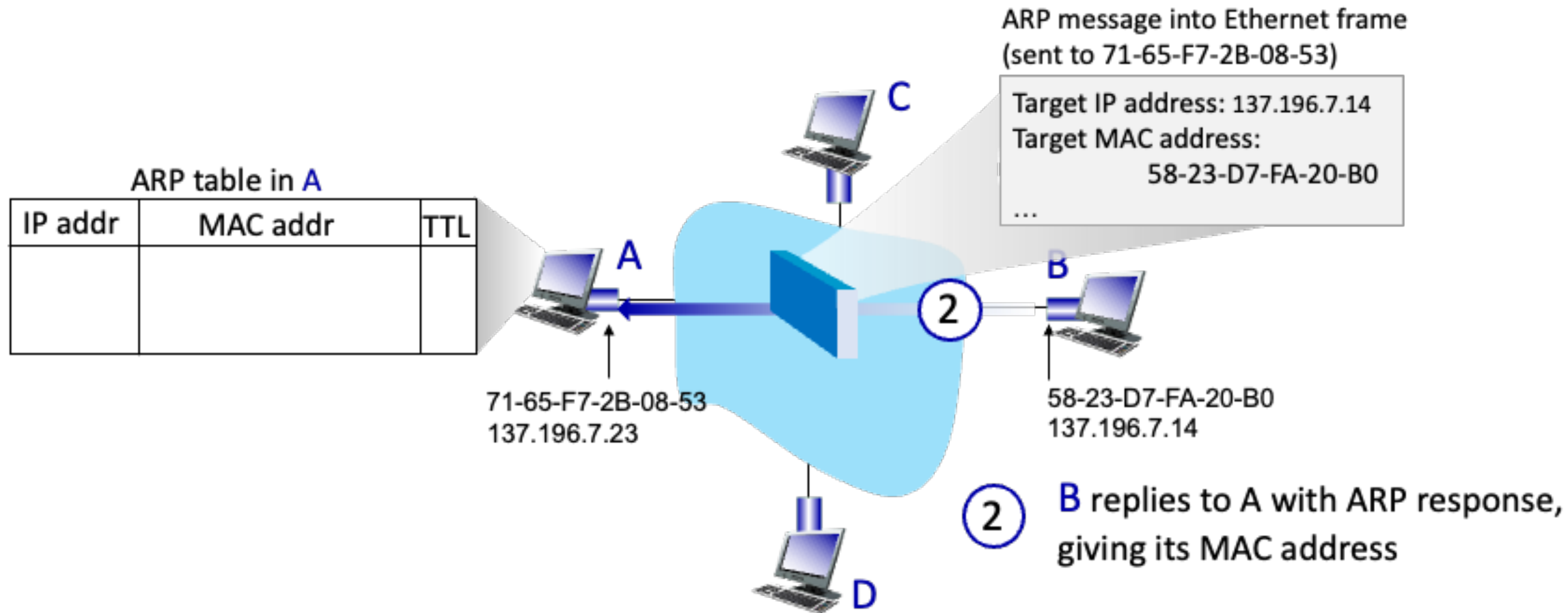
| IP addr | MAC addr | TTL |
|---------|----------|-----|
|         |          |     |



# ARP protocol in action

example: A wants to send datagram to B

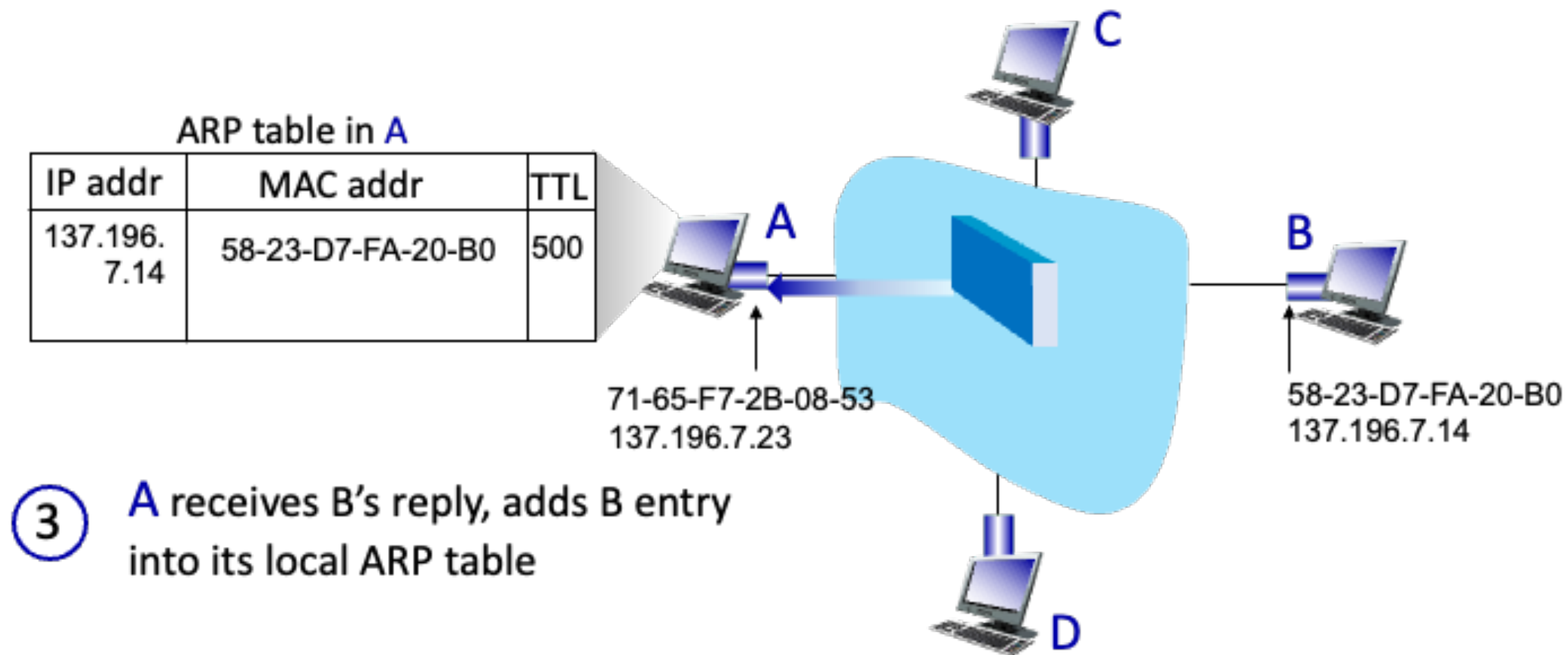
- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address



# ARP protocol in action

example: A wants to send datagram to B

- B's MAC address not in A's ARP table, so A uses ARP to find B's MAC address



## ARP SPOOFING OR ARP POISONING

- ▶ Any host on the LAN can send ARP requests and replies: any host can claim to be another host on the local network!
- ▶ This is called ARP spoofing (the act of faking another person's MAC address) or ARP poisoning (the act of corrupting one or more victims' ARP table)
- ▶ This allows any host  $X$  to force IP traffic between any two other hosts  $A$  and  $B$  to flow through  $X$  (MitM!)
  - ▶ Claim  $IP_A$  is at attacker's MAC address  $M_X$
  - ▶ Claim  $IP_B$  is at attacker's MAC address  $M_X$

**DEFENSES?**

# MAC ADDRESS RANDOMIZATION AND PRIVACY

## iOS 8 to stymie trackers and marketers with MAC address randomization

When searching for Wi-Fi networks, iOS8 devices can hide their true identities.

by Lee Hutchinson - Jun 9, 2014 10:56am EDT

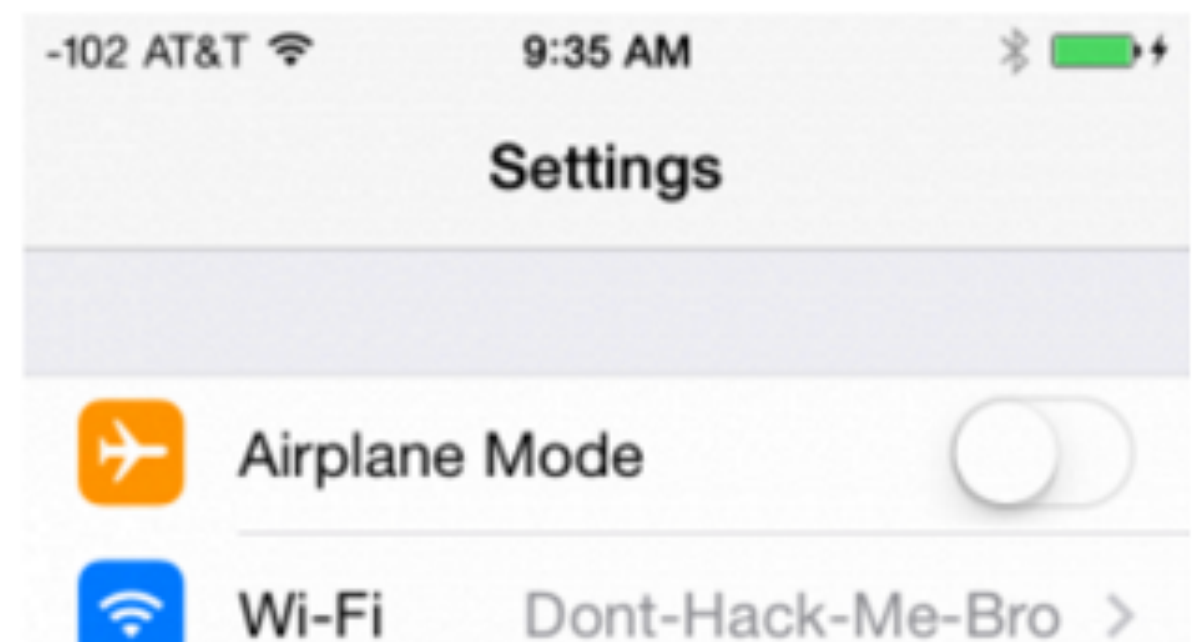
 Share

 Tweet

 Email

88

Quartz is **reporting a change** to how iOS 8-equipped devices search out Wi-Fi networks with which to connect. The new mobile operating system, which is on track for a release in the fall, gives iOS 8 devices the ability to identify themselves not with their unique burned-in hardware MAC address but rather with a random, software-supplied address instead.



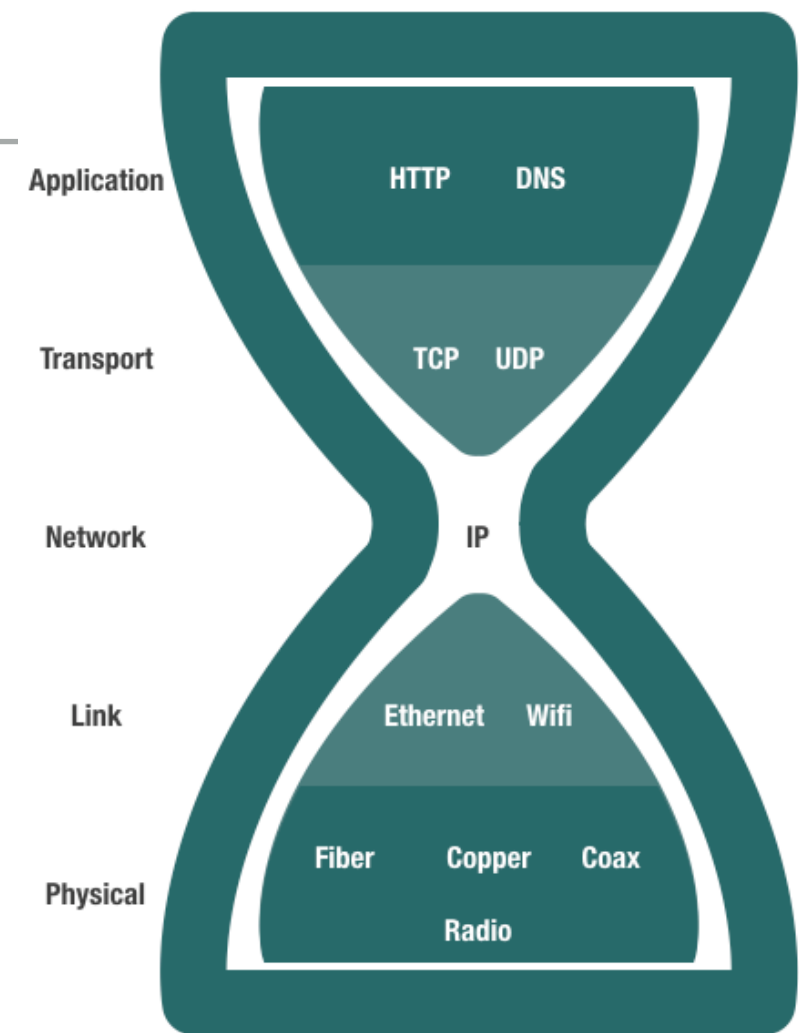


Keep in mind....

**SECURITY PROBLEM INHERENT  
IN THE DESIGN “PROTOCOL-  
LEVEL PROBLEM” VS. BUG**

# NETWORK LAYER

- ▶ Internet Protocol (IP) defines what packets that cross the Internet need to look like to be processed by routers
- ▶ Every host is assigned a unique identifier ("IP Address")
- ▶ Every packet has an IP header that indicates its sender and receiver
- ▶ Routers forward packet along to try to get it to the destination host
- ▶ Rest of the packet should be ignored by the router



# IP SERVICE MODEL IS BEST EFFORT: SIMPLE AND FAST

## ► Yes:

- Routing. If host knows IP of destination host, route packet to it.
- Fragmentation and reassembly: Split data into packets and reassemble
- Error reporting: (maybe, if you're lucky) tell source it dropped your packet

## ► No:


- Everything else. No ordering. No retransmission. No (real) error checking. No acknowledgement of receipt. No "connections". No security. Just packets. Packets can be dropped, corrupted, repeated, or reordered.

## NO CIA: SIMILAR ATTACKS TO LINK LAYER

- ▶ Sniffing
- ▶ IP spoofing
- ▶ DHCP attacks

**DEFENSES?**

**But, there's more!!!**

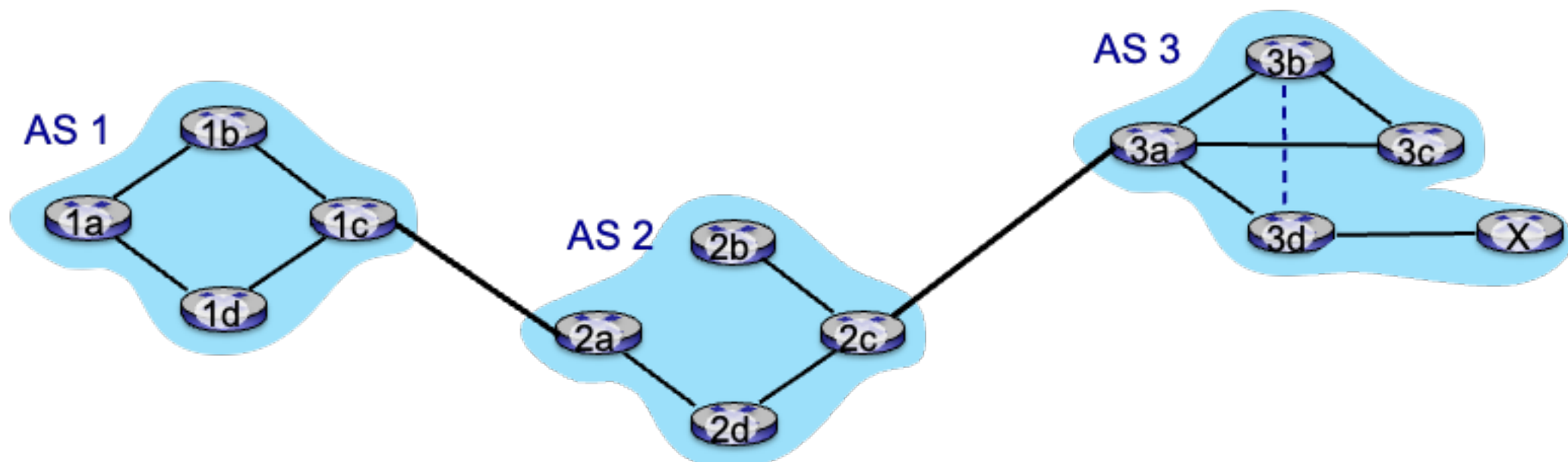
- 
1. ROUTING
  2. IP FRAGMENTATION
  3. ICMP
  4. LIVING AN EXPOSED LIFE (NMAP)

---

## KNOWN ISSUES

## WHAT ABOUT INTERNET ROUTING? IS IT SECURE?

- ▶ BGP (Border Gateway Protocol): protocol that allows routers to exchange information about their routing tables
- ▶ Each router announces what it can route to all of its neighbors.
- ▶ Every router maintains a global table of routes
- ▶ Routers 'believe' what they hear from their neighbors





## BGP PREFIX HIJACKING ATTACKS HAPPEN...

- ▶ On 24 February 2008, Pakistan Telecom (AS 17557) began advertising a small part of YouTube's (AS 36561) assigned network
- ▶ PCCW (3491) did not validate Pakistan Telecom's (17557) advertisement for 208.65.153.0/24
- ▶ Youtube offline.



**DEFENSES?**

**ALSO COMMON ARE MISCONFIGURATION "OOPS"  
AND PHYSICAL "OOPS."**

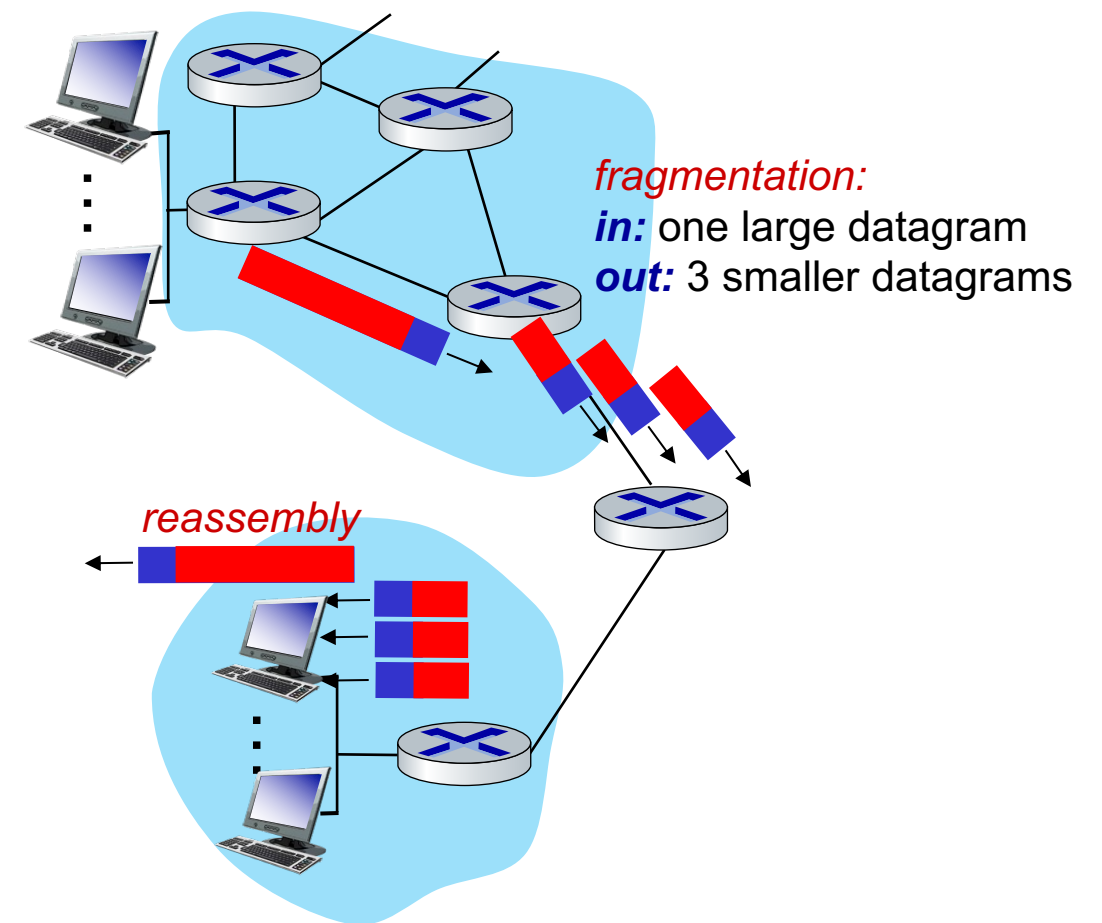
## IS SECURE BGP (SBGP) THE SOLUTION

- ▶ Cryptographic signing of route announcements.
- ▶ Must know who is allowed to announce every particular IP prefix.
- ▶ Requires someone to distribute keys / certificates for every IP prefix.
- ▶ Bootstrapping problem is tricky; some performance overheads too.
- ▶ Getting some traction but still not widely deployed.

# WHAT ABOUT IP FRAGMENTATION?

## IP fragmentation/reassembly

- network links have MTU (max. transfer size) - largest possible link-level frame
  - different link types, different MTUs
- large IP datagram divided (“fragmented”) within net
  - one datagram becomes several datagrams
  - “reassembled” only at *destination*
  - IP header bits used to identify, order related fragments



# TONS OF ATTACKS

- ▶ Tons of attacks
  - ▶ Low bandwidth method that causes DOS-style resource exhaustion at target during fragment reassembly
  - ▶ Ping-of-death
  - ▶ Abnormal (invalid) flags to rewrite headers (teardrop attack) or exploit vulnerabilities
  - ▶ Evade detection/bypass firewalls
- ▶ This site has good examples: <https://www.imperva.com/learn/ddos/ip-fragmentation-attack-teardrop/>

# ICMP: INTERNET CONTROL MESSAGE PROTOCOL

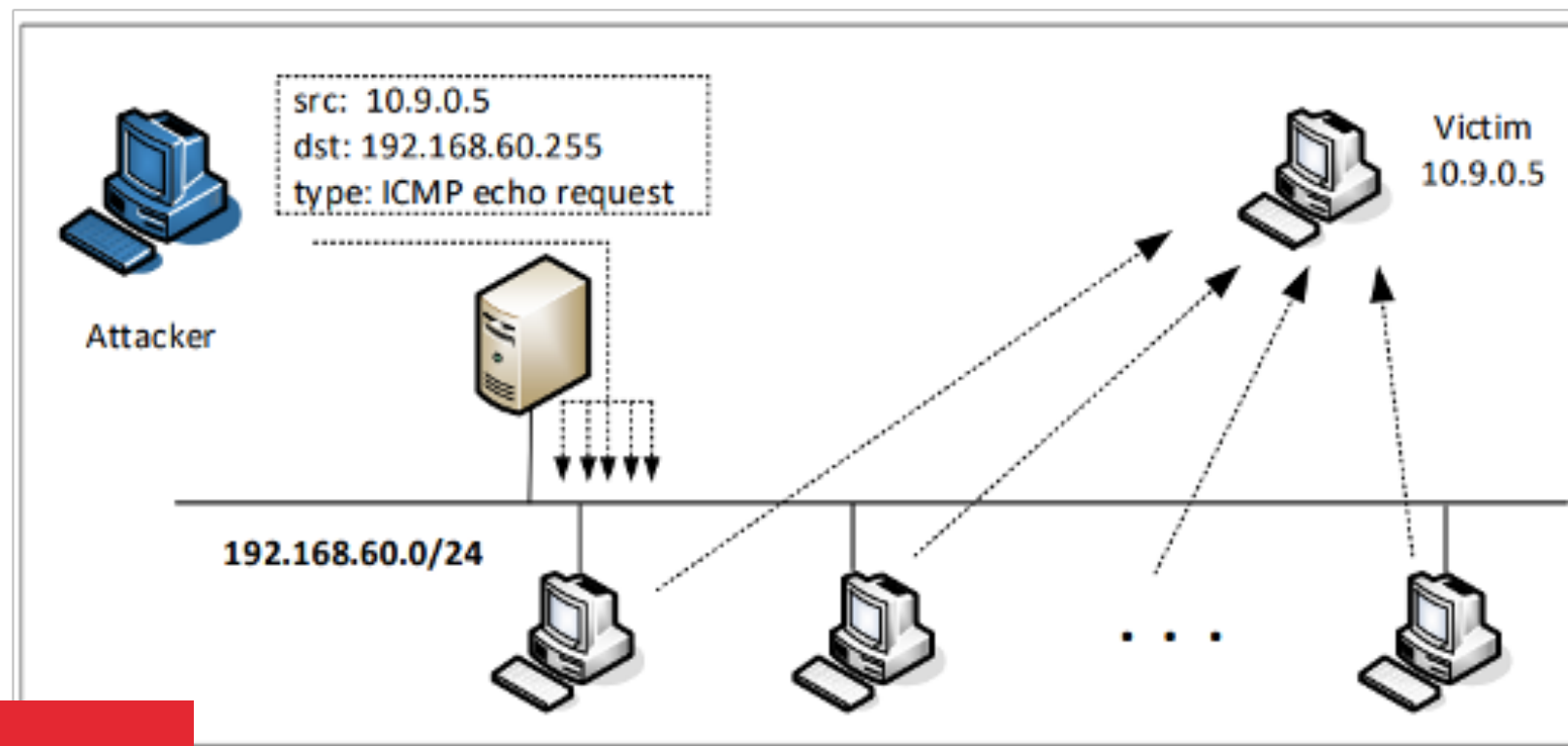
- used by hosts and routers to communicate network-level information
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- network-layer “above” IP:
  - ICMP messages carried in IP datagrams
- *ICMP message*: type, code plus first 8 bytes of IP datagram causing error

| <u>Type</u> | <u>Code</u> | <u>description</u>                            |
|-------------|-------------|---|
| 0           | 0           | echo reply (ping)                             |
| 3           | 0           | dest. network unreachable                     |
| 3           | 1           | dest host unreachable                         |
| 3           | 2           | dest protocol unreachable                     |
| 3           | 3           | dest port unreachable                         |
| 3           | 6           | dest network unknown                          |
| 3           | 7           | dest host unknown                             |
| 4           | 0           | source quench (congestion control - not used) |
| 8           | 0           | echo request (ping)                           |
| 9           | 0           | route advertisement                           |
| 10          | 0           | router discovery                              |
| 11          | 0           | TTL expired                                   |
| 12          | 0           | bad IP header                                 |

5 ——— redirect

# Smurf Attack

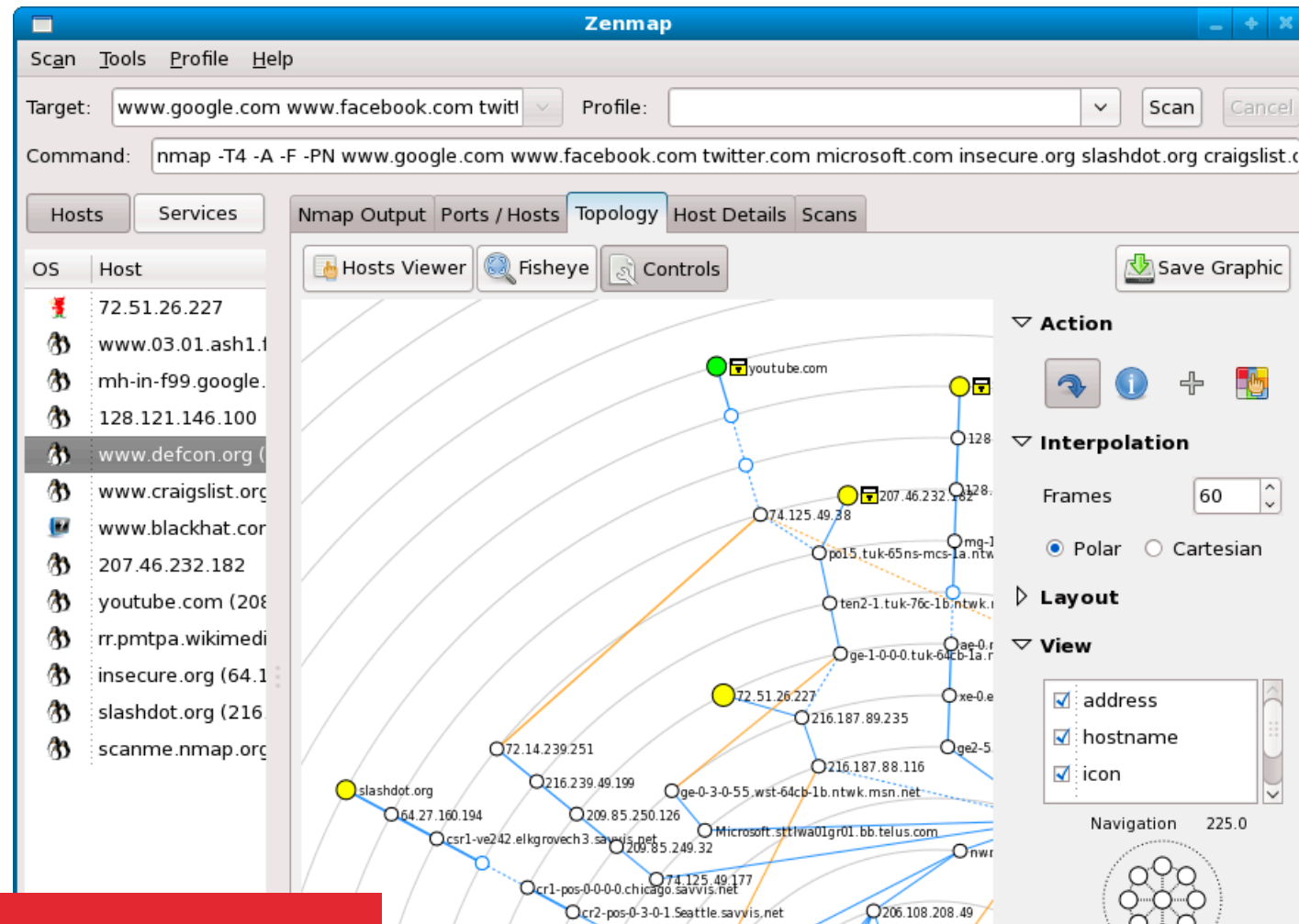
- Direct broadcast address
  - Example: 192.168.60.**255** for network 192.168.60.0/24



**DEFENSES?**

Routers now block directed broadcast (packets sent to broadcast addresses).

# HOW CAN A HACKER LEARN MORE ABOUT THE TARGET: NETWORK SIDE



map the target/target network

typically using nmap

look for open services,  
OS fingerprinting,  
service fingerprinting

DEFENSES?

**BE CAREFUL WHEN YOU USE NMAP AS YOU COULD TRIGGER ACTION FROM THE TARGET**



## ACTIVE OS FINGERPRINTING: HOW



- ▶ Different OS's have different TCP/IP behavior
- ▶ Send the target a bunch of 'crafted' TCP packets and observe the response
- ▶ Map back to known fingerprints to determine OS

# IDENTIFY POSSIBLE VULNERABILITIES IN NETWORK SERVICES

- ▶ OS vulnerabilities (Network stack)
- ▶ Network service vulnerabilities
- ▶ Application vulnerabilities

DEFENSES?



## Overview

The Vulnerability Notes Database provides information about software vulnerabilities. Vulnerability Notes include summaries, technical details, remediation information, and lists of affected vendors. Most Vulnerability Notes are the result of private coordination and disclosure efforts. For more comprehensive coverage of public vulnerability reports, consider the National Vulnerability Database (NVD). + Read More

## Recent Vulnerability Notes

|             |           |   |                |
|-------------|-----------|---|----------------|
| 16 Oct 2017 | VU#307015 | Infineon RSA library does not properly generate RSA key pairs         | CVE-2017-15361 |
| 16 Oct 2017 | VU#228519 | Wi-Fi Protected Access (WPA) handshake traffic can be manipul...      | Multiple CVEs  |
| 12 Oct 2017 | VU#590639 | NXP Semiconductors MQX RTOS contains multiple vulnerabilities         | Multiple CVEs  |
| 02 Oct 2017 | VU#973527 | Dnsmasq contains multiple vulnerabilities                             | Multiple CVEs  |
| 13 Sep 2017 | VU#101048 | Microsoft .NET framework SOAP Moniker PrintClientProxy remot...       | CVE-2017-8759  |
| 12 Sep 2017 | VU#240311 | Multiple Bluetooth implementation vulnerabilities affect many devi... | Multiple CVEs  |
| 08 Sep 2017 | VU#166743 | Das U-Boot AES-CBC encryption implementation contains multip...       | Multiple CVEs  |
| 06 Sep 2017 | VU#112992 | Apache Struts 2 framework REST plugin insecurely deserializes ...     | CVE-2017-9805  |
| 29 Aug 2017 | VU#403768 | Akeo Consulting Rufus fails to update itself securely                 | CVE-2017-13083 |
| 03 Aug 2017 | VU#824672 | Microsoft Windows automatically executes code specified in shor...    | CVE-2017-8464  |

kunwa

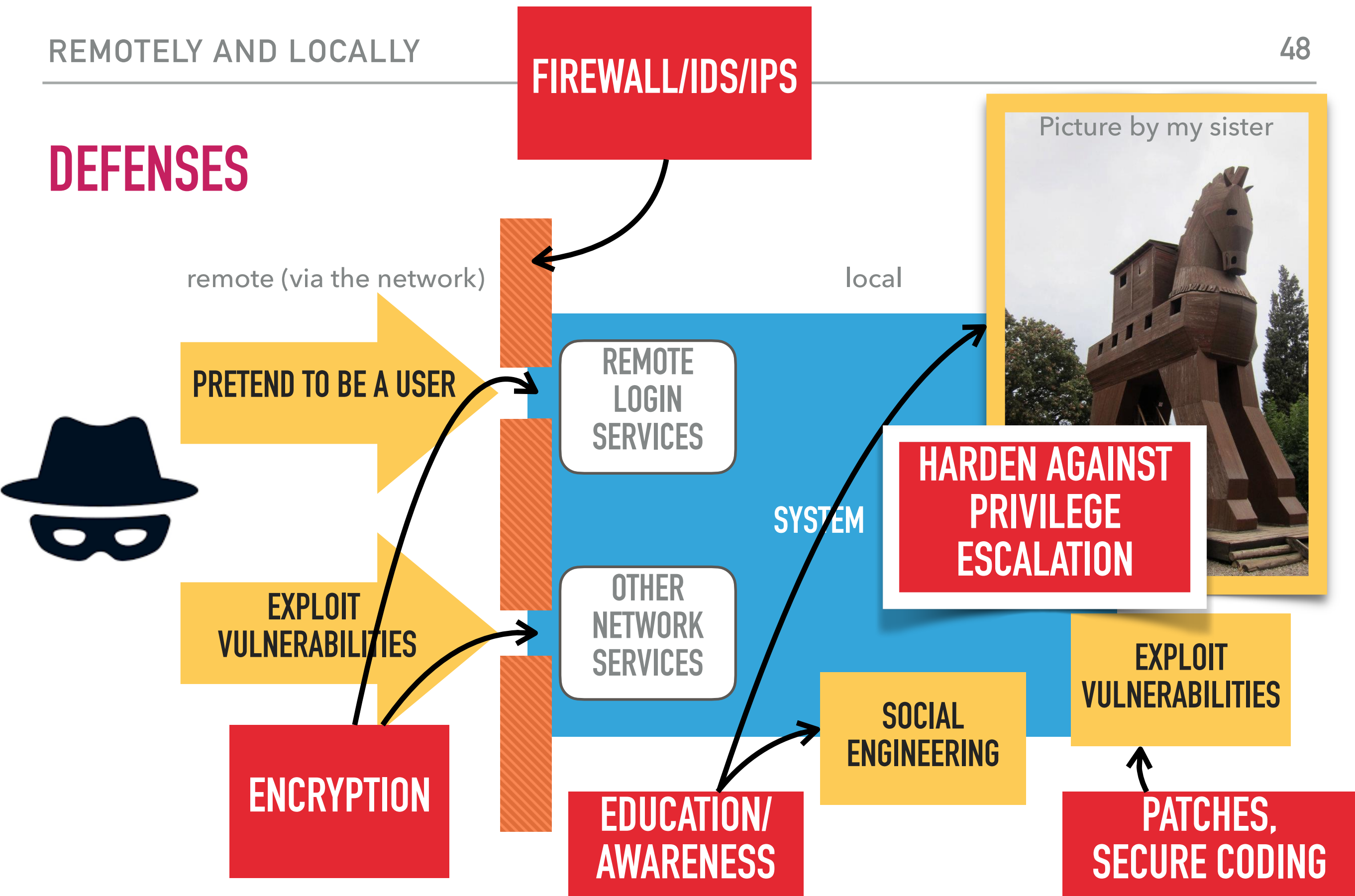
## WHY ARE THINGS SO INSECURE AT THE TCP/IP LEVEL?

- ▶ Historically, designers did not worry as much about security.
  - ▶ Even Bellovin says: "The Internet in 1989 was a much friendlier place".
  - ▶ Original Internet had a small number of relatively trustworthy users.
  - ▶ Design requirements changed over time.
- ▶ End-to-end argument in action.
  - ▶ Must provide security at the application level anyway.
  - ▶ Things are "good enough" at the transport level to let applications work.
- ▶ Some fixes do get added, but only for the worst problems / easier solutions.

# ARE THERE NO DEFENSES, BUT ONLY WAYS TO IMPROVE SECURITY?

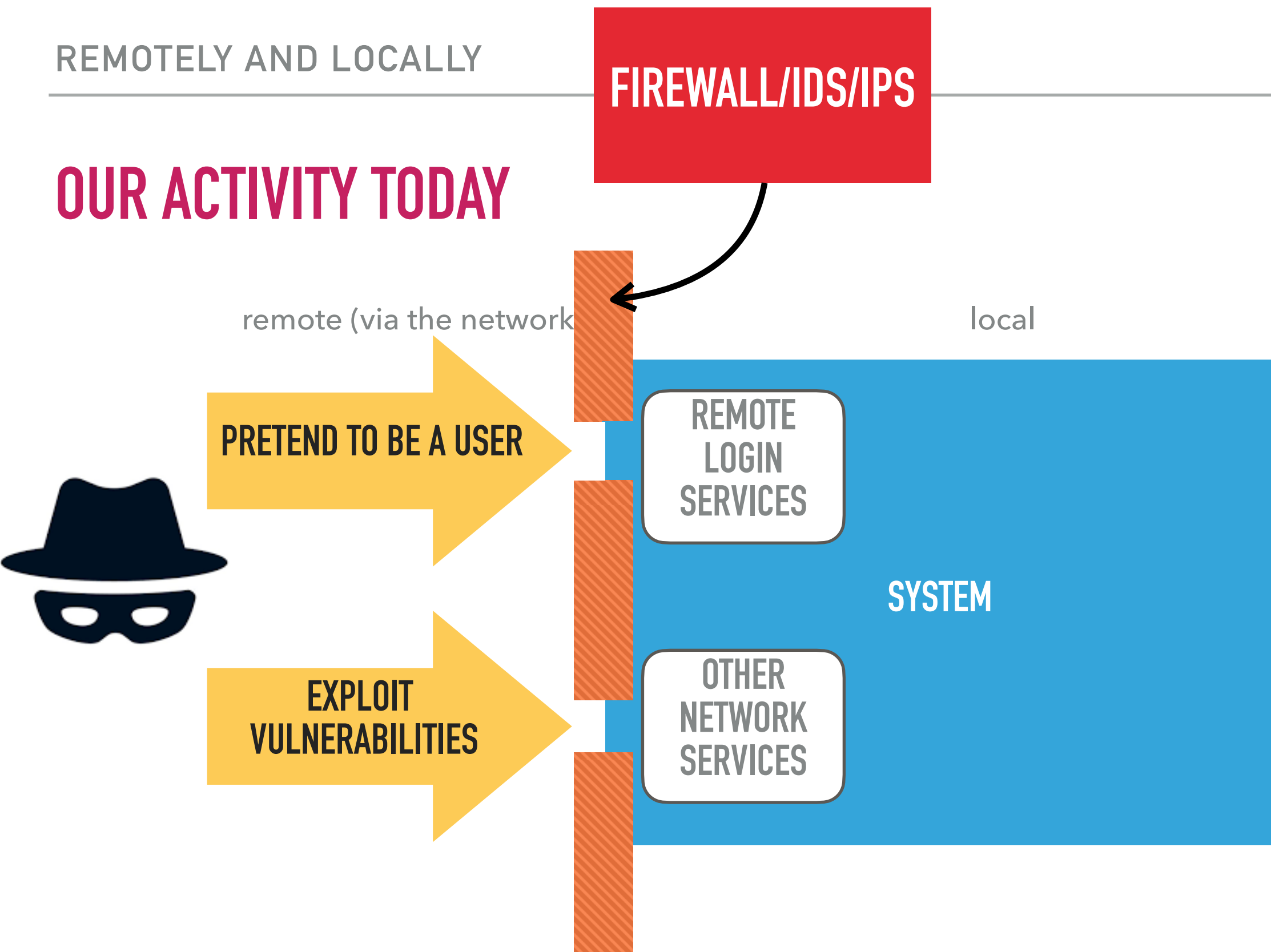
- ▶ Protocol-compatible fixes to TCP implementations.
- ▶ Firewalls.
  - ▶ Partial fix, but widely used.
  - ▶ Issue: adversary may be within firewalled network.
  - ▶ Issue: hard to determine if packet is "malicious" or not.
  - ▶ Issue: even for fields that are present (src/dst), hard to authenticate.
- ▶ TCP/IP's design not a good match for firewall-like filtering techniques.
  - ▶ E.g., IP packet fragmentation: TCP ports in one packet, payload in another.
- ▶ Implement security on top of TCP/IP: SSL/TLS, Kerberos, SSH, etc.
- ▶ Use cryptography (encryption, signing, MACs, etc).
  - ▶ Quite a hard problem: protocol design, key distribution, trust, etc.
- ▶ Some kinds of security hard to provide on top: DoS-resistance, routing.
- ▶ Deployment of replacement protocols: SBGP, DNSSEC.

DEFENSES



**ASSUME THE NETWORK IS  
OUT TO GET YOU.**

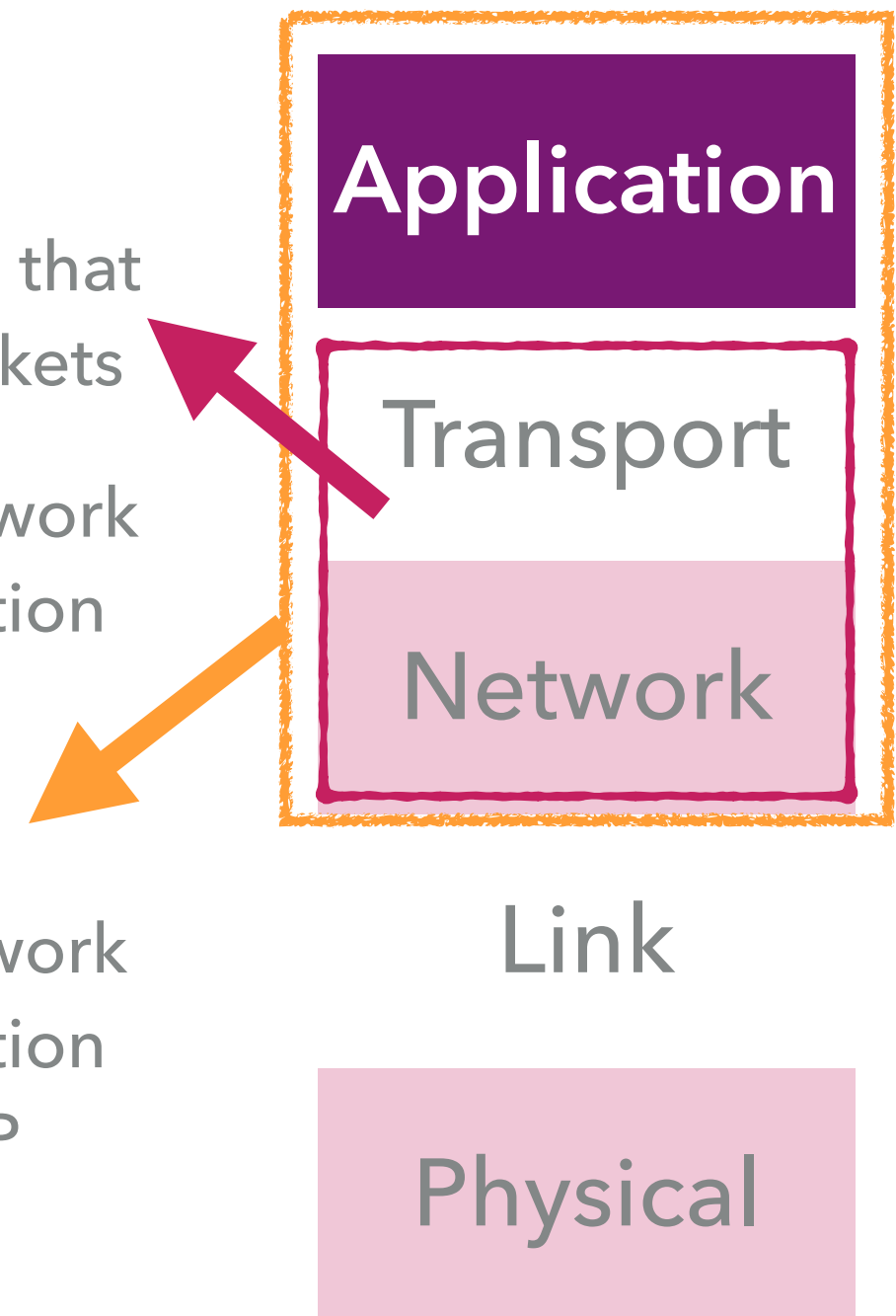
# OUR ACTIVITY TODAY





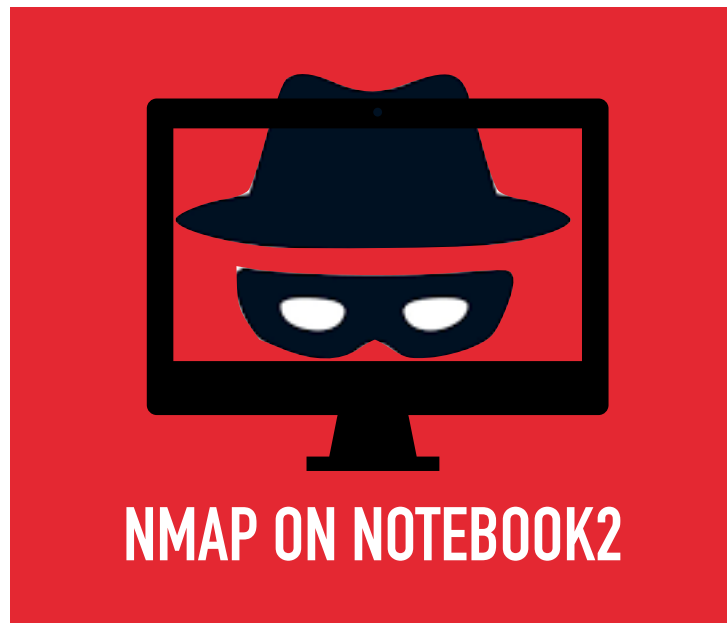
## FIREWALL VS. INTRUSION DETECTION SYSTEM VS. INTRUSION PREVENTION SYSTEM

- ▶ Firewall: Analyze packet headers for IP addresses (network layer) and port numbers (transport layer) that match pre-configured rules to DROP/ACCEPT packets
- ▶ IDS: Analyze packet headers for IP addresses (network layer), port numbers (transport layer), and application data (application layer) that match rules and generates logs of the event
- ▶ IPS: Analyze packet headers for IP addresses (network layer), port numbers (transport layer), and application data (application layer) that match rules and DROP packets that match

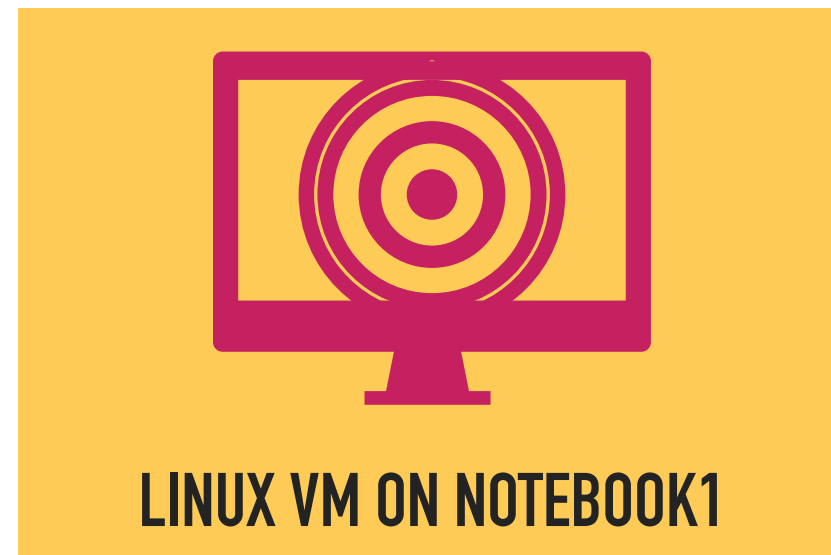


## GOALS FOR TODAY

2) Run nmap



1) Install a web server here



3) Defense: block packets

**Writing firewall rules is a challenge in a large environment.**  
**The rule of thumb is to first DROP,**  
**then ACCEPT only specific things in.**

## CREDITS

- ▶ Some of the slides and illustrations for this lecture are modified from
  - ▶ Stanford University Security Class
  - ▶ Syracuse University SEED Project
  - ▶ Kurose and Ross Computer Networking: A Top-Down Approach
  - ▶ MIT System Security Class