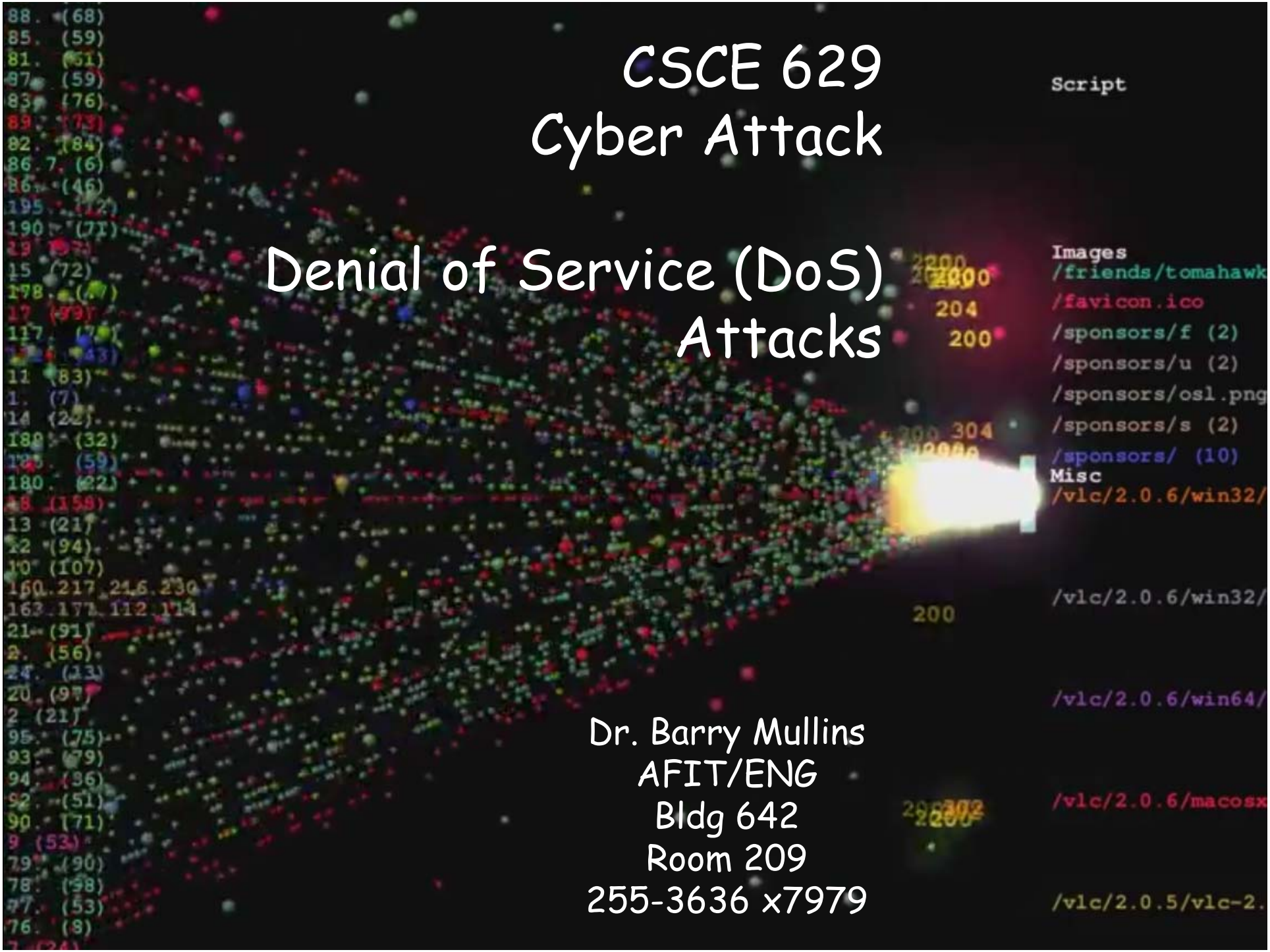


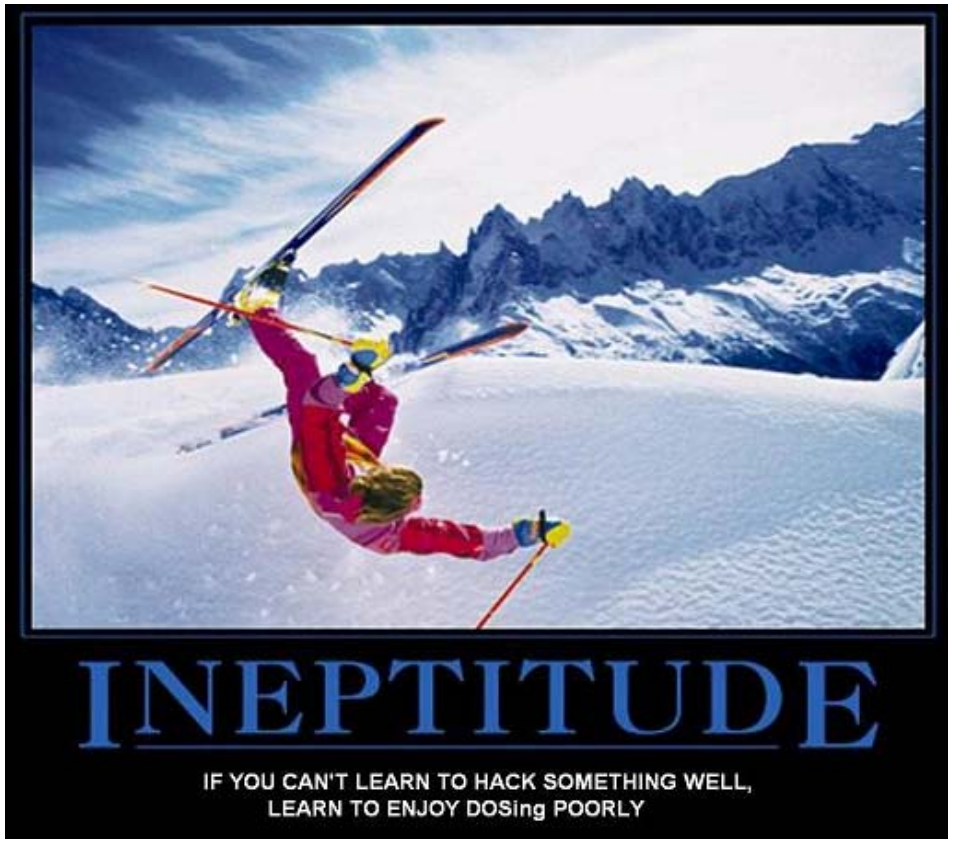
CSCE 629
Cyber Attack

Denial of Service (DoS) Attacks

Dr. Barry Mullins
AFIT/ENG
Bldg 642
Room 209
255-3636 x7979

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding



Denial of Service Attacks

- Often used to prevent access to legitimate users or stop system processes
- A DoS attack could be
  - flooding a machine with requests thereby effectively shutting down the operating system
  - saturating the pipe (bandwidth) to the target so legit traffic is blocked
- Generally, DoS attacks are not technically elegant
  - But could be used as an integral part of an elaborate attack
- You cannot totally eliminate the possibility of DoS attacks

http://nakedsecurity.sophos.com/2014/08/18/shark-attack-google-wraps-underwater-cables-in-kevlar-like-vests/

Categories of Denial-of-Service Attacks

	Stopping Services	Exhausting Resources
Locally	❒ Process killing – kill service as root ❒ Process crashing – exploit service (buffer overflow)	❒ Spawning processes to fill the process table ❒ Filling up the file system ❒ Sending bogus outbound traffic to saturate link
Remotely (across the network)	❒ Malformed packet attacks ❒ ARP cache poisoning ❒ RESET TCP connections	❒ Packet floods (e.g., SYN Flood, Smurf, DDoS, etc.) ❒ Most popular technique

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding

Malformed Packet DoS Attacks

- Most common method of remotely stopping a service
- Malformed IP header (header length = 2 and version = 3)
  - **`send(IP(dst='10.1.1.5', ihl=2, version=3)/ICMP())`**

IPv3

Min header length=5

- Land
  - Packet with Src IP = Dest IP and Src port = Dest port
  - **`send(IP(src=target,dst=target)/TCP(sport=135,dport=135))`**

- Latierra
  - Send multiple Land packets to multiple ports simultaneously

Malformed Packet DoS Attacks

- Ping of Death
  - Send a ping packet that is > 64KB
  - **`send(fragment(IP(dst='10.0.0.5')/ICMP()/("X"*66000)) )`**
- Jolt2
  - Send stream of packet fragments & none have frag offset = 0
- Rose
  - Sends a stream of packet fragments, but keeps retransmitting the last fragment over and over again – thousands of times
  - "Rose" – developer named tool after wife

Malformed Packet DoS Attacks

- Teardrop, Nestea, Newtear, Bonk, Syndrop
  - Send overlapping packet fragments
- WinNuke
  - Send garbage data (not proper SMB format) to open Windows file sharing port 139 → blue screen of death

- Instead of launching each one of these individual attacks against a target, attackers have rolled together several individual DoS exploits together into a DoS suite
  - Try all attacks, just to see if one will crash the target
  - Targa, Spike, Toast (contains 56 attacks)

TCP RESET Spoofing

- Send a spoofed RESET packet to one side of a connection
  - Forces the connection to close
  - **`send(IP(src='10.1.1.1',dst='10.2.2.2')/TCP(dport=443, flags='R'))`**

- RESET may be ignored if sequence number is incorrect
  - Attacker must guess a valid number
  - Odds of guessing the number → 1 of 2^{32}?
    - Nope
  - OS will accept packet if number is within TCP window
    - Assume window is 65,535 → odds now 1 of 2^{16}
  - Send barrage of RESET packets with number spaced every 65536

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding

SYN Floods

- SYN flood exploits the TCP 3-way handshake
  - Attacker keeps opening half-connections until it fills the victim's connection queue
  - Send SYN packets to victim but do not reply to the SYN-ACKs
    - Server maintains state until timeout (typically ~60 sec)
    - Each session initiation (SYN) ties up resources on victim
      - Victim's machine has finite number (~1024) of open connections it can handle

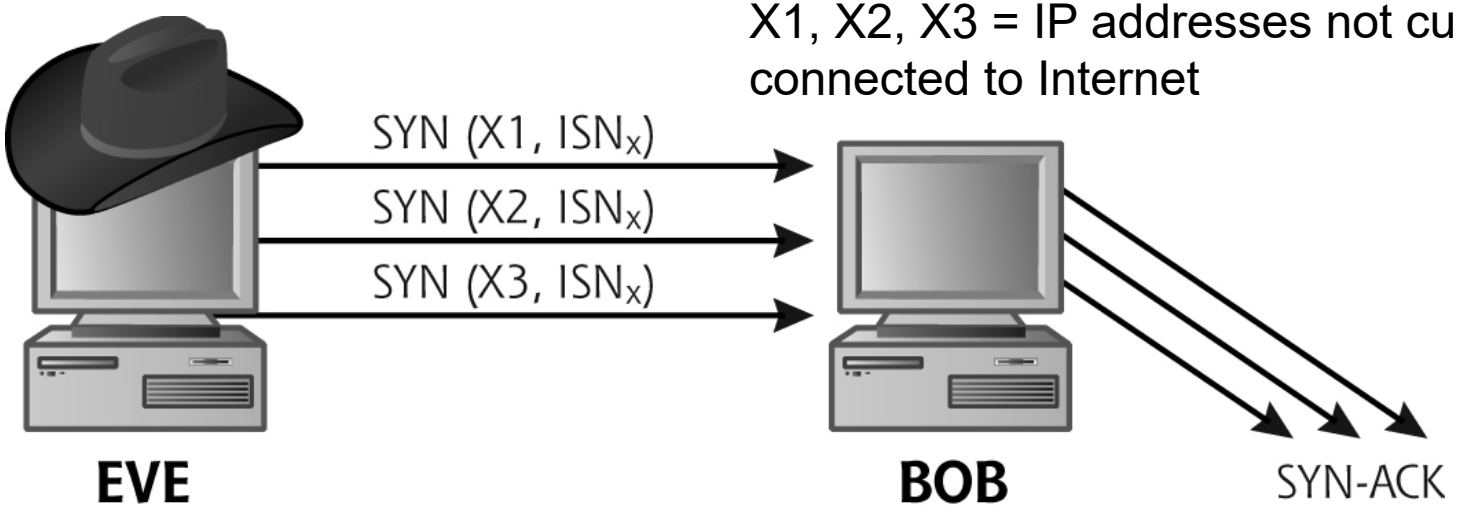
- SYN flood attacks have been around since at least 1996
- RFC 4987 "TCP SYN Flooding Attacks and Common Mitigations"

SYN Floods

- Even if connection queue is enormous and server resources are not exhausted, the SYN flood may consume all bandwidth
  - Prevents legit traffic on the link
- Tools
  - Synflood.c
  - Synful.c and Synk4.c
- Scapy
  - **`p=IP(dst='10.1.0.65')/TCP(sport=RandShort(),dport=[22,80],flags="S")/'Syn Flood'`**
  - **`send(p,loop=1,inter=0.3)`**
    - inter=0 → send as fast as possible (~770 per second) ☺

SYN Floods

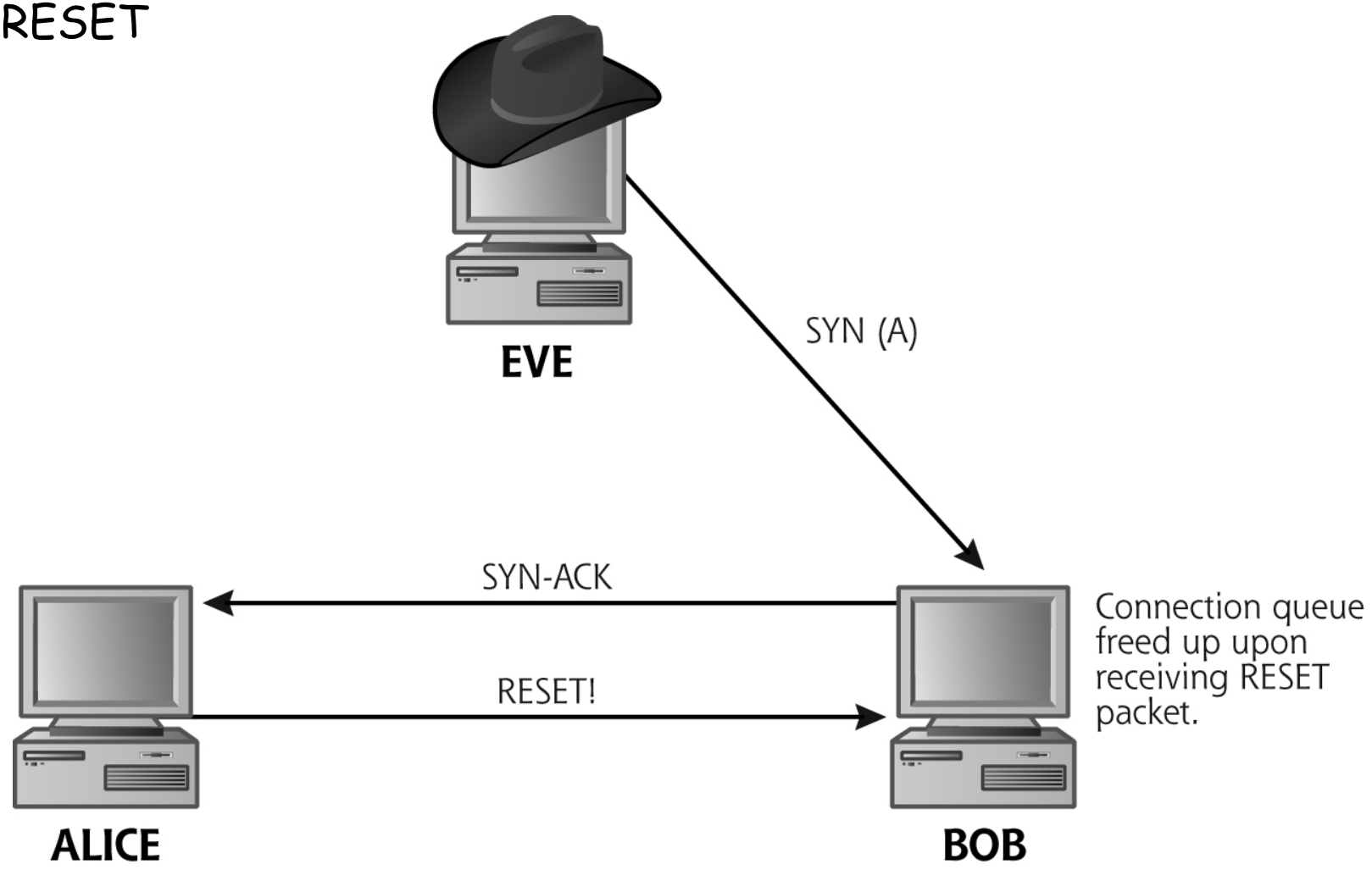
- If you do not want
  - your IP showing in the victim's log
  - to DoS yourself with all the SYN-ACKs from the server
- Send SYN packets to victim from a spoofed, unresponsive machine



Trying to kill Bob

SYN Floods and Address Spoofing

- If attacker uses a responsive (Alice) machine, the connection is RESET



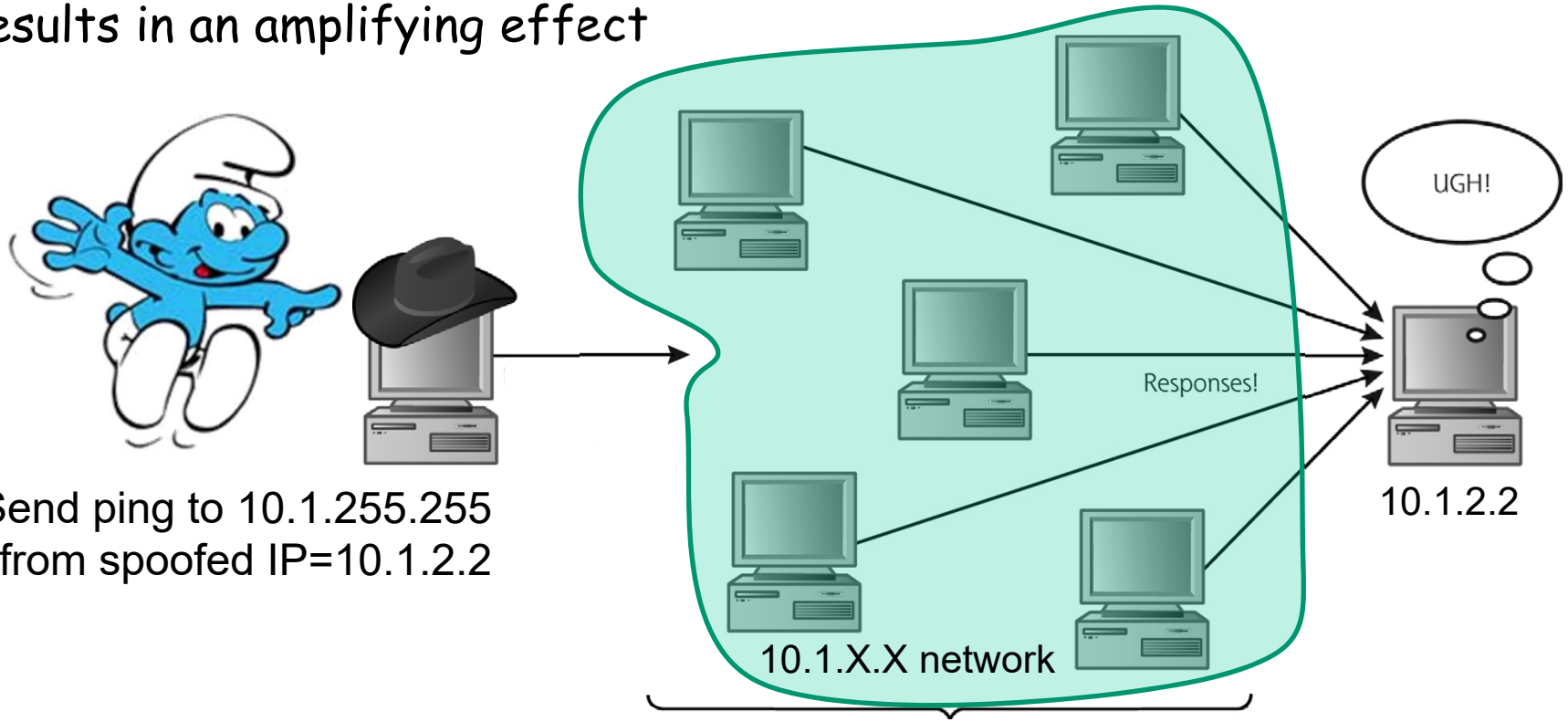
Trying to kill Bob

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding

Smurf Attacks (aka Directed Broadcast Attacks)

- Send spoofed ICMP echo request (aka "ping") to a network's broadcast address
- Spoofed machine receives many responses consuming most, if not all, of its bandwidth
- Results in an amplifying effect



Scapy

```
send(IP(src='10.1.2.2',dst='10.1.255.255')/ICMP())
```

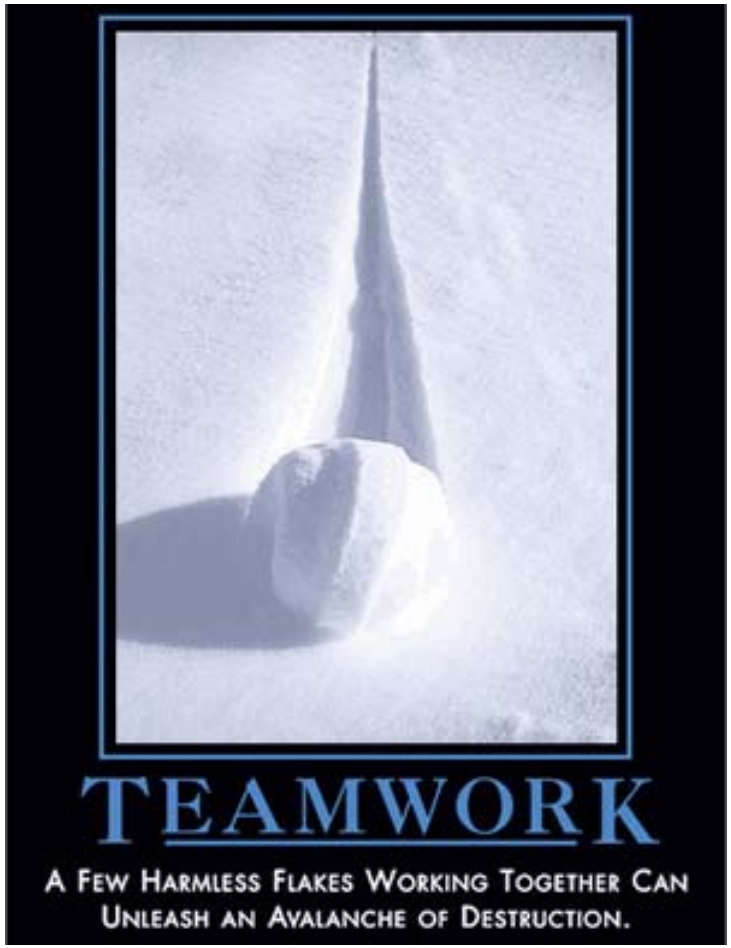
Other Smurf Tools and Amplifiers

- Fraggle, UDP-based cousin of Smurf
  - Uses the UDP echo service (port 7)
    - Service echoes data received back to sender
  - Can also send data to closed UDP ports
    - Generates an ICMP port unreachable

- PapaSmurf-improved/optimized version - combines Smurf/Fraggle
  - Allows use of multiple amplifier networks

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding



Distributed Denial of Service Attacks (DDoS)

- SYN Flood and Smurf are limited to one machine generating attacks
- DDoS attacks allow a coordinated attack using an unlimited number of machines
- Discovered in late Summer 1999
  - Proliferated in November and December 1999 with major attacks in February 2000
    - Amazon, eBay, E*Trade
- Attacker first takes over numerous victim machines (zombies)
  - Zombies run zombie code installed by attacker
  - Zombie code responds to commands from the attacker

DDoS Tools

- Tribe Flood Network 2000 (TFN2K)
- Low Orbit Ion Cannon (LOIC)

- High Orbit Ion Cannon (HOIC)
- XOIC
- HULK (HTTP Unbearable Load King)
- DDOSIM—Layer 7 DDOS Simulator
- R-U-Dead-Yet
- Tor's Hammer
- PyLoris
- OWASP DOS HTTP POST
- DAVOSET
- GoldenEye HTTP Denial Of Service Tool
- Scapy / hping

Tribe Flood Network 2000 (TFN2K)

- TFN architecture is based on clients and servers
  - Clients control the servers
  - Servers attack
  - Single client can control thousands of servers
- TFN servers flood victim machine with a massive DoS using
  - UDP flood
  - SYN flood
  - ICMP flood
  - Smurf attacks
  - Malformed packet attacks
- www.packetstormsecurity.org/distributed/

TFN Architecture

DEFAULT COMMUNICATION:
Attacker to client: Any port
Client to server:
ICMP_ECHOREPLY

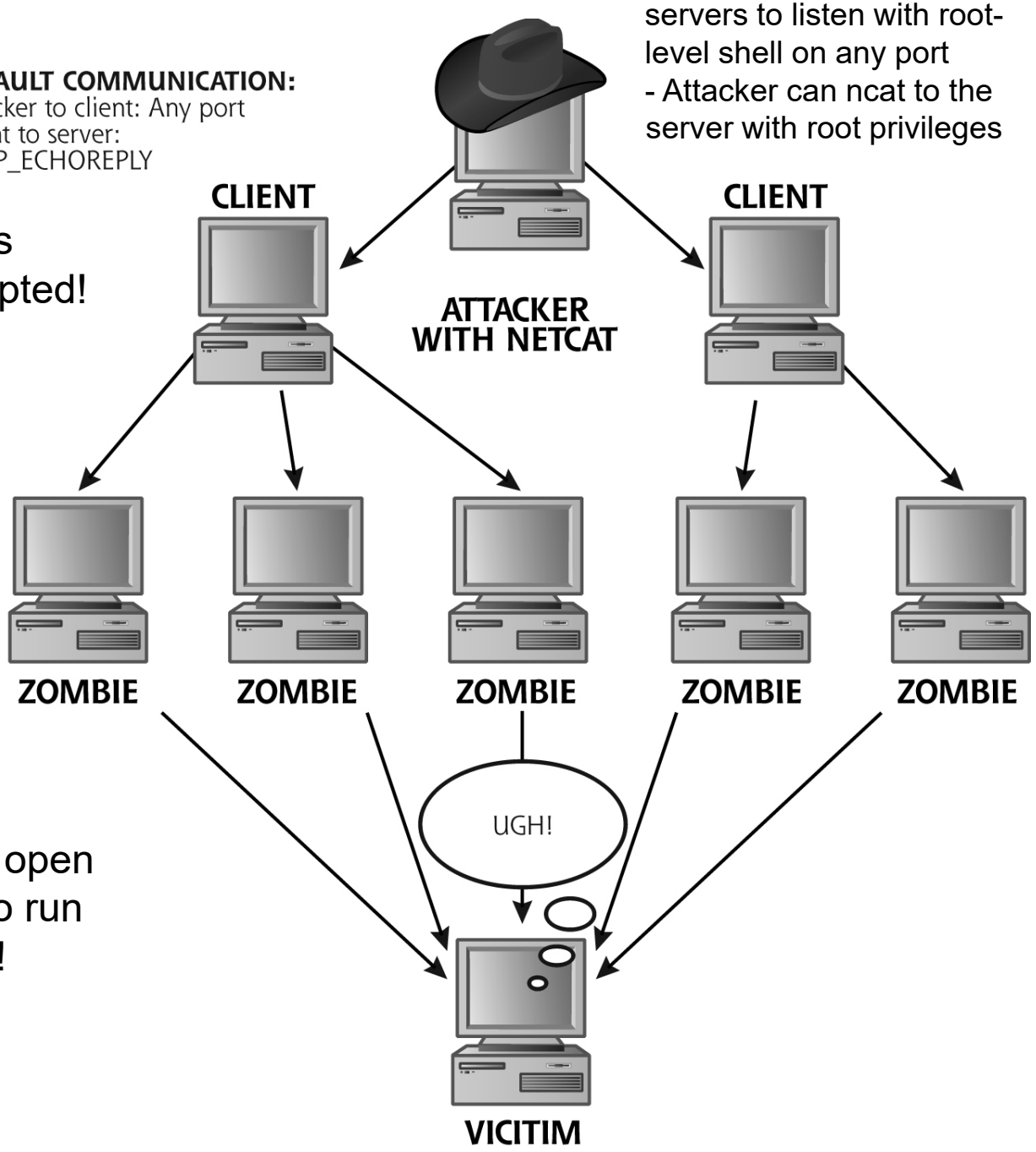
- Client can instruct servers to listen with root-level shell on any port
- Attacker can ncat to the server with root privileges

File listing controlled servers is stored on client, but it is encrypted!

Zombie = server

Server must be installed as root on numerous systems

Attacker has a “spoofed” path open to a huge number of servers to run any command simultaneously!



TFN Client-to-Server Communication

- All communication is carried by ICMP_ECHOREPLY
  - No corresponding ICMP_ECHOREQUEST was sent
  - Commands are sent in data field of reply
  - Makes detection and tracing much more difficult
    - No port # so nmap or netstat will not list a new port open
  - Echo reply often allowed into a network
- One-way, spoofed communication is supported
- Or two-way communication is allowed
  - With ICMP_ECHOREPLY going both ways, from client to server and server to client!
- CAST-256 encryption of data between client and server
  - So admins on zombie machines cannot take control of the zombie

Another Popular DDoS Tool - LOIC

- Low Orbit Ion Cannon (LOIC)
  - Floods server with TCP or UDP packets, or HTTP requests
- Used to attack
  - Scientology websites
  - Recording Industry Association of America's website in October 2010
  - Companies and organizations that opposed WikiLeaks
    - Operation Payback in December 2010
- Now has web-based counterpart
  - Low Orbit Web Cannon

Computer and Network Hacker Exploits

- Step 1: Reconnaissance
- Step 2: Scanning
- Step 3: Gaining Access
  - Application and Operating System Attacks
  - Network Attacks
  - Denial of Service Attacks
    - Malformed Packet Attacks
    - SYN Floods
    - Smurf Attacks
    - Distributed DoS Attacks
      - Reflected DDoS Attacks
      - Pulsing Zombies
      - Reflection DDoS Attacks - DNS
- Step 4: Maintaining Access
- Step 5: Covering Tracks and Hiding

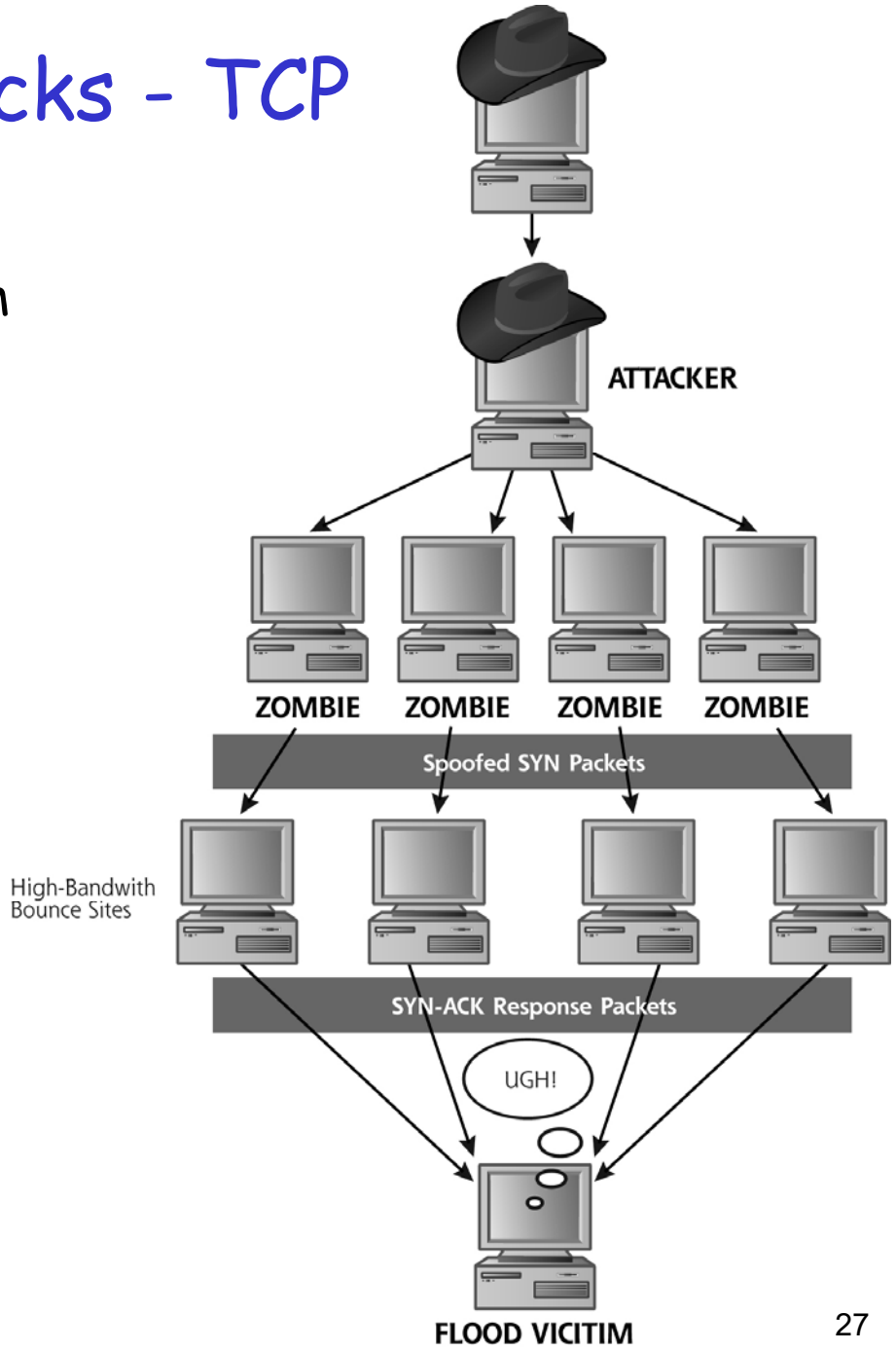
Attackers use reflection techniques for larger DDoS attacks

Posted on 17 April 2014.

http://www.net-security.org/secworld.php?id=16707

Reflected DDoS Attacks - TCP

- Using TCP three-way handshake, attacker can bounce a flood from the zombie to the victim
  - Obscures the source
- Zombie sends a SYN to a legit site with lots of bandwidth
  - Major WWW service
  - Software vendors
- Legit site sends a SYN-ACK to flood the victim
- Makes tracing the attack even more difficult



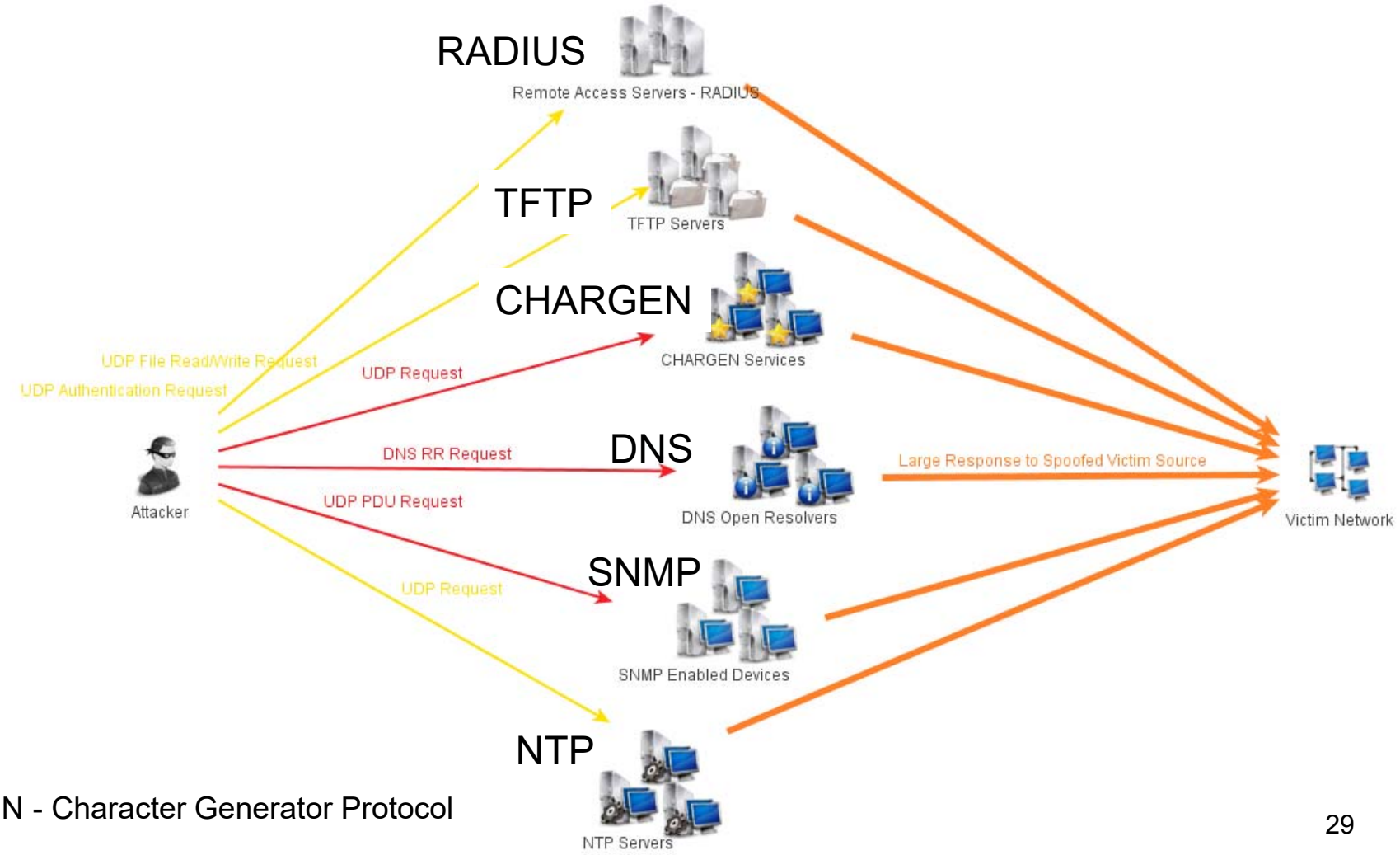
Pulsing Zombies

- Tracing back an active DDoS attack is doable but very laborious
  - ISP can step router-by-router & ISP-by-ISP to find attacker

- To confound investigations, tools implement pulsing zombies
  - Each zombie floods target for say 10 min,
    then goes dormant for 30 min
  - With lots of async zombies, the flood is still very effective
  - Silent zombies are much more difficult to locate!

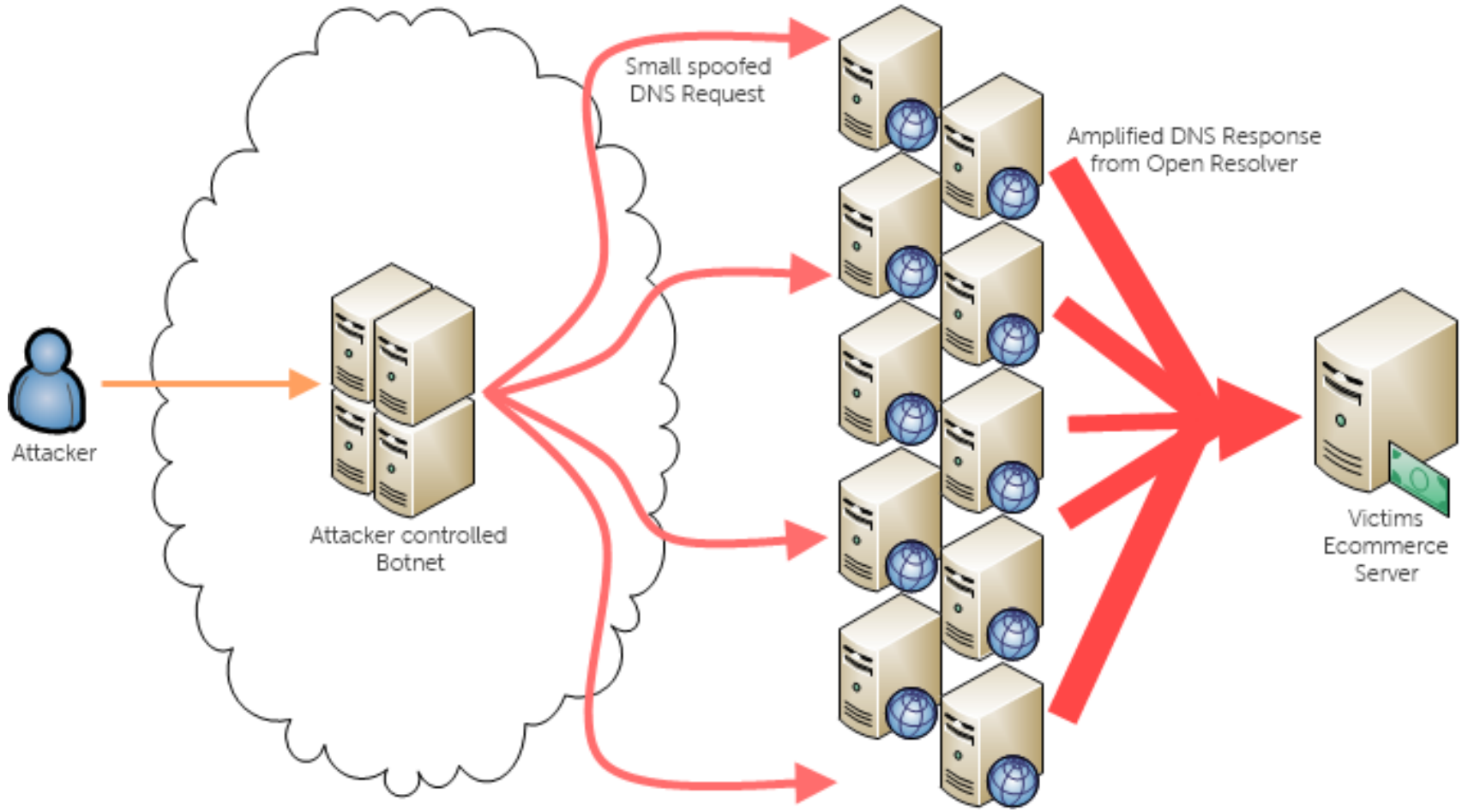
Reflection DDoS Attacks - UDP

- Any UDP-based service is a potential vector for DDoS attacks
  - Source IP addresses can be spoofed



CHARGEN - Character Generator Protocol

Reflection DDoS Attacks - DNS



http://www.secureworks.com/assets/image_store/png/large.research.blog.dns.amplification.fig2.png

Reflection DDoS Attacks - DNS

- DNS uses UDP – fire and forget ☺
- Goal - saturate the target name server's bandwidth rather than the name server itself
- Exploits name servers that allow open recursion
  - DNS resolver is *open* if it provides recursive name resolution for clients outside of its administrative domain

Reflection DDoS Attacks - DNS

- Attacker previously
  - compromised a poorly-configured name server
  - modified the server's zone file to include a DNS TXT resource record of approximately 3000-4000 bytes to serve as the amplification resource record
  - One example showing 2052/81 = 25 amplification

Protocol	Length	Info
DNS	81	Standard query 0x801b ANY ripe.net
DNS	2052	Standard query response 0x801b DNSKEY DNSKEY

DNS zone file contains the entire IP to domain mapping of the domain

Reflection DDoS Attacks - DNS

- Relies on an extension to the DNS protocol (EDNS0) that enables large DNS messages

- Attacker composes a DNS request of about 60 bytes to trigger delivery of a response of approximately 4000 bytes to target
  - Resulting amplification factor → 66:1
  - Responses contain 4000-byte DNS TXT record
    - Exceeds Ethernet maximum transmission unit
      - It is broken into multiple IP packet fragments
      - Forces reassembly at the destination
        - » Increases the processing load at the target
        - » Enhances deception

Reflection DDoS Attacks - DNS

IP of an open DNS resolver

- 64-byte DNS query: **`dig ANY isc.org @x.x.x.x`**
- 3,223-byte DNS response:

```
; <<>> DiG 9.7.3 <<>> ANY isc.org @x.x.x.x
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5147
;; flags: qr rd ra; QUERY: 1, ANSWER: 27, AUTHORITY: 4, ADDITIONAL: 5

;; QUESTION SECTION:
;isc.org.                       IN      ANY

;; ANSWER SECTION:
isc.org.                4084    IN      SOA     ns-int.isc.org. hostmast
isc.org.                4084    IN      A       149.20.64.42
isc.org.                4084    IN      MX      10 mx.pao1.isc.org.
```

<<snip>>

```
;; Query time: 176 msec
;;SERVER: x.x.x.x#53(x.x.x.x)
;; WHEN: Tue Oct 30 01:14:32 2012
;; MSG SIZE  rcvd: 3223
```

http://blog.cloudflare.com/deep-inside-a-dns-amplification-ddos-attack

DDoS Reflection

LILY HAY NEWMAN SECURITY 03.01.18 11:01 AM

GITHUB SURVIVED THE BIGGEST DDOS ATTACK EVER RECORDED

- On 28 Feb 18, 12:15 pm ET, 1.35 terabits per second of traffic hit the developer platform GitHub all at once. It was the most powerful distributed denial of service attack recorded to date—and it used an increasingly popular DDoS method, no botnet required.

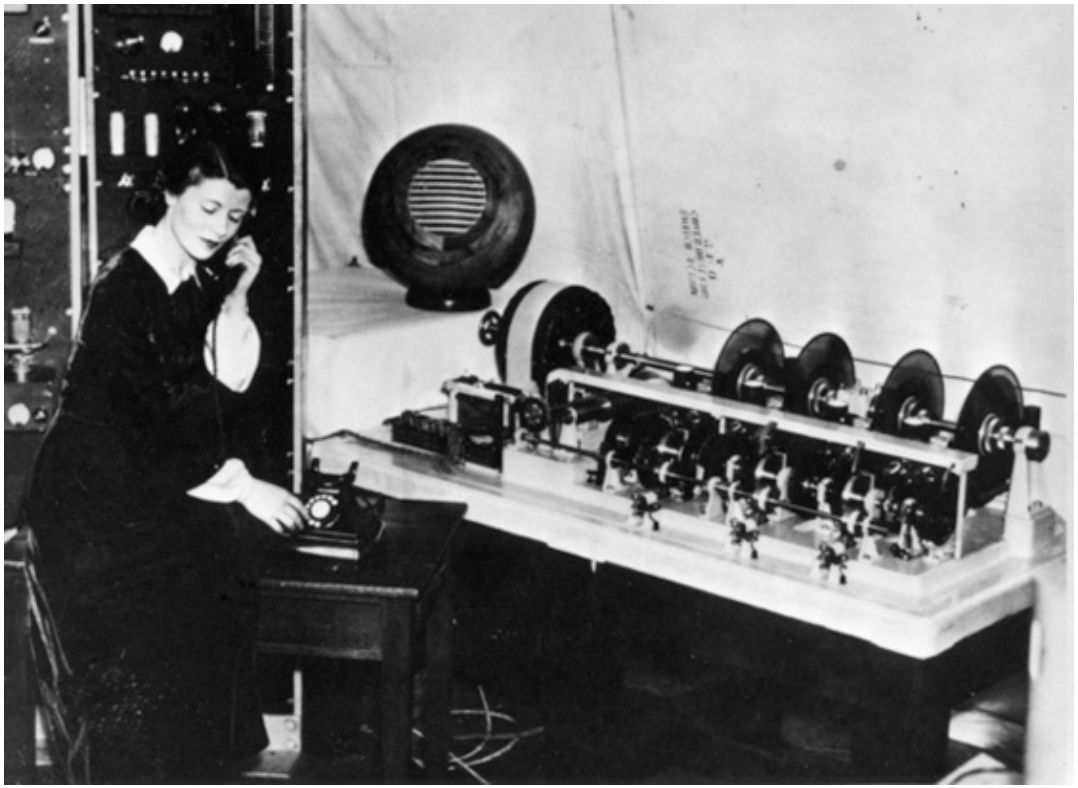
- About 100,000 memcached servers, mostly owned by businesses and other institutions, currently sit exposed online with no authentication protection, meaning an attacker can access them, and send them a special command packet that the server will respond to with a much larger reply.

- 15 byte request results in a 750kB response (51,200x amplification)

https://www.wired.com/story/github-ddos-memcached/
https://blog.cloudflare.com/memcrashed-major-amplification-attacks-from-port-11211/

Reflection DDoS Attacks - NTP

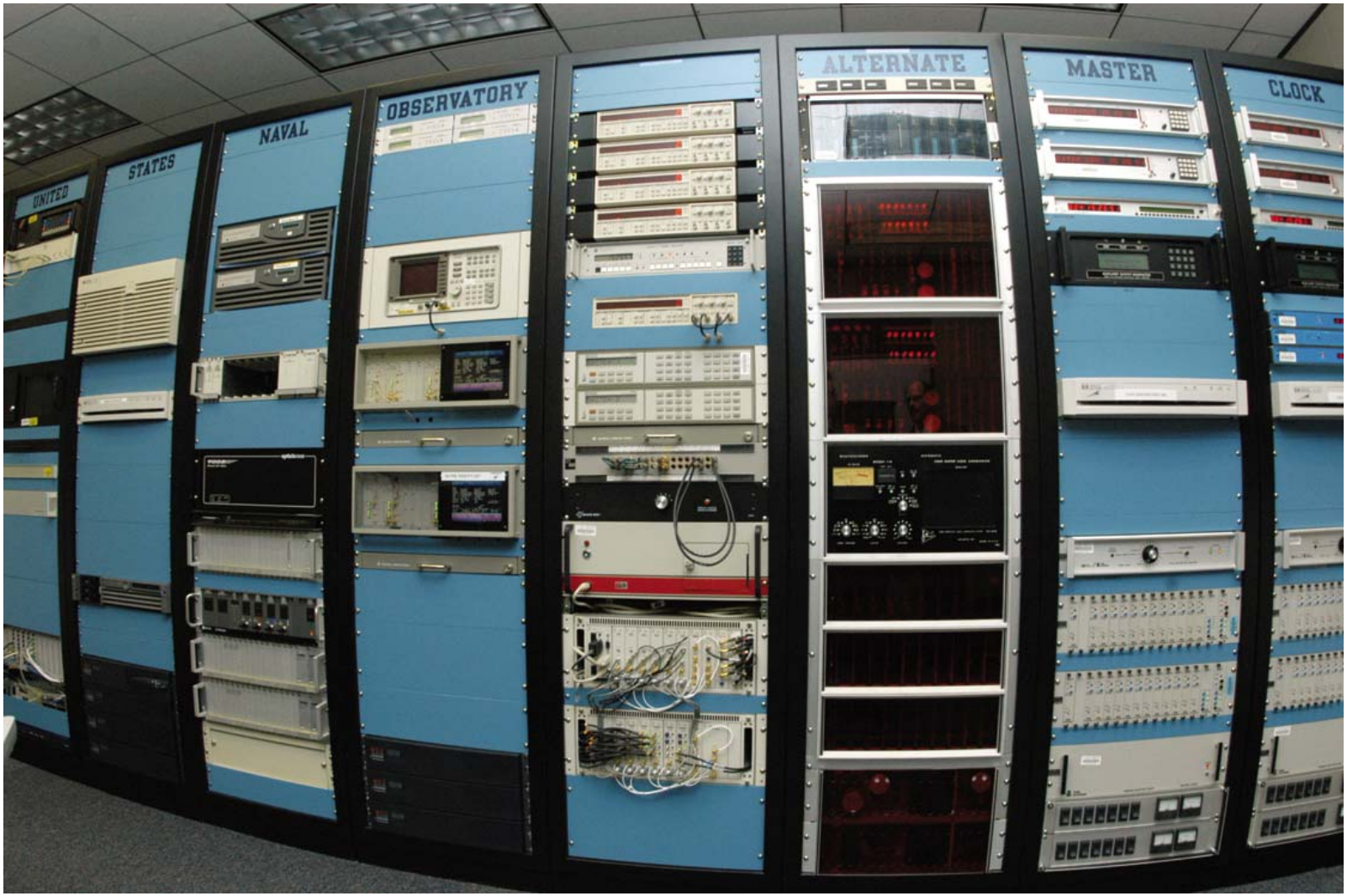
- Network Time Protocol (NTP)
  - UDP (port 123) protocol to sync clocks over computer networks
- UK's original speaking clock & original voice of the clock Jane Cain
- A common way to synchronize clocks was to telephone the speaking clock to get the precise time
  - UK telephone number for speaking clock was 123



http://blog.cloudflare.com/understanding-and-mitigating-ntp-based-ddos-attacks

Reflection DDoS Attacks - NTP

- U.S. Naval Observatory Alternate Master Clock at Schriever AFB



Reflection DDoS Attacks - NTP

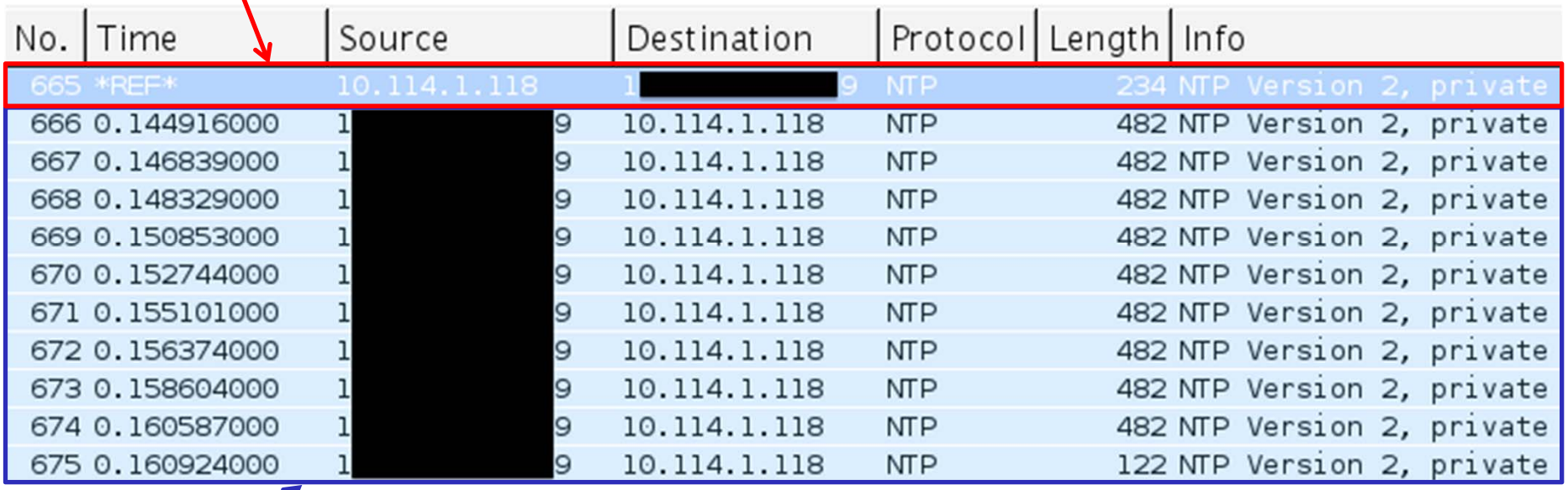
- What makes NTP so attractive to DDoS? Amplification!

- NTP command called monlist (aka MON_GETLIST) can be sent to an NTP server for monitoring purposes
  - Returns the addresses of up to the last 600 machines that the NTP server has interacted with
  - This response is much larger than the request sent making it ideal for an amplification attack

http://blog.cloudflare.com/understanding-and-mitigating-ntp-based-ddos-attacks

Reflection DDoS Attacks - NTP

- **`ntpdc -c monlist <<NTP server IP>>`**
- Request packet is 234 bytes



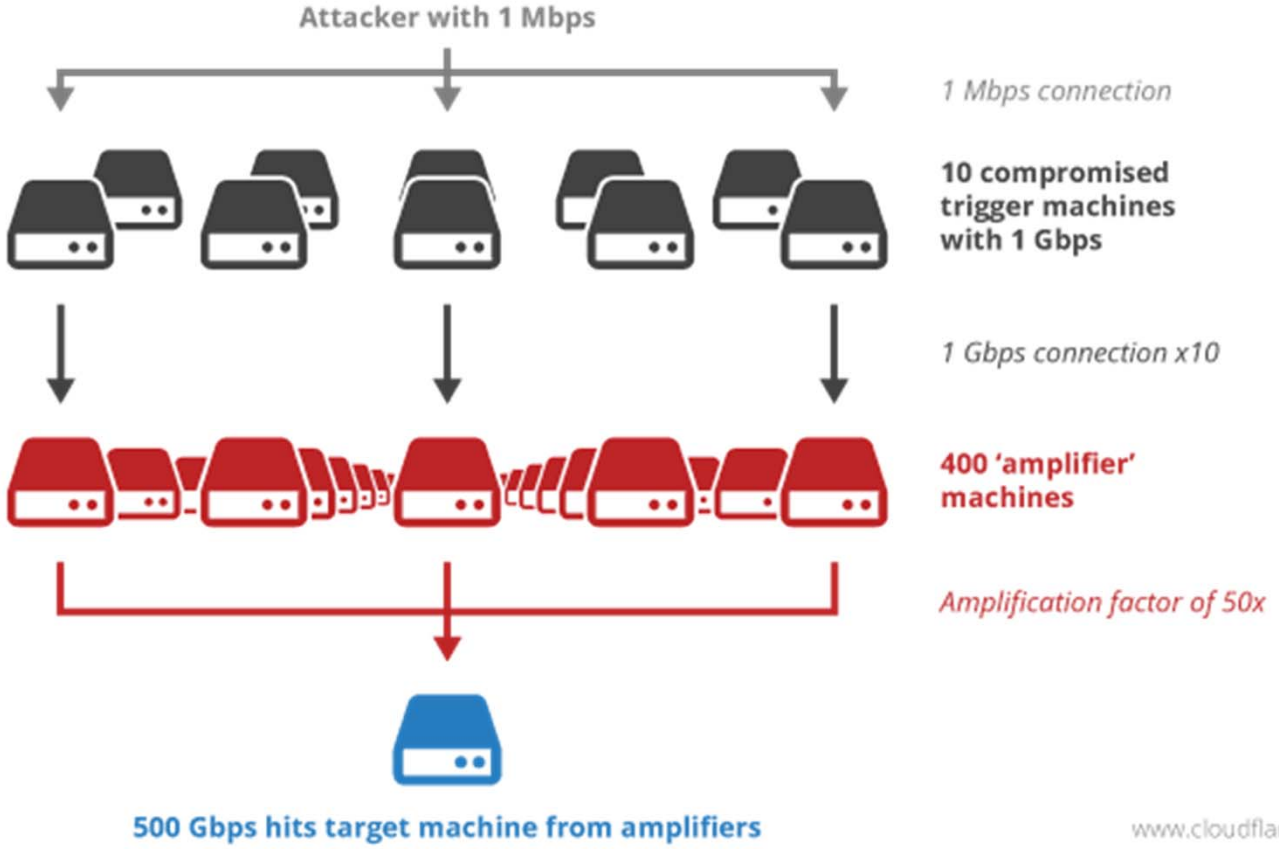
No.	Time	Source	Destination	Protocol	Length	Info
665	*REF*	10.114.1.118	1[redacted]9	NTP	234	NTP Version 2, private
666	0.144916000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
667	0.146839000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
668	0.148329000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
669	0.150853000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
670	0.152744000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
671	0.155101000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
672	0.156374000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
673	0.158604000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
674	0.160587000	1[redacted]9	10.114.1.118	NTP	482	NTP Version 2, private
675	0.160924000	1[redacted]9	10.114.1.118	NTP	122	NTP Version 2, private

- Response (only 55 IP addresses) is split across 10 packets (4,460B) bytes—each response packet contains 6 addresses
  - Amplification factor of 19
    - Since the response is sent in many packets an attack using this would consume a large amount of bandwidth and have a high packet rate

http://blog.cloudflare.com/understanding-and-mitigating-ntp-based-ddos-attacks

Reflection DDoS Attacks - NTP

- What if the NTP server responded with 600 IP addresses?
  - 100 packets—over 48KB in response to a 234B request
  - That's an amplification factor of 206



http://blog.cloudflare.com/understanding-and-mitigating-ntp-based-ddos-attacks

From SYN Floods to HTTP Floods

- SYN floods
  - Typically spoofed
  - Focused on consuming bandwidth or connection queue
  - Easier to block with SYN cookies in OS
  - ISPs can block by looking for abnormal traffic patterns
    - Lots of SYNs with no follow-up packets
- HTTP floods
  - Zombie
    - completes three-way handshake and
    - sends HTTP GET for various common pages, such as index.html
    - uses real (not spoofed) IP address of zombie (not the attacker)
  - Much harder to differentiate from normal traffic