Basic Networking Concepts and Tools

Tony Espinoza

am.espinoza@utexas.edu

Networks

- What are some networks you are familiar with?
  - Local Area Network, home network.
  - Office network.
  - University network.

Networks

- Let's go into detail with a common network everyone uses every day.
- The Internet.
- What is the Internet?
  - On a basic level it is just a network of networks.

The Internet

- When going to a website how does your computer know where to go?
  - Type in the Uniform Resource Locator (URL) bar, e.g. google.com, utexas.edu...
- Your computer needs to translate that URL into something the network knows how to use.
  - Internet Protocol (IP) address.
  - utexas.edu $->$ 23.185.0.4

IP address

- Is a 32 bit number represented by a grouping of 4 octets.
  - 192.168.0.1
  - In hex: c0 a8 00 01

DNS[1] resolution

- How do domain names get resolved to IP addresses?
- i.e. How does my browser know how to take me to wikipedia.org
  - A query (IPv4)
  - AAAA query (IPv6)
- How to get IP address of wikipedia.org
  - `nslookup wikipedia.org`

[1] Domain Name System

nslookup output

```
> nslookup wikipedia.org

Server:         128.83.185.40
Address:        128.83.185.40#53

Non-authoritative answer:
Name:   wikipedia.org
Address: 208.80.153.224
Name:   wikipedia.org
Address: 2620:0:860:ed1a::1
```

Server: is the DNS server your computer is querying.

Address: is the DNS server and the port.

Why port 53?[2]

---

[2] Click

Your Local DNS server

For linux /etc/resolve.conf

```
> cat /etc/resolv.conf

# Generated by resolvconf
domain public.utexas.edu
nameserver 128.83.185.40
nameserver 128.83.185.41
```

Your Local DNS server

- How does your local DNS server know where to go?
- DNS is a distributed hierarchical database
  - Root DNS server
    - 13 labeled A-M
  - Top Level Domain (TLD) server
    - com, org, edu
  - Authoritative DNS server
    - amazon.com, pbs.org, utexas.edu

Example:

Let's look at wikipedia.org while recording a TCP dump which we will open with wireshark.

Tools:

- whois
  - Additional information about the IP address from the whois database
- dig
  - Similar to nslookup
- traceroute
  - Tries to find all the intermediary machines to a host
  - use with -T or -I and run as sudo
- nmap
  - -A Aggressive
  - -O OS detection

Tools:

- Zmap
  - Is a network tool for scanning the entire Internet (or large samples).
  - `wget http://64.106.81.7/blacklist.txt`
  - `sudo zmap --bandwidth=1M --target-port=80 --output-file=results.csv -b blacklist.txt`
- If we were to zmap ece.utexdas.edu how would we go about it?
  - Find out the range of IPs assigned to http://www.ece.utexas.edu/
    - dig or nslookup to get IP
  - Whois acquired IP to get the range of IP's in the network

RFC

- Request for Comments.
- Internet Engineering Task Force (IETF).
- Internet Research Task Force (IRTF).
- Internet Architecture Board (IAB).
- Independent authors.
- Engineers and computer scientists.

CIDR

- Classless Inter-Domain Routing.
- Notation for talking about ranges of IP address.
- Rare to see 192.168.0.0 - 192.168.0.255.
- Instead you would see 192.168.0.0/24.
- Equevalant to matching a netmask of 255.255.255.0.

CIDR

- Value after the / is called the prefix length.
- Number of address is
  - $2^{addressLength-prefixLength}$
- Prefix length is the number of leading 1's in the subnet netmask.

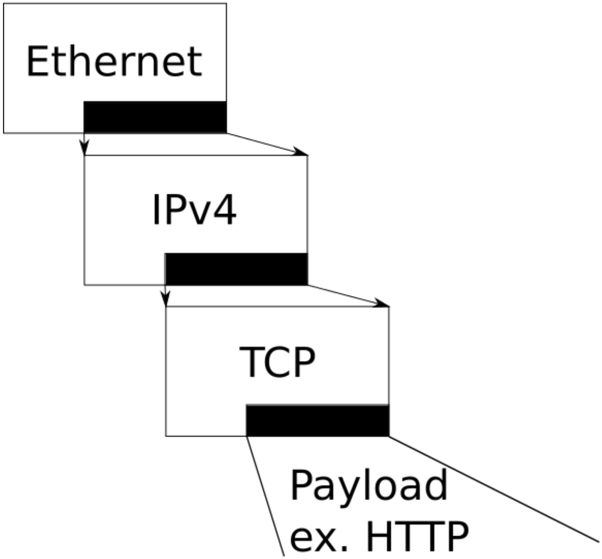
CIDR

- $0.0.0.0/8$ = Class A
- $0.0.0.0/16$ = Class B
- $0.0.0.0/24$ = Class C

CIDR

- $/29$
  - $32 - 29 = 3$
  - $2^3 = 8$
- $/32$
  - size of 1
- $/9$
  - $32 - 9 = 23$
  - $2^{23} = 8388608$

Packets



Ethernet

Preamble	Destination MAC address	Source MAC address	Type	User Data	Frame Check Sequence (FCS)
8	6	6	2	46 - 1500	4

Preamble:Ethernet hardware filters this field so it won't be visible in wireshark

FCS:Often missing from wireshark

IPv4

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Version|  IHL  |Type of Service|          Total Length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         Identification        |Flags|      Fragment Offset    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Time to Live |    Protocol   |         Header Checksum       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Source Address                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Destination Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IHL: Internet Header Length, number of 32-bit words.

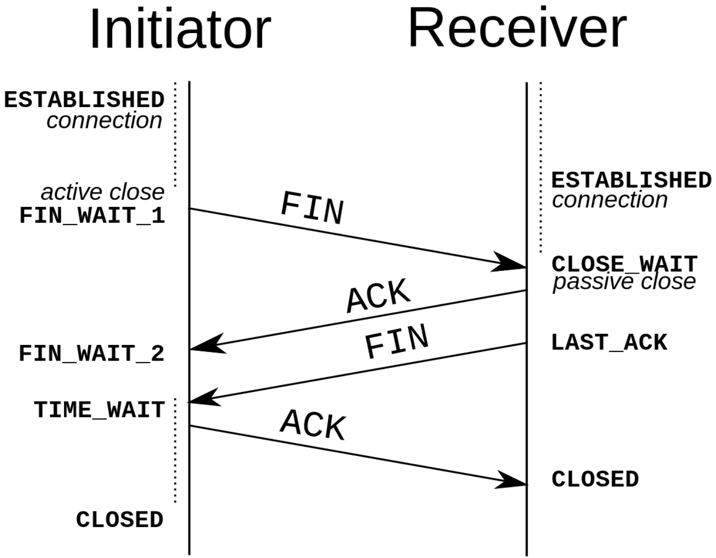
TCP

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Source Port          |       Destination Port        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Acknowledgment Number                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Data |       |C|E|U|A|P|R|S|F|                               |
| Offset|Resrvd |W|C|R|C|S|S|Y|I|            Window             |
|       |       |R|E|G|K|H|T|N|N|                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |         Urgent Pointer        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Options                    |    Padding    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Three way hand shake

- Client Sends SYN packet.
  - Client chooses a random sequence number.
- Server Sends SYN/ACK packet.
  - The acknowledgment number is set to one more than the received sequence number.
  - Server chooses a random sequence number.
- Client sends ACK packet.
  - The sequence number is set to the received acknowledgement value.
  - The acknowledgement number is set to one more than the received sequence number.

Terminate connection



But what if we don't finish the handshake?

We end up with a half open connection.

- What is a half open connection?
- Two ways to store half open connections.
  - TCP backlog.
    - size: `sysctl net.ipv4.tcp_max_syn_backlog`
  - SYN cookies.
    - Stateless, require no system resources.
    - Limited in entropy.
    - Stored in the sequence number.

SYN cookies

Return a special sequence number where they encode the following:

- Top 5 bits: t mod 32, where t is a 32-bit time counter that increases every 64 seconds;
- Next 3 bits: an encoding of an MSS selected by the server in response to the client's MSS;
- Bottom 24 bits: a server-selected secret function of the client IP address and port number, the server IP address and port number, and t.

Why SYN cookies

- Pro
  - Defend against DOS/DDOS attacks
  - Stays up when SYN cache is exhausted
- Con
  - Loss of entropy
  - Attacks that require the attacker to know the initial sequence number are easier to execute with a decress of entropy.
  - Attacks: blind RST, blind injection, blind connection.

Sequence and Acknowledgment number

- Reliable transmission of data.
  - If a packet is not received, the protocol retransmits the data.
- Other uses of sequence numbers?
  - Out of order packets.

Windows

- Each endpoint has a receive buffer size.
- There are many ways to send data...
  - However sending one packet at a time can be wasteful.
- Windows are solution.
  - The receiver has a window of packets for which it will accept sequence numbers.
  - The sender has a window as well..
- Two common methods to implementing windows.
  - Go-Back-N [3]
  - Selective Repeat Protocol(SRP) [4]

---

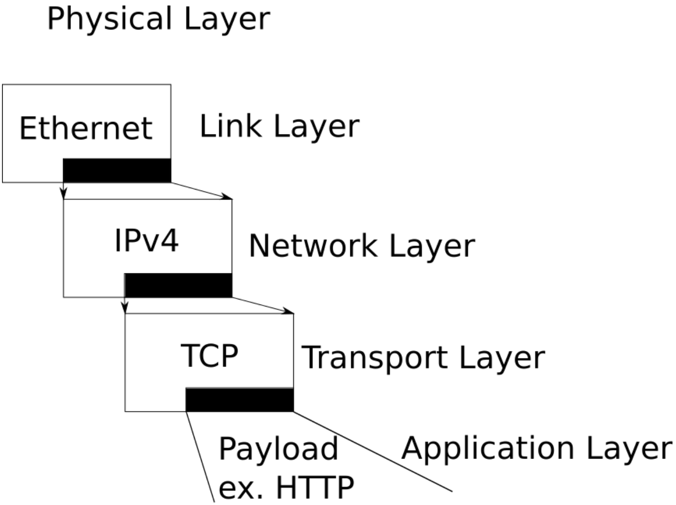
[3] Click the link

[4] Click the link

OSI stack

- Traditionally had 7 layers:
  - Application layer, presentation layer, session layer, transport layer, network layer, data link layer, and physical layer.
  - Antiquated as the OSI model was invented during the Internet's infancy.
- More common model is 5 layered.
  - Application
  - Transport
  - Network
  - Link
  - Physical

OSI stack


Physical Layer
Ethernet
Link Layer
IPv4
Network Layer
TCP
Transport Layer
Payload
ex. HTTP
Application Layer

Scapy

- Must use as sudo if you want to send packets.
- Can import the scapy library into python.
- Can use scapy to make send and receive packets.
- `IP()`
- `IP()/TCP()`
- `IP(dst="slashdot.org")/TCP()`
- `IP(dst="slashdot.org")/TCP(dport=80)`
- `IP(dst="slashdot.org")/TCP(dport=[80,443])`
- `z = IP(dst="slashdot.org")/TCP(dport=80)`
- r = sr(z)

Scapy

- `p = IP(dst="slashdot.org")/TCP(dport=80)`
- `p[1]` = TCP section
- In python `import scapy.all` give you everything but you need to use scapy.all.SCAPYFUNC
- `from scapy.all import IP, TCP, sr`
- use \ to compose e.g. `a = IP(dst="slashdot.org")/TCP(dport=80)/"GET / HTTP/1.0\r\n\r\n"`

Cryptography basics

- Symmetric encryption.
  - AES, twofish, serpent.
  - Public key exchange.
    - Diffie–Hellman.
- Asymmetric encryption.
  - RSA, named after the inventers Rivest,Shamir and Adleman.
- Hashing for integrity.
  - H-MAC.

Symmetric encryption

- Encrypt and decrypt with same key.
- Relatively fast.
- How to get both parties the key?
  - Key exchange
- AES
  - Block cypher

AES

1. KeyExpansion—round keys are derived from the cipher key using Rijndael's key schedule. AES requires a separate 128-bit round key block for each round plus one more.
2. Initial round key addition: AddRoundKey—each byte of the state is combined with a block of the round key using bitwise xor.

AES

3. 9, 11 or 13 rounds: (key size dependant)
   3.1 SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
   3.2 ShiftRows—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
   3.3 MixColumns—a linear mixing operation which operates on the columns of the state, combining the four bytes in each column.
   3.4 AddRoundKey
4. Final round (making 10, 12 or 14 rounds in total):
   4.1 SubBytes
   4.2 ShiftRows
   4.3 AddRoundKey

Key exchange

- Diffie-Hellman key exchange.
  - Allows two parties that have no prior knowledge of each other to establish a shared secret key over an insecure channel.
  - Uses a multiplicative group of integers modulo a prime p.
- No authentication, possible MITM.
- Provides forward secrecy.
  - Protects past sessions against future compromises of secret keys.

Diffie-Hellman

1. Alice and Bob publicly agree to use a modulus p = 6700417 and base g = 4095 (which is a primitive root modulo p).
2. Alice chooses a secret integer a = 90, then sends Bob $A = g^a (mod) p$. $A = 4095^{90} (mod) 6700417 = 4081248$
3. Bob chooses a secret integer b = 50, then sends Alice $B = g^b (mod) p$. $B = 4095^{50} (mod) 6700417 = 4251305$
4. Alice computes $s = B^a (mod) p$ $s = 4251305^{90} (mod) p = 608102$
5. Bob computes $s = A^b (mod) p$ $s = 4081248^{50} (mod) p = 608102$
6. Alice and Bob now share a secret (the number 608102).

Diffie-Hellman

There is another form of DH key exchange know as elliptic curve Diffie-Hellman ECDH. ECDH uses a multiplicative group of points on an elliptic curve.

Here is a link to a great article that describes in detail how elliptic curves work.

Same idea as regular DH in the sense that you are creating a shared secret on an insecure channel.

Asymmetric encryption

- Public and private key pairs.
- Slower than symmetric systems.
- RSA
  - Relies on the difficulty of factoring large numbers.
- How to share keys?
  - Public Key Infrastructure (PKI)

Message Authentication Codes (MAC)

- MAC are used to detect a messages integrity.
  - Verify the message is from the correct person, and has not been changed.
- HMAC
  - $h(K \oplus a || h(K \oplus b || m))$

$h(K \oplus a || h(K \oplus b || m))$

- K is a key padded with 0's
- h is a cryptographic hash function
- m is the message to be authenticated
- $||$ denotes concatenation
- $\oplus$ denotes bitwise exclusive or (XOR)
- a is the block-sized outer padding, consisting of repeated bytes valued 0x5c
- b is the block-sized inner padding, consisting of repeated bytes valued 0x36

Servers

- How do you know you are talking to the correct server?
- With what we learned what could we do as adversaries?
- Does the TCP checksum help?
- How can we be sure that the communication with the server is private?

Encryption

- What issues does encryption solve?
- What issues still exist?

Verification

- How to verify a server is who they say they are?
  - A trusted third party.
    - IdenTrust, Comodo, DigiCert
  - Certificate Authorities(CA).
  - X.509 protocol.
  - Check out a certificate in Firefox.

Transport Layer Security [5]

- Probably the Internet's most important security protocol
- Designed over 20 years ago by Netscape for Web transactions
- Back then, called Secure Sockets Layer
- But used for just about everything you can think of
  - HTTP
  - SSL-VPNs
  - E-mail
  - Voice/video
  - IoT

[5] Heavily lifted from Eric Rescorla

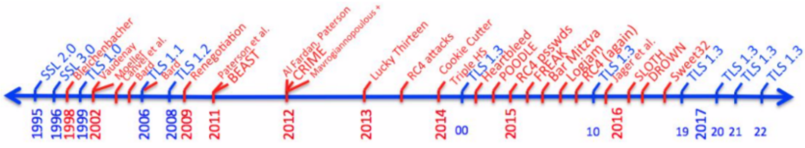
TLS attacks

NETSCAPE

SSL 2.0 (1995) → SSL 3.0 (1996)

I E T F

TLS 1.0 (1999) → TLS 1.1 (2006) → TLS 1.2 (2008)



*Slide from van der Merwe and Paterson

TLS Structure

- Handshake protocol
  - Establish shared keys (typically using public key cryptography)
  - Negotiate algorithms, modes, parameters
  - Authenticate one or both sides
- Record protocol
  - Carry individual messages
  - Protected under symmetric keys

mitmproxy

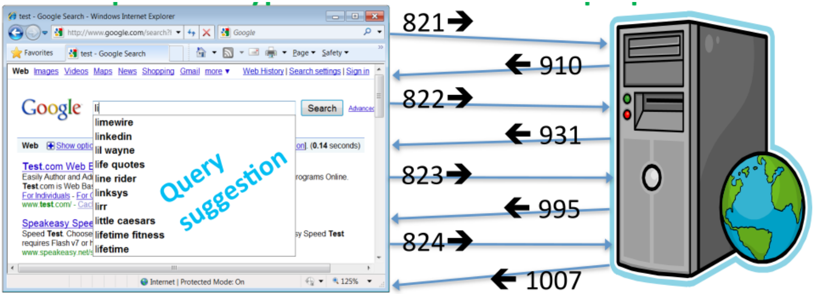
- Off the shelf tool to preform a man in the middle attack
- Can intercept your own https traffic.
- MUST download certificate. - mitmproxy generages unique certs for every install.
- Configure network settings of your browser to use a manual proxy - 127.0.0.1 - port 8080 - check `use this proxy server for all protocols`

Side channel attacks [6]

- Surprisingly detailed user information is being leaked out from several high-profile web applications
  - personal health data, family income, investment details, search queries
- The root causes are some fundamental characteristics in today's web apps
  - stateful communication, low entropy input and significant traffic distinctions.

[6] Side-channel-leaks in Web Applica2ons: A Reality today, A Challenge Tomorrow

Side channel attacks



Side channel attacks

- Similar methods can deanonymize other types of traffic as well.
  - Investment information.
    - Each price history curve is a GIF image from MarketWatch
  - Medical information.
    - Similar to search example
  - Tax filing web sites.

Anonymity

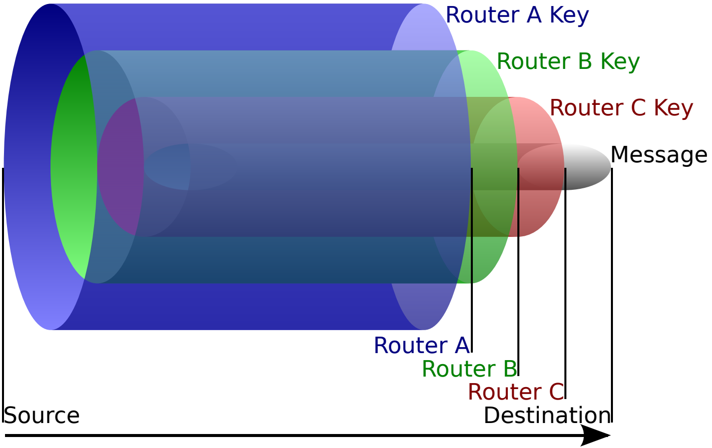
What tools do people use to try to be anonymous?

- VPN
  - Uses?
  - Trust model.
  - DNS Leak.
- TOR
  - Onion routing.

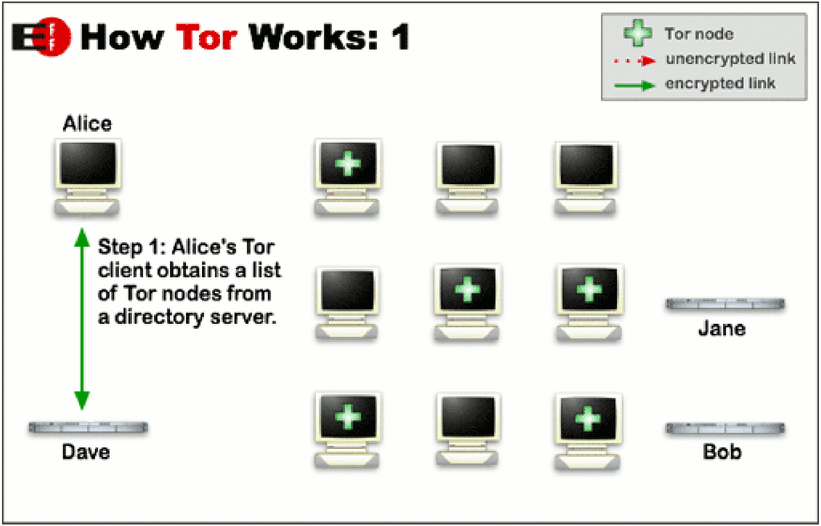
What is TOR

- Online anonymity
  1. Software
  2. Network
  3. Protocol
- Open source, freely available
- Community of researchers, developers, users, and relay operators
- Funding from US DoD, Electronic Frontier Foundation, Voice of America, Google, NLnet, Human Rights Watch

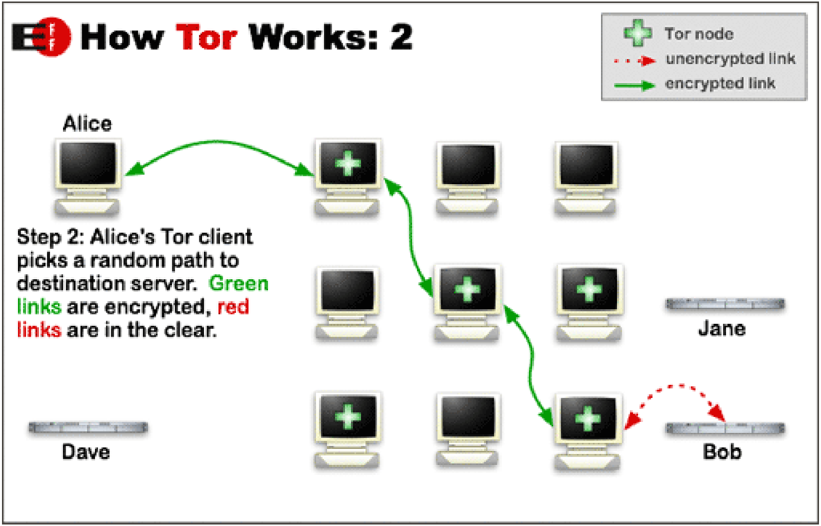
Onion Routing



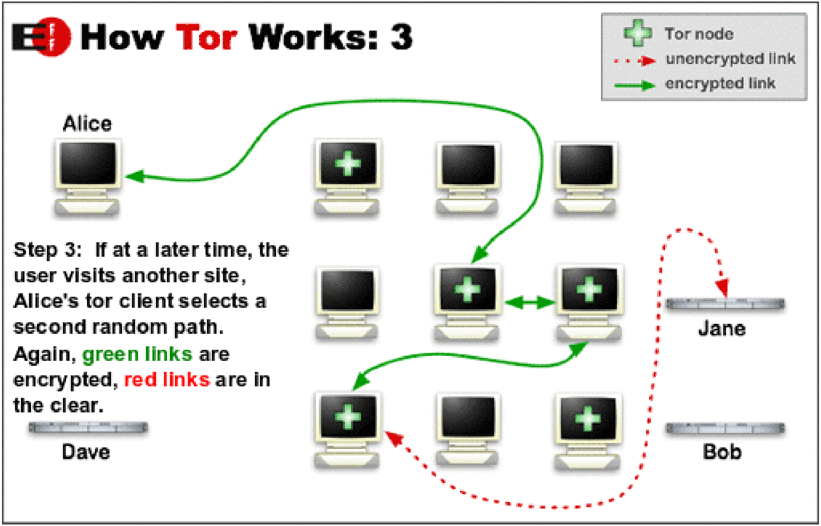
TOR


How Tor Works: 1
Tor node
unencrypted link
encrypted link
Alice
Step 1: Alice's Tor client obtains a list of Tor nodes from a directory server.
Jane
Dave
Bob

TOR


How Tor Works: 2
Tor node
unencrypted link
encrypted link
Alice
Step 2: Alice's Tor client picks a random path to destination server. Green links are encrypted, red links are in the clear.
Jane
Dave
Bob

TOR



How Tor Works: 3
Tor node
unencrypted link
encrypted link
Alice
Step 3: If at a later time, the user visits another site, Alice's tor client selects a second random path. Again, green links are encrypted, red links are in the clear.
Jane
Dave
Bob

TOR

Attackers can block users from connecting to the Tor network:

- By blocking the directory authorities
- By blocking all the relay IP addresses in the directory
- By filtering based on Tor's network fingerprint
- By preventing users from finding the Tor software

TOR

- For places that block by IP.
- Request a bridge.
- A relay not listed in the main directory.
  - Some countries blacklist all IPs in the main directory.

TOR

- For places that block by traffic shape:
  - Plugable transports are the solution.
  - Shape traffic so that it looks like something else.
  - Skype, meek, obs4...

Bro/zeek

- Bro is being re-named as zeek.
- Bro is a passive, open-source network traffic analyzer.
- It is primarily a security monitor that inspects all traffic on a link in depth for signs of suspicious activity.
- Can be used as an IDS
- Originally developed by Vern Paxson to detect network intruders in real time.

Zeek

- Captures packets.
- Runs through an event engine which accepts or rejects.
- Forwards accepted events to policy script interpreter.

Zeek

- Events handled by policy scripts.
- Scripts are written in zeek's scripting language.

Zeek

```
#Create a new event handler "file_new"
#When Bro finds a file being transferred
#(via any protocol it knows about),
# write a basic message to stdout and then
#tell Bro to save the file to disk.
event file_new( f: fa_file)
{
local fuid = f$id;
local fsource = f$source;
local ftype = f$mime_type;
local fname = fmt(" extract-%s-%s", fsource, fuid);
print fmt("*** Found %s in %s. Saved as %s. File ID is %s", ftype,
fsource, fname, fuid);
Files:: add_analyzer(f, Files:: ANALYZER_EXTRACT,
[$ extract_filename = fname]);
}
```

Zeek

- Can be run on the command line:
- `sudo bro -i enp0s3`
- Where enp0s3 is your networking interface.
- Creates log files in the directory it is run from.

Snort

- IDS
- Intrusion Prevention System (IPS)
- Real time packet analysis, and packet logging
- Can also be used to detect probes or attacks,
  - Such as, operating system fingerprinting attempts, semantic URL attacks, buffer overflows, server message block probes, and stealth port scans

HTTP protocol

HyperText Transfer Protocol

- A request response protocol in the client server computing model.
- GET is the main request (eg. retrieve the contents of a webpage).
- For a list of the other requests see the RFC

HTTP protocol

A response has the following structure:

- a status line which includes the status code and reason message.
- response header fields (e.g., Content-Type: text/html)
- an empty line
- an optional message body