Network security

jedimaestro@asu.edu, CSE 365 Fall 2021

Outline

- Internet in a nutshell and the OSI model
  - Ethernet, ARP, IP, TCP, BGP, *etc.*
- Attacks in different layers
  - Off-path *vs.* in/on-path
- Firewalls and NIDSs
- VPNs
- Port scanning, SYN floods

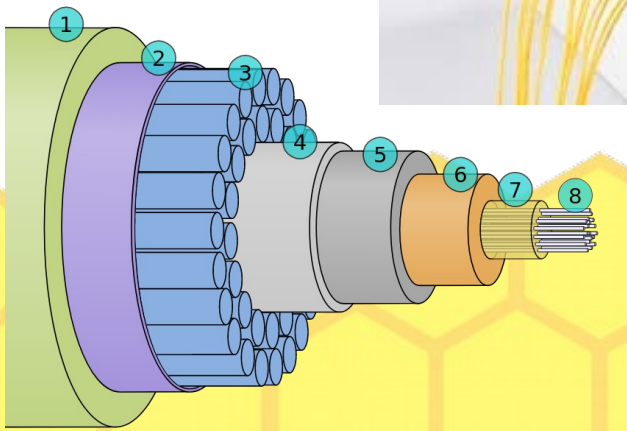
Some comments

- Bits matter
- Self reliance
  - Linux machine with root
- RTFTB doesn’t apply in this class, so really it’s RTFSC and RTFM
- These slides have a lot of info, consider it to be an overview and then use the homework as a focal point

Internet in a nutshell...

You want to connect two machines...

- Machines = desktops, laptops, mobile devices, routers, embedded devices, ...

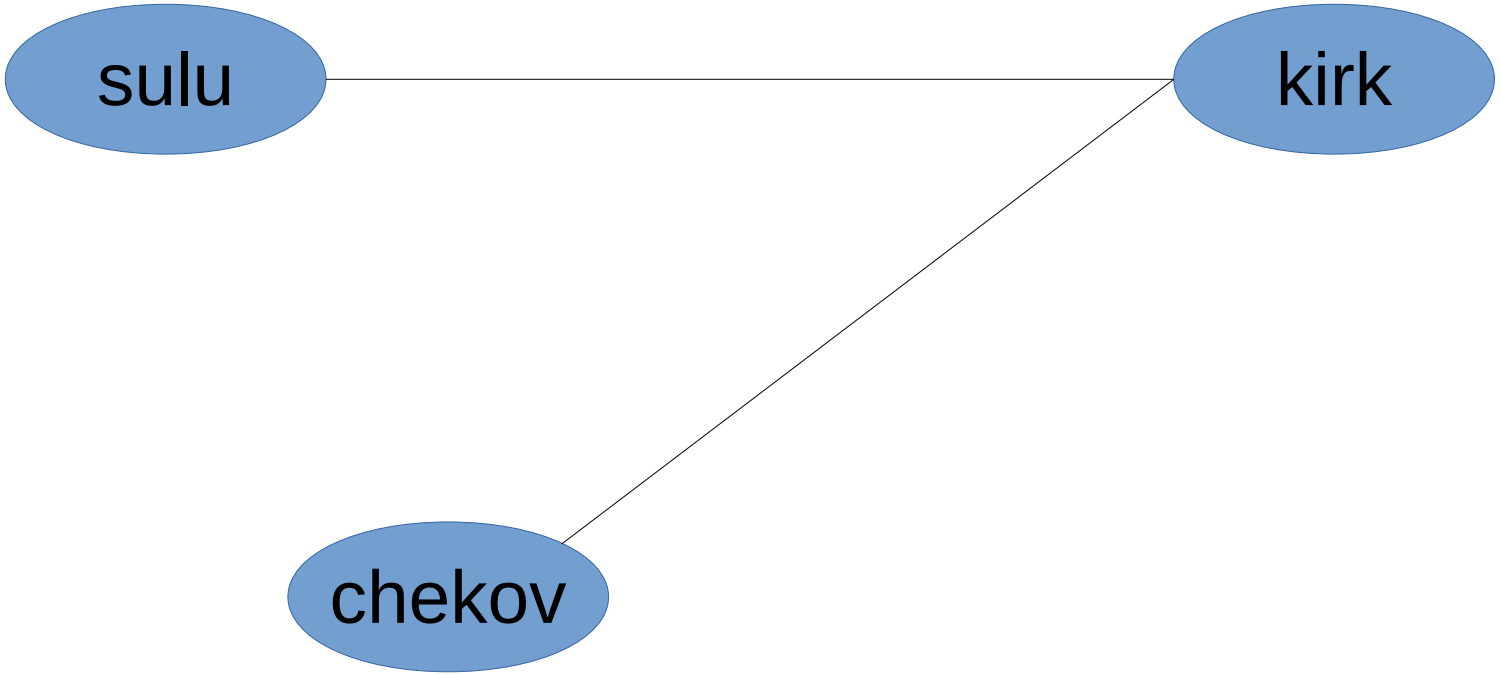


A “hop”

A “hop”

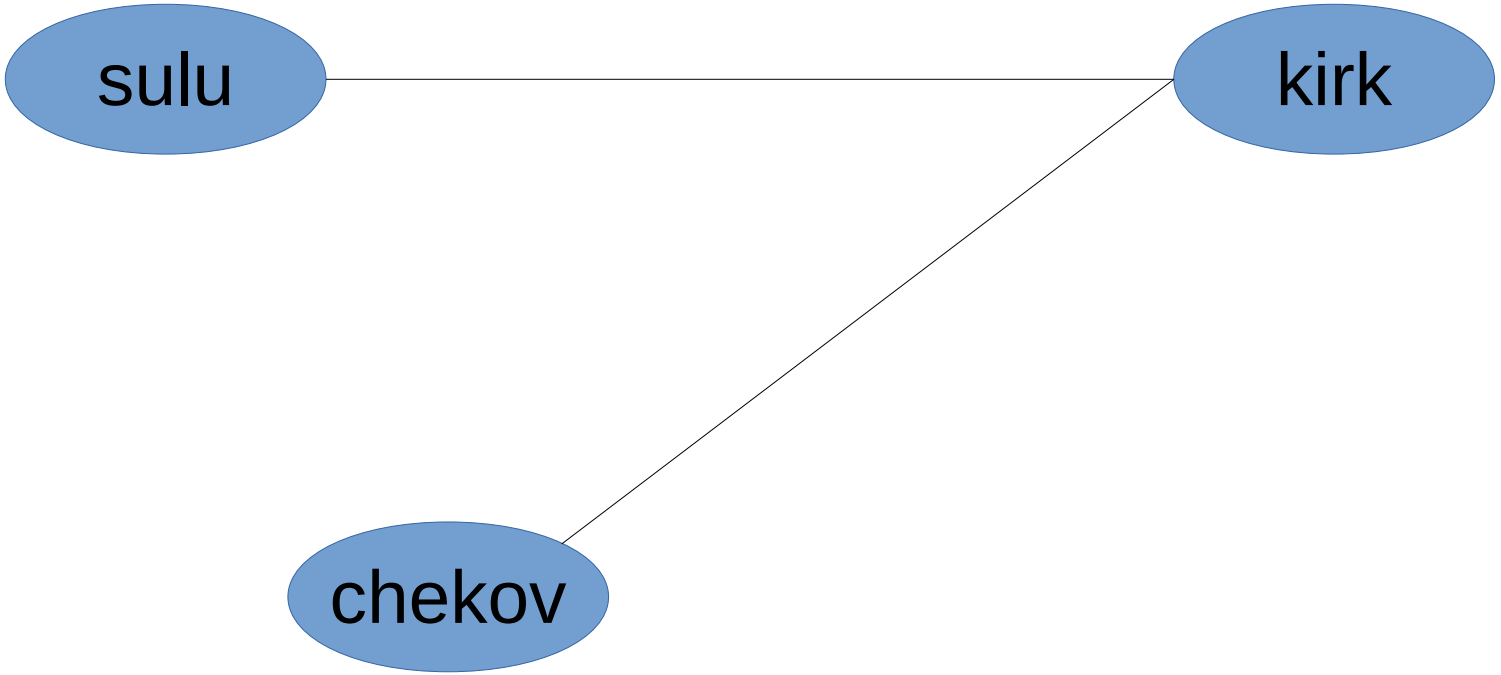
Ethernet

A “subnet”

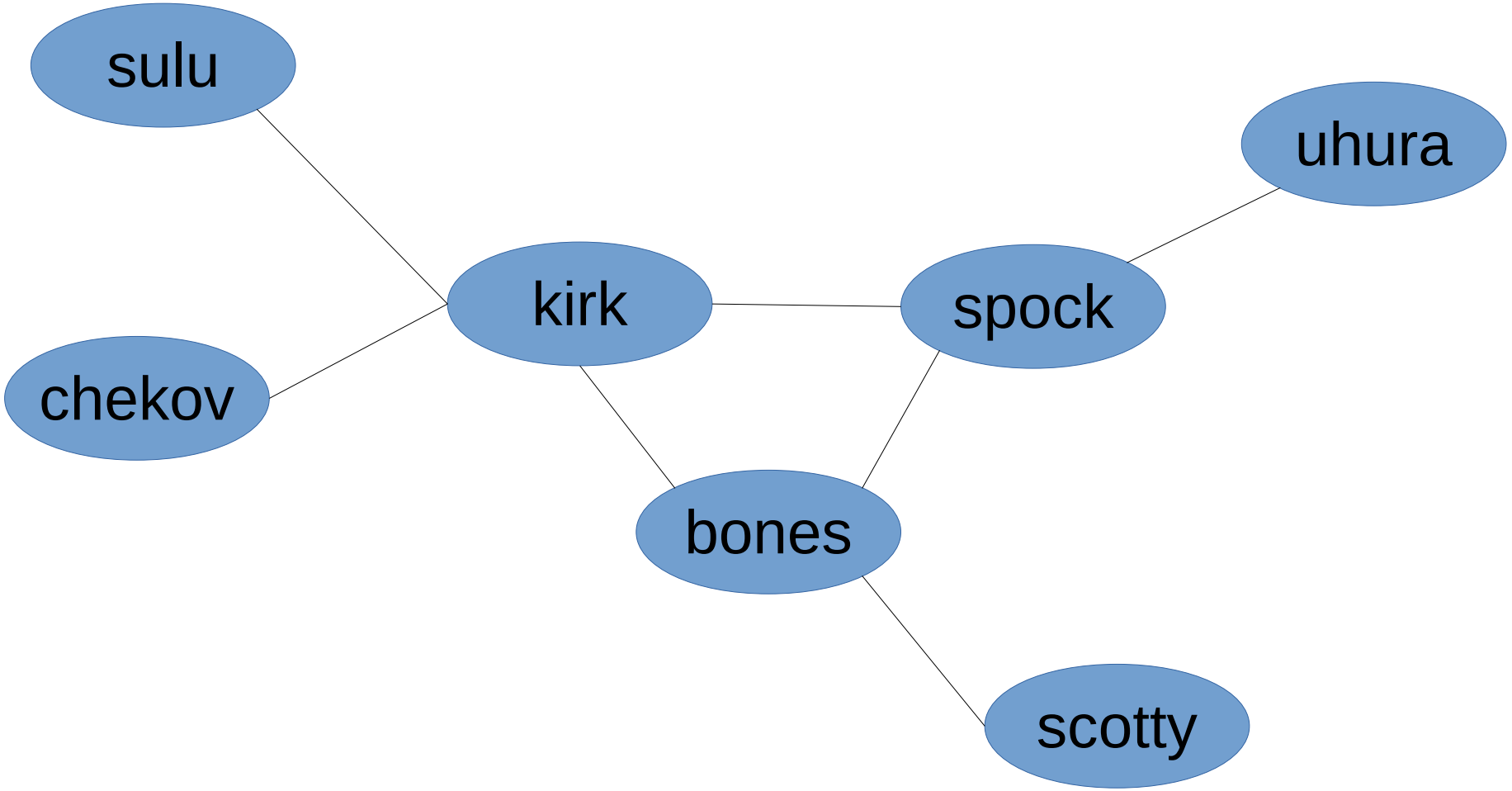


A “subnet”

ARP = Address Resolution Protocol



A network with routers



More terminology

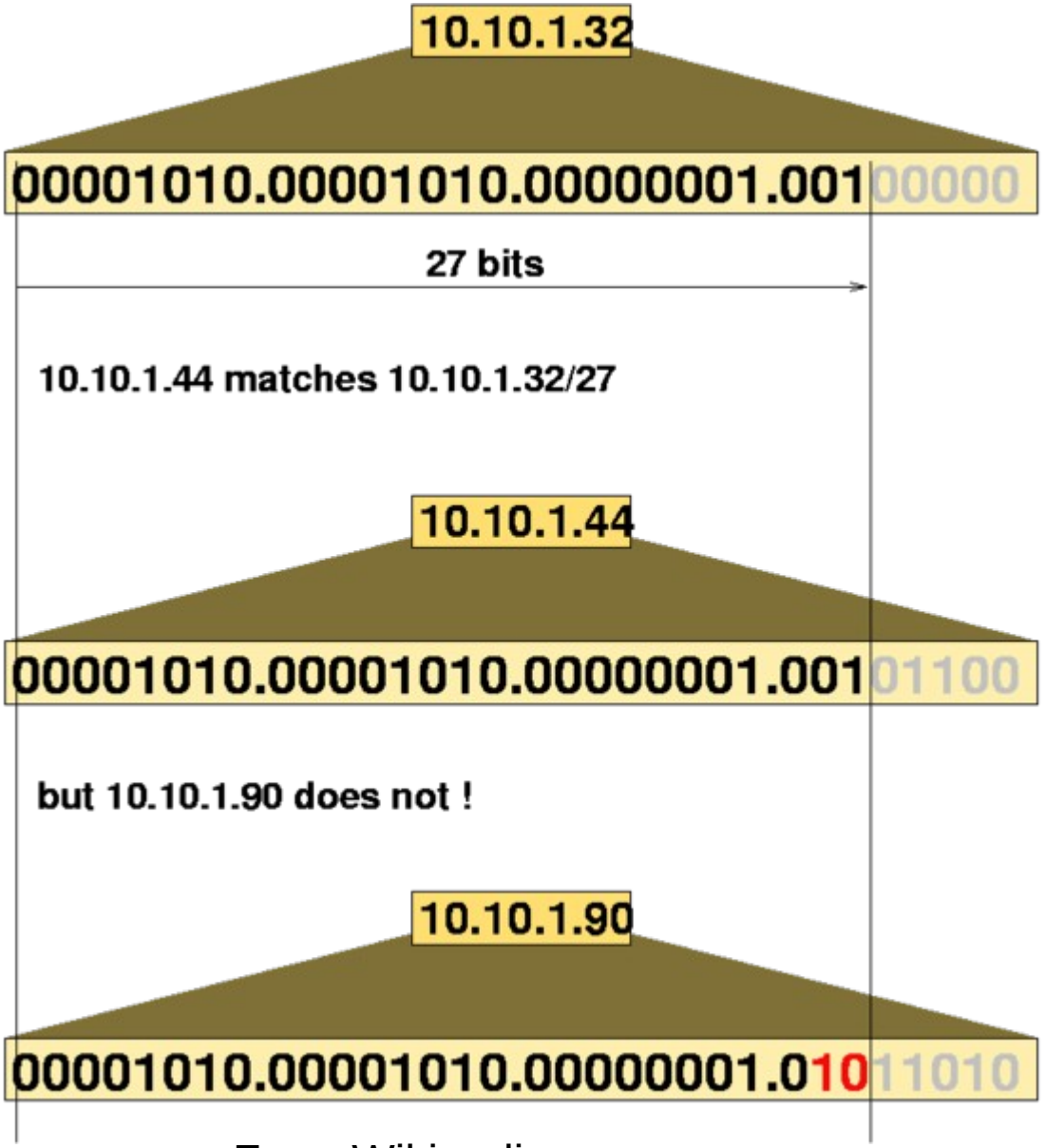
- IP = Internet protocol
- Forwarding, or “routing”
  - How packets get across the network
- Interface
  - WiFi, cellular, ...
- Path (or “route”), reverse path

IP address

- IPv4 is 32-bits, broken into 4 bytes
  - 192.168.7.8
  - 64.106.46.20
  - 8.8.8.8
- IPv6 is 128 bits
  - 2001:0db8:85a3:0000:0000:8a2e:0370:7334

CIDR

- Classless Inter-Domain Routing
- /27 has a net mask of 255.255.255.224



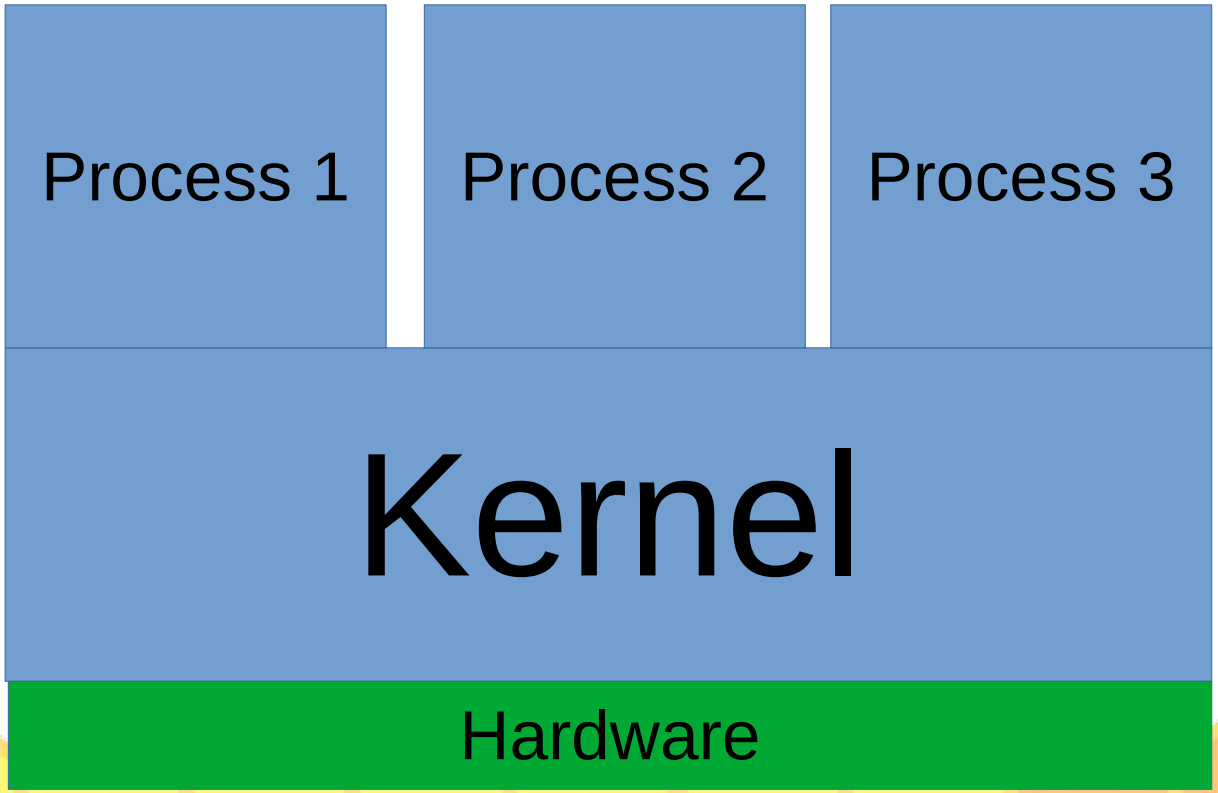
From Wikipedia

A connection

- For now, just know TCP, UDP, and ICMP
  - Stream sockets *vs.* datagrams
- TCP and UDP have “ports”
  - Port helps identify a process for incoming packets
  - Open port == “listening”
- Three-way handshake

Process?

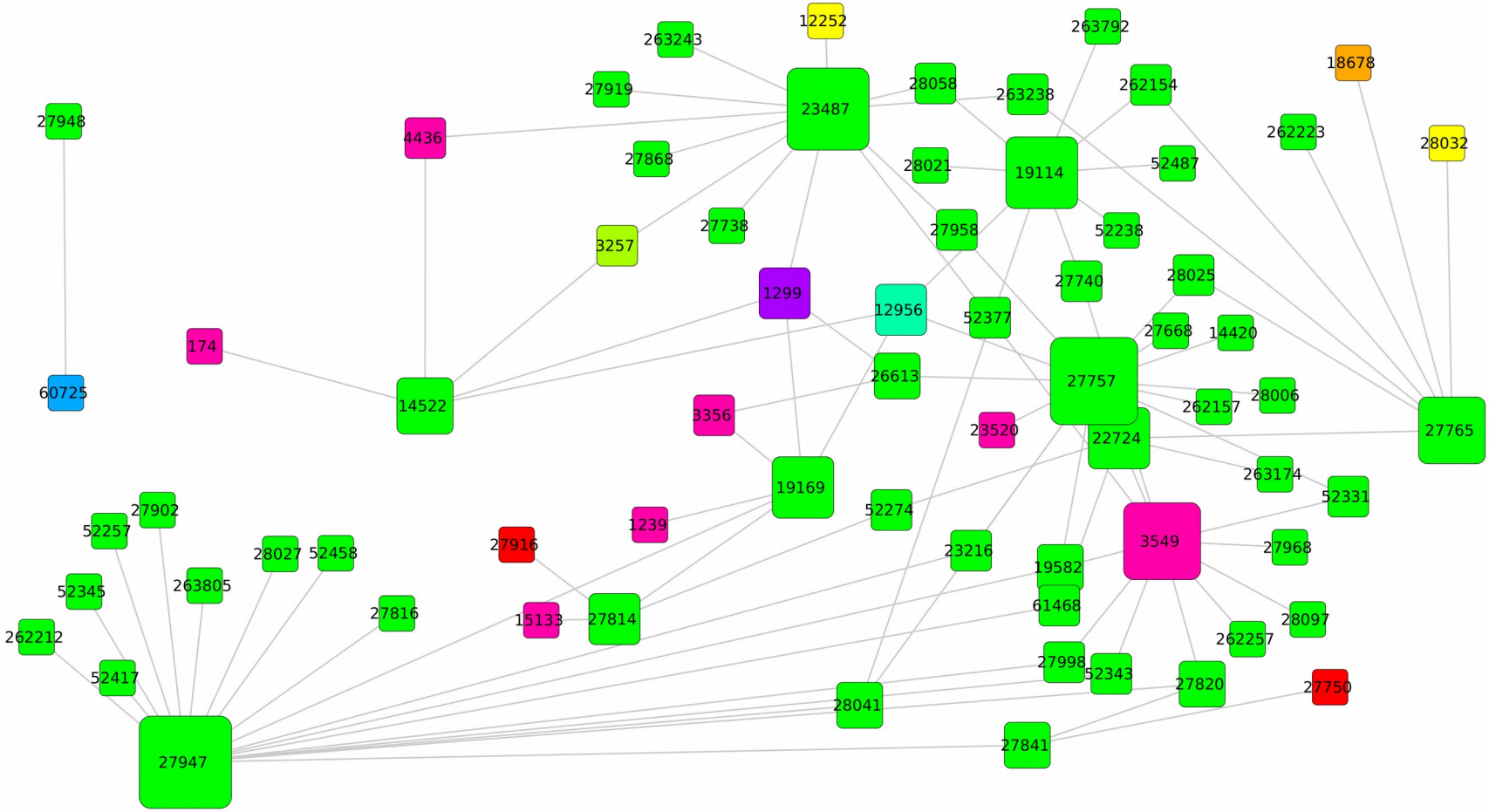
Separated by virtual memory, access system resources *via* system calls.

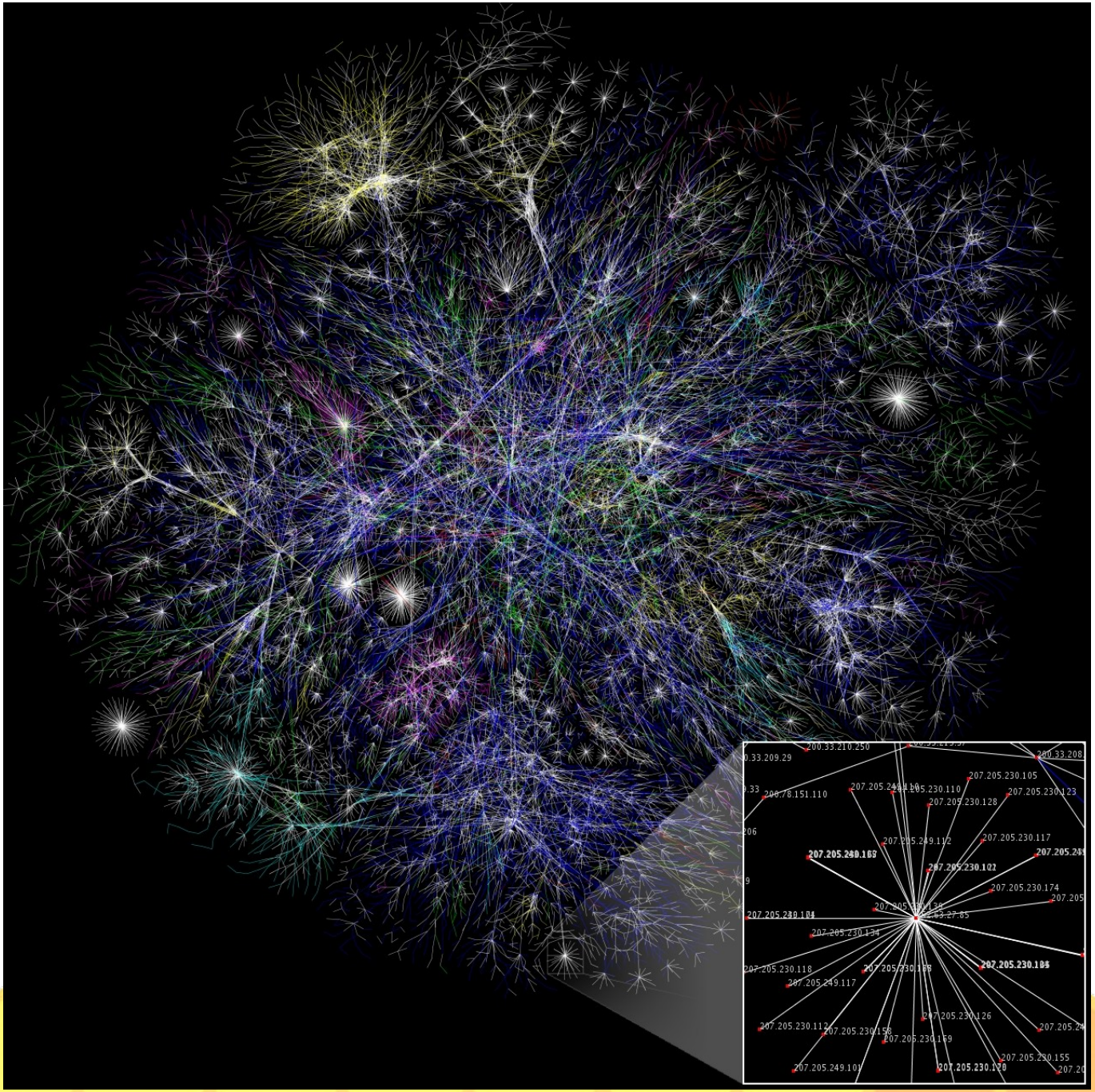


Almost there…

- DNS for resolving hostnames to IPs
  - breakpointingbad.com becomes 149.28.240.117
- BGP to scale to the size of the Internet
  - Path vector protocol
- HTTP as another example of an application layer protocol

Internet in Ecuador...




200.33.210.250
207.205.230.105
207.205.230.110
207.205.230.123
207.205.230.128
207.205.230.117
207.205.230.174
207.205.230.134
207.205.230.118
207.205.249.117
207.205.230.126
207.205.230.155

OSI model

- 1. Physical
- 2. Link
- 3. Network
- 4. Transport
- 5. Session
- 6. Presentation
- 7. Application

Attacks in different layers

Physical and link

- “Network adjacent”
- Can sniff (promiscuous mode)
- Can spoof
  - ARP cache poisoning
  - Goal is often to pretend to be the gateway

IP and transport layer

- Can spoof
- Can hijack

BGP or DNS

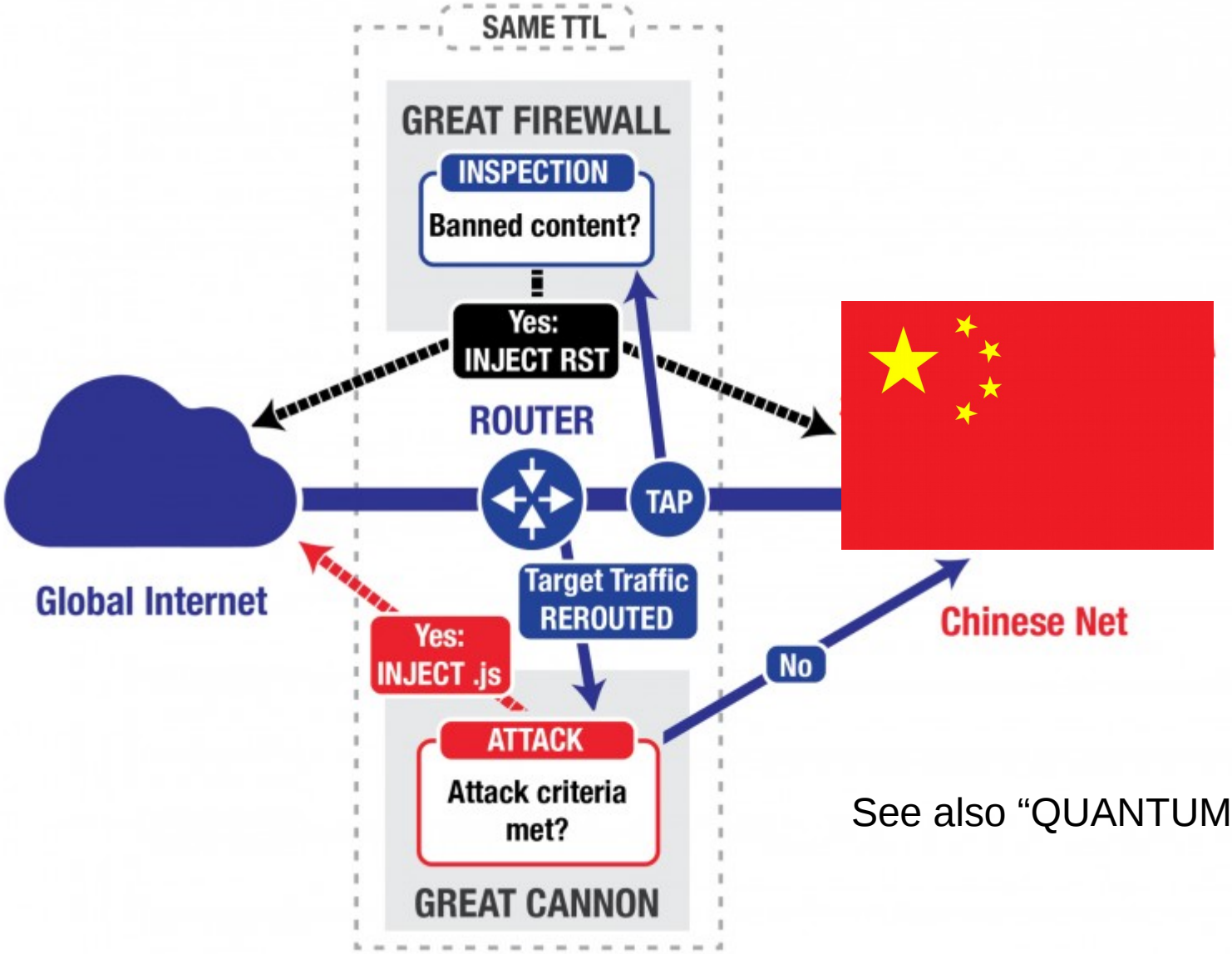
- Can spoof anything that doesn’t have crypto
- DNS cache poisoning
- BGP prefix attacks

Firewalls and NIDSs

Firewalls and NIDSs

- Basic idea is to sit in between two machines and apply some policy
- Firewall... “no packets enter my network with destination port 25”
- NIDS: Network Intrusion Detection System…. “Don’t allow TCP connections to send ‘%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3’”

https://citizenlab.ca/2015/04/chinas-great-cannon/



See also “QUANTUM Insert”

In- *vs.* On-path

- In-path … Attacker (or “security” device) gets to hold on to the packet and look at it, or modify it, before forwarding it
- On-path … Attacker (or “security” device) gets a copy, *via* something like a port mirror, but the packet has already been forwarded

Jed’s opinion: There is no firewall or NIDS that can’t be broken/evaded.

Ptacek and Newsham

- Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection
- Also see the work of Vern Paxson on “Bro” (now “Zeke”)
- The following is an example that uses IP fragments, all images from:

  https://www.sans.org/reading-room/whitepapers/detection/ip-fragment-reassembly-scapy-33969

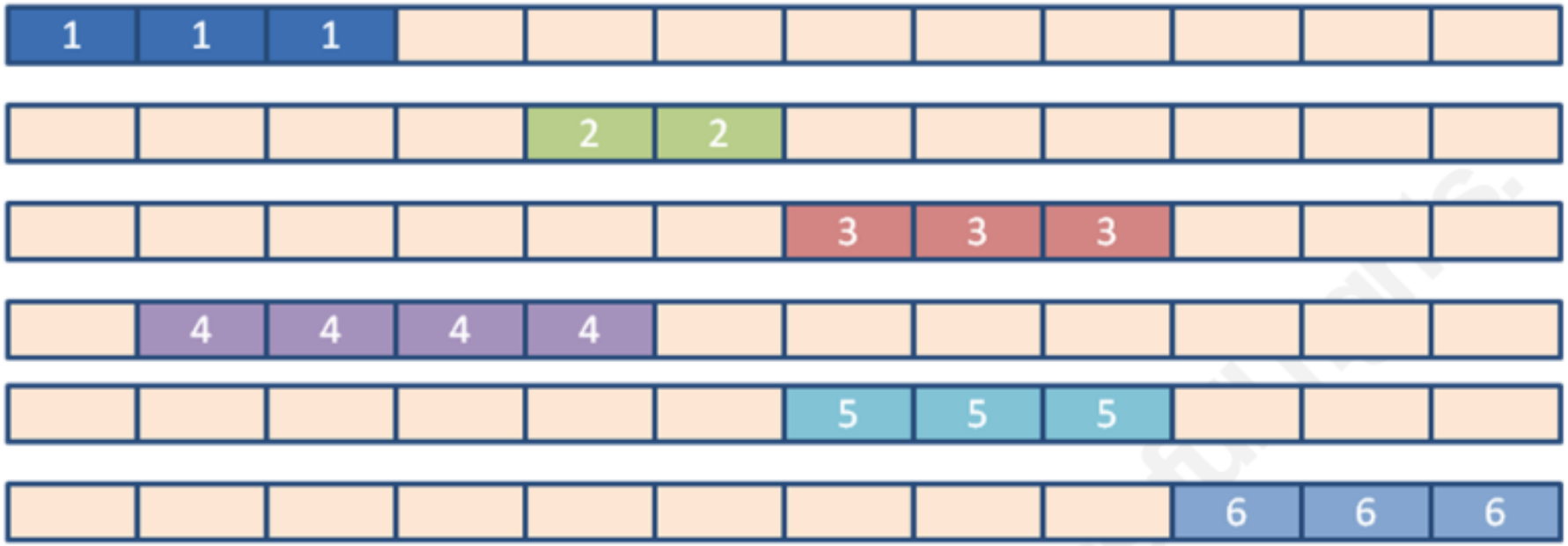


Figure 1: 6 Fragmented Packets (Shankar & Paxson, 2003)(Novak, 2005)

Reassembled using policy: First (Windows, SUN, MacOS, HPUX)

Reassembled using policy: Last/RFC791 (Cisco)

Reassembled using policy: Linux (Linux)

Reassembled using policy: BSD (AIX, FreeBSD, HPUX, VMS)

Reassembled using policy: BSD-Right (HP Jet Direct)

Figure 2: 5 Reassembly Methods (Shankar & Paxson, 2003)(Novak, 2005)

Step 1 - Attacker Crafts Linux and Windows Exploit fragments targeting a Windows host

ATTACKER VIEW

Windows Exploit

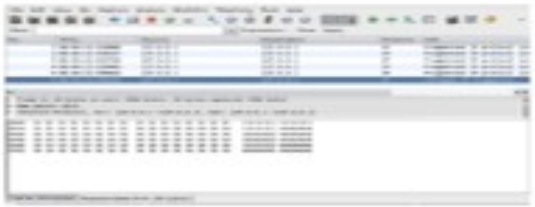
Linux Exploit

Step 2 - IDS correctly assembles packets as the target host would and alerts that the attack has occurred

IDS VIEW

Windows on Windows. successful attack ALERT!!!

Windows Exploit



Step 3 - Analyst Examines the full packet capture, sees a Linux exploit targeting Windows and dismisses the false positive

ANALYST VIEW

Linux on Windows failed attack. Stupid IDS. Next packet!!

Linux Exploit

Figure 3: Views of the attacker, IDS and analyst

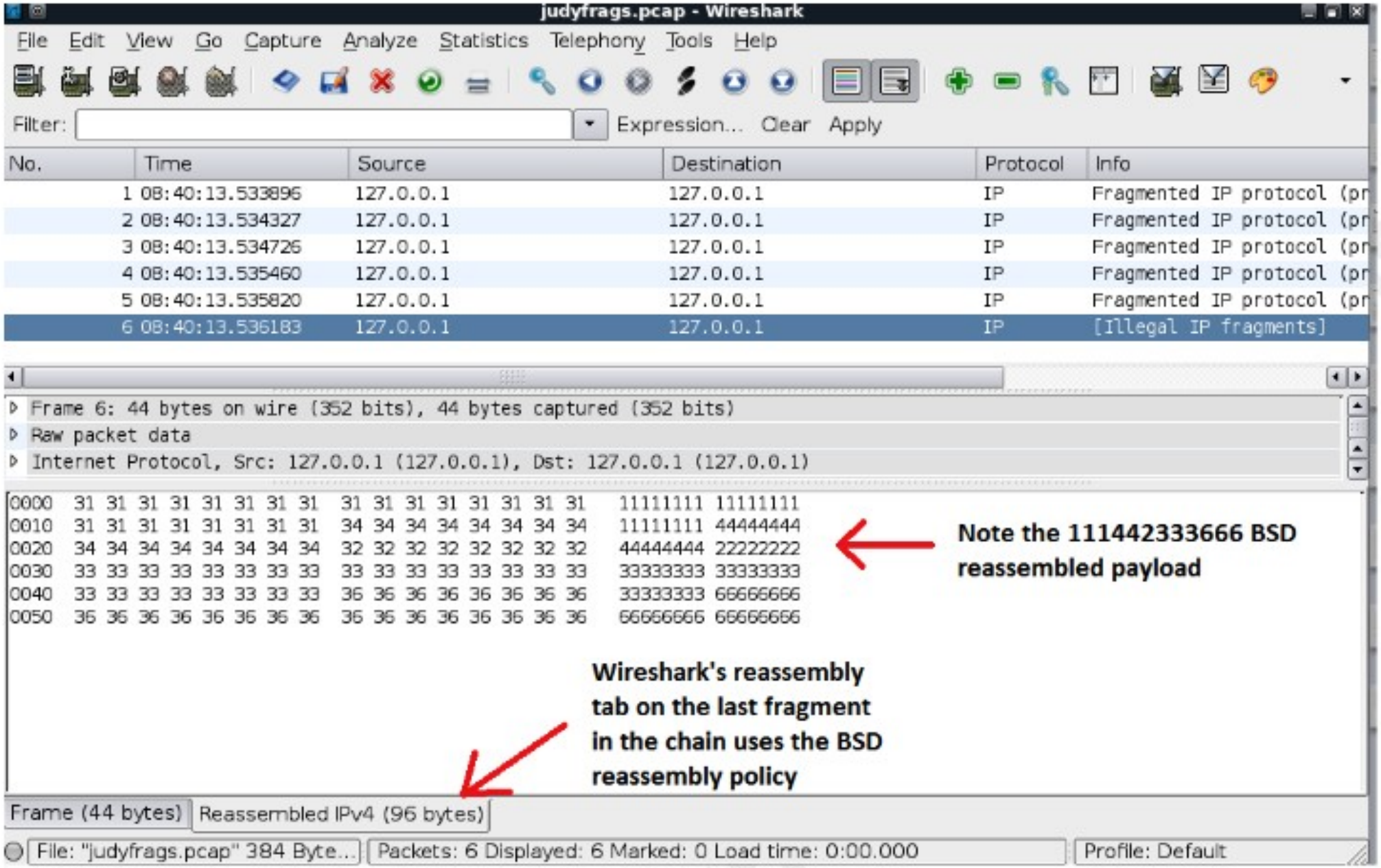
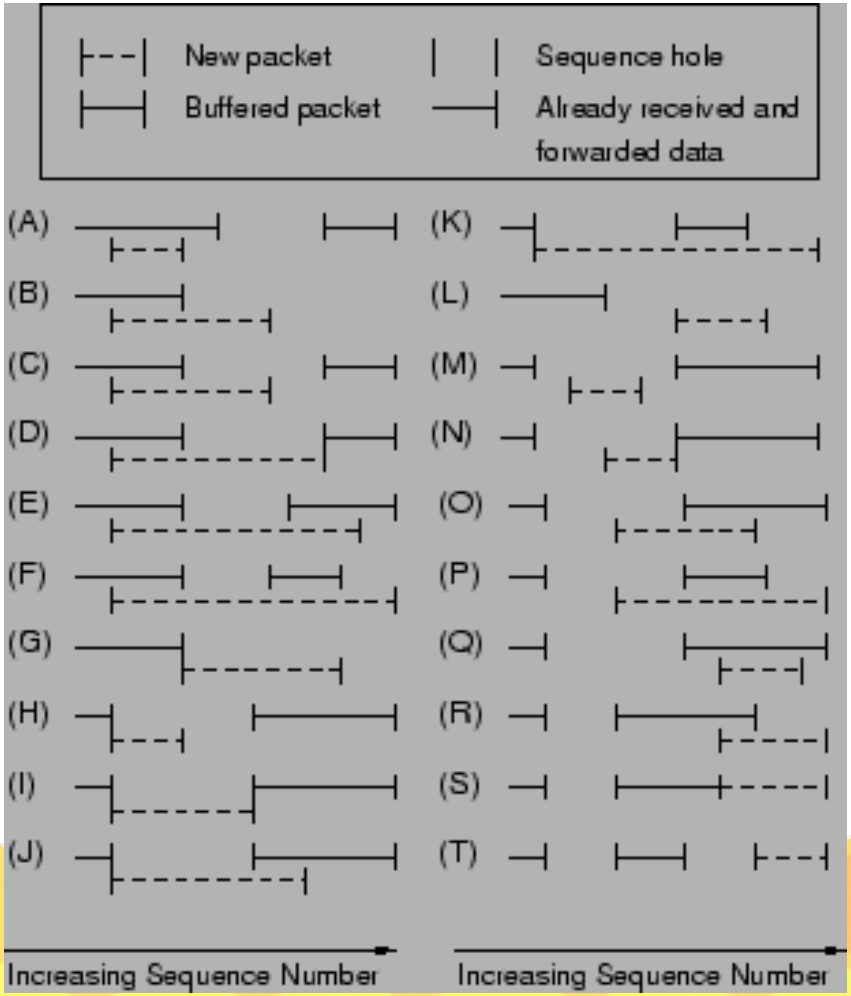


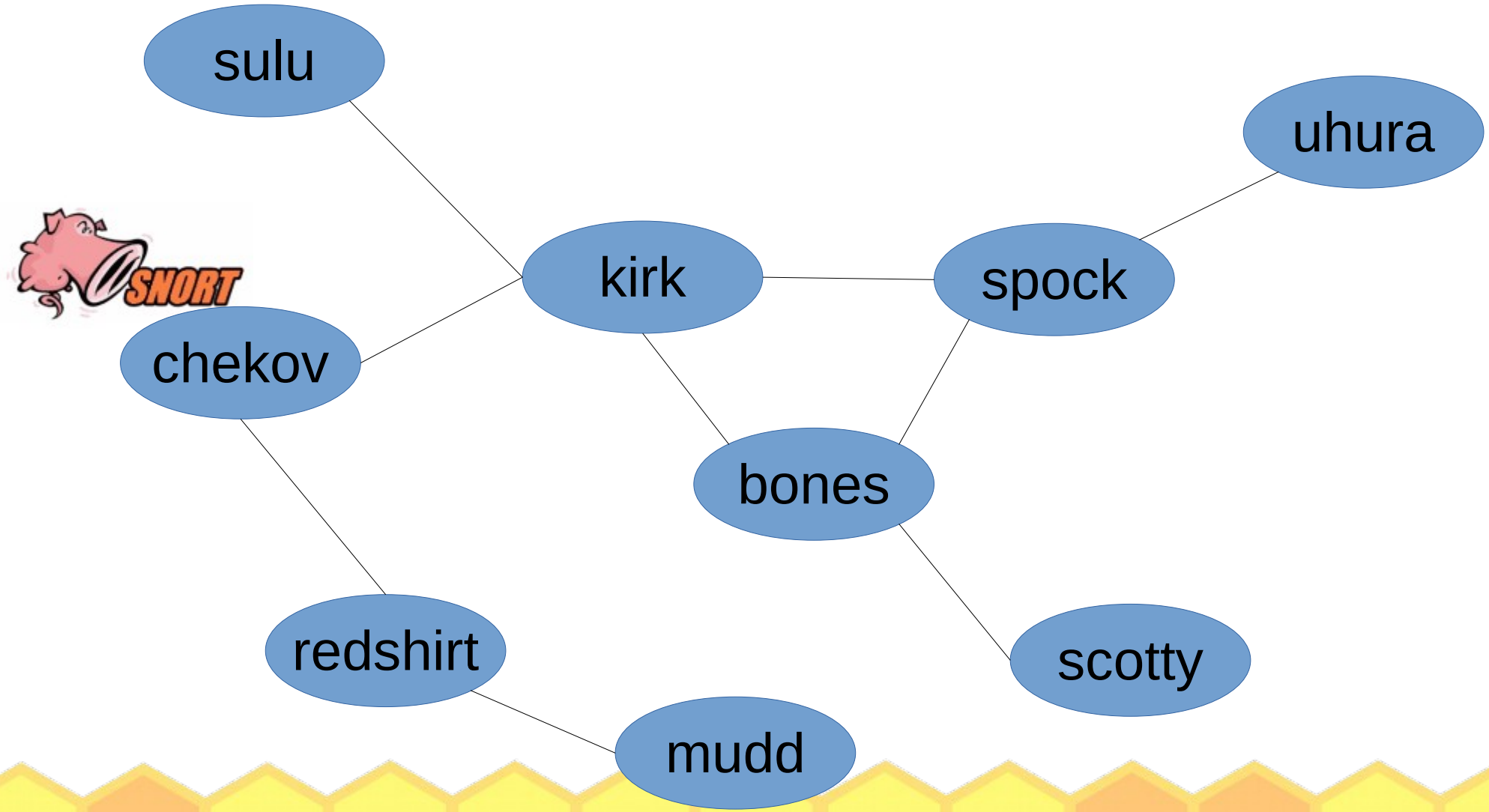
Figure 4: Wireshark uses BSD reassembly technique

TCP is even worse...

- http://www.icir.org/vern/papers/TcpReassembly/



TTL tricks



“Information only has meaning in that it is subject to interpretation”

–Computer Viruses, Theory and Experiments by Fred Cohen, 1984

“The only laws on the Internet are assembly and RFCs”

–Phrack 65 article by julia@winstonsmith.info

“Information is inherently physical”

--(*Lots of people said this, but see Richard Feynman's Lectures on Computation*)

OSI model

- 1. Physical
- 2. Link
- 3. Network
- 4. Transport
- 5. Session
- 6. Presentation
- 7. Application

A layer 7 example (XSS) due to Jeff Knockel

- Suppose “`<script>...</script>`” is blacklisted
- Use “`<script>...`” instead, many browsers will happily run the script anyway despite the missing closing tag
- Information only has meaning in that it is subject to interpretation
  - IDS interprets things one way, web browser another

Physical layer injection

- From https://www.usenix.org/legacy/events/woot11/tech/final_files/Goodspeed.pdf

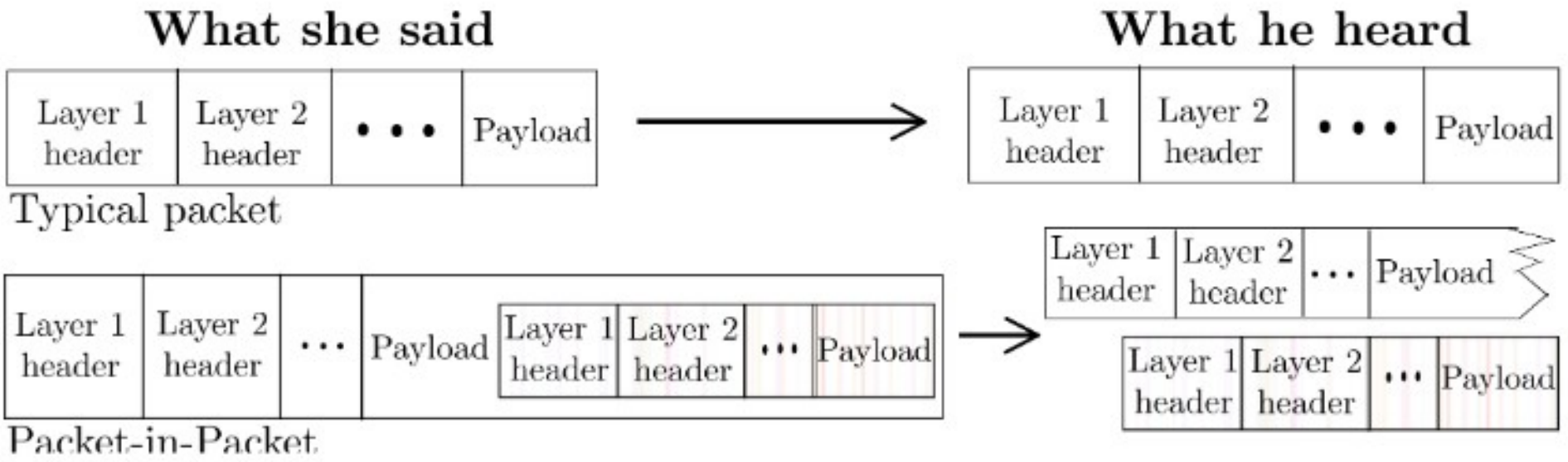
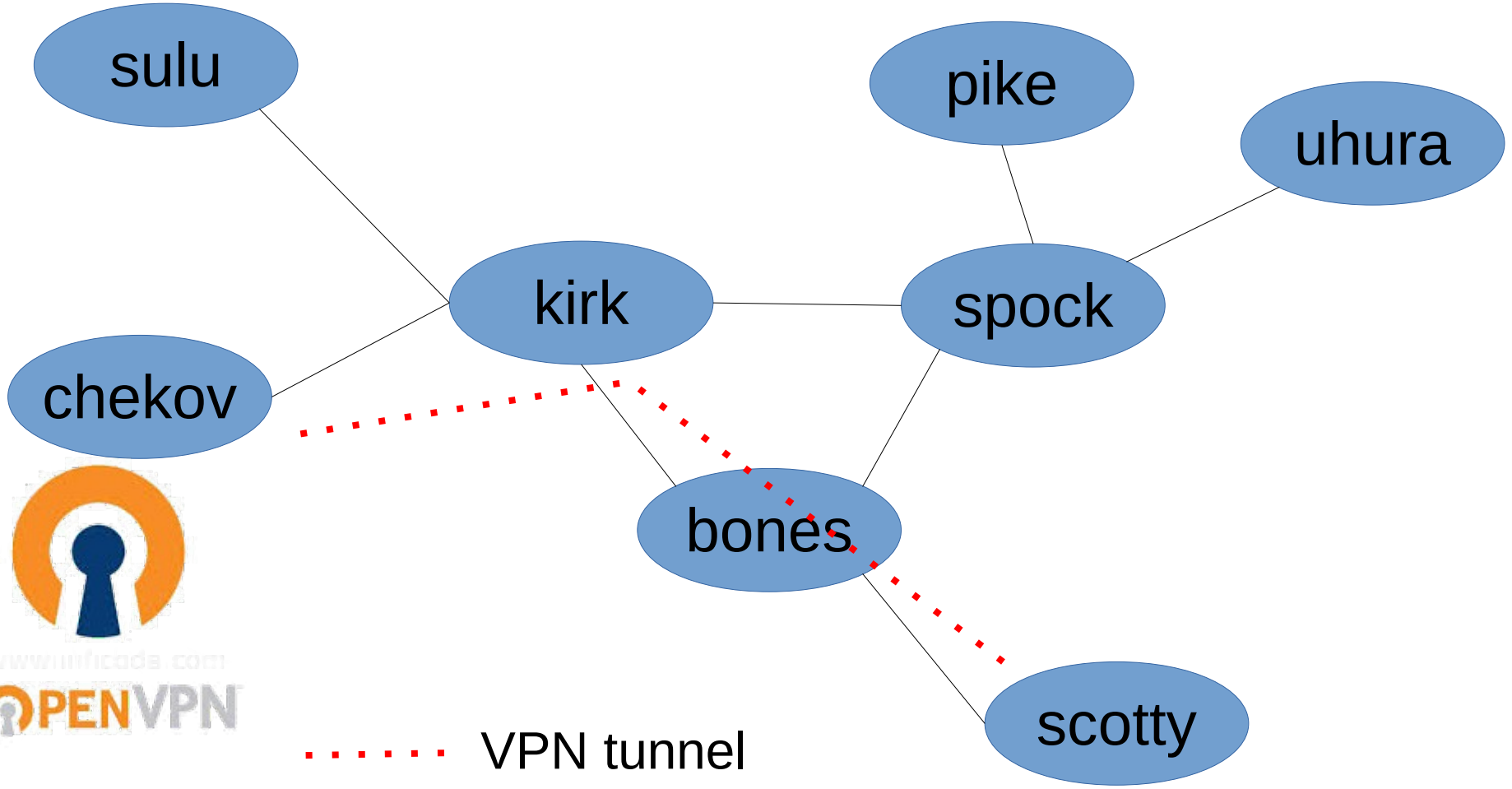


Figure 2: A typical packet's interpretation contrasted with that of a PIP.

Virtual Private Networks (VPNs)...

VPNs



Jed’s opinion… VPNs add very little, if anything, in terms of securing tunneled connections (unless you use them as originally intended).

Read the details if you’re interested

- breakpointingbad.com, find the blog
- Traditionally, a blind off-path attacker achieves things (like hijacking TCP or DNS) without even seeing the packets coming or going in a connection
- What about a blind in/on-path attacker?

Port scanning and SYN floods, coming soon…
(slides end here for now)