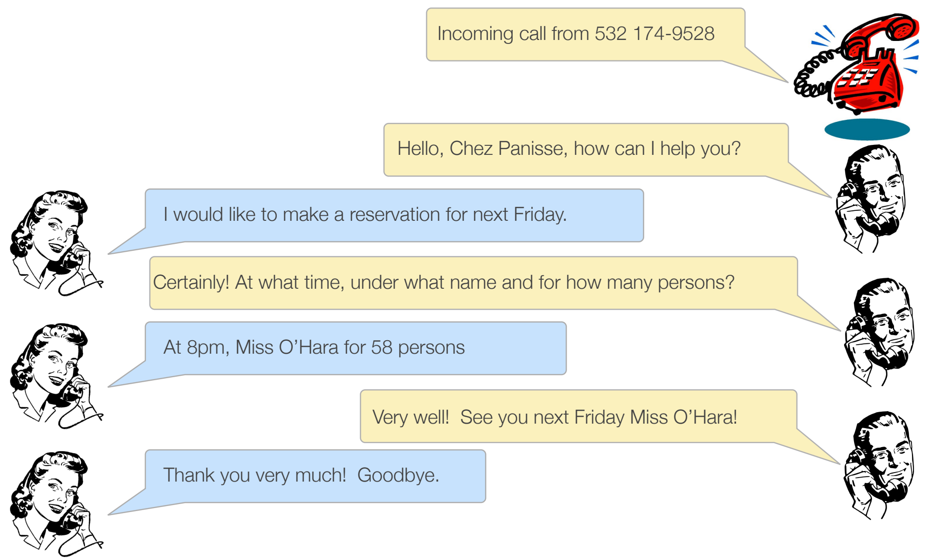
Cybersecurity: Denial of Service

Ozalp Babaoglu

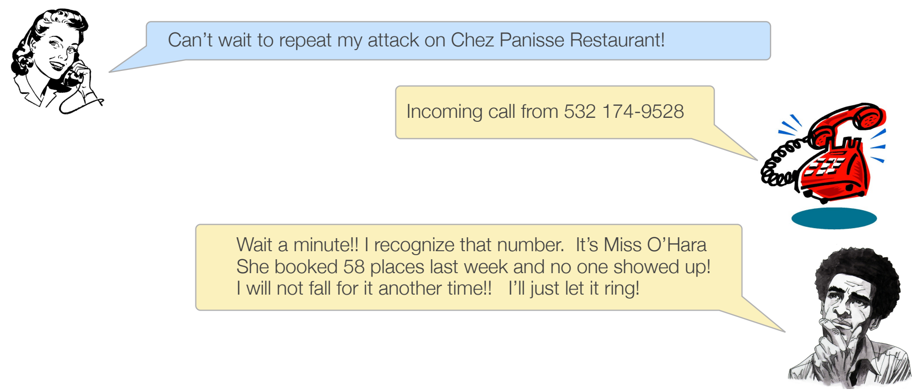
Denial of Service

- *Availability* refers to the ability to use a desired information resource or service
- A *Denial of Service attack* is an attempt to make that information (resource or service) unavailable to legitimate users
- The most common attacks are aimed at Internet hosts, whose services are temporarily denied
- Different motivations: economic interests, cyber-extortion, cyber-warfare, protest, hacktivism, etc.
- Started in late 1990s, still very common (and dangerous) today

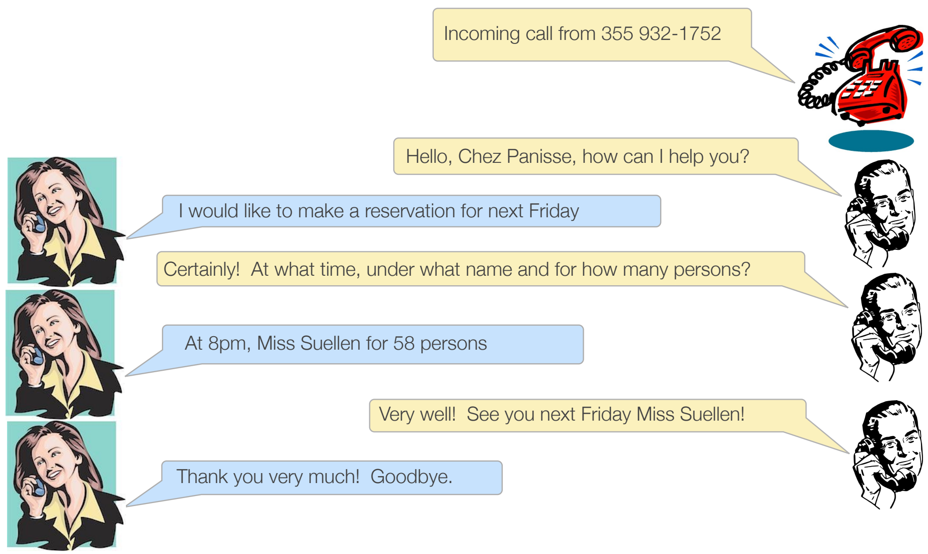
A metaphor: Denial-of-Dinner Attack



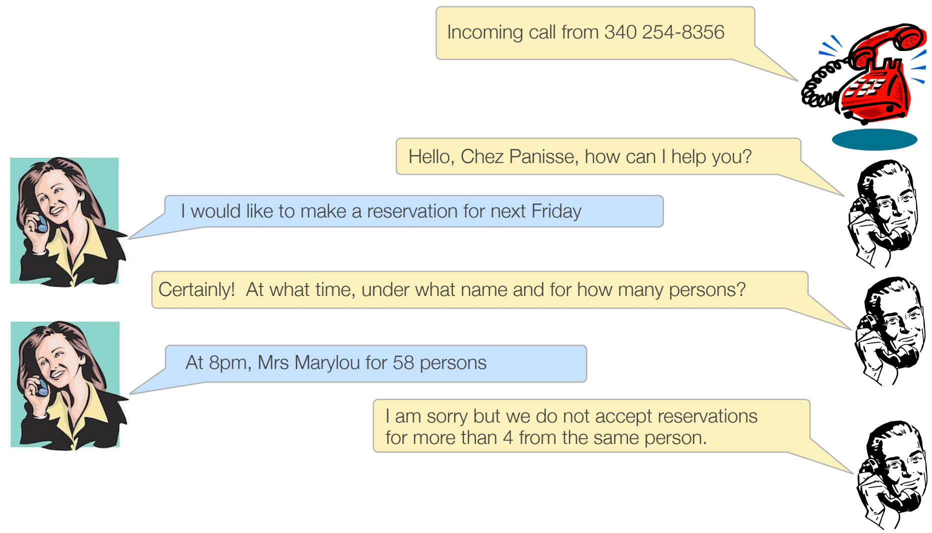
Denial-of-Dinner Attack 2



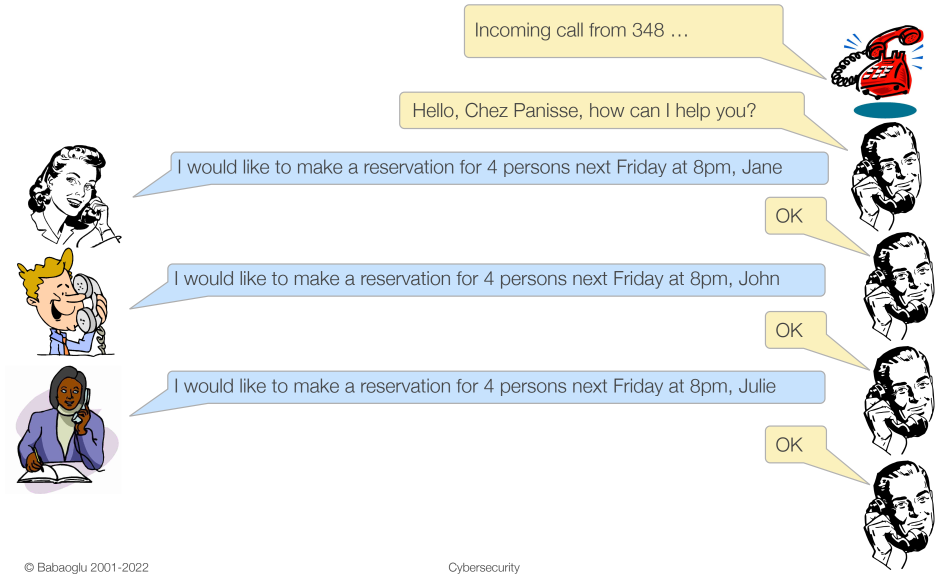
Denial-of-Dinner Attack 3



Denial-of-Dinner Attack 4



Distributed Denial-of-Dinner Attack 5



Asymmetry of costs

- With the existing economic model for reservations, the restaurant is fighting a losing battle
  - It costs very little for the customer to *make* a reservation
  - It costs a lot for the restaurant to *lose* a reservation
- This asymmetry opens up the possibility for exploitation
- Need to balance the two costs to avoid exploitation
- We can try one of two possibilities
  - Lower the cost of losing a reservation
  - Increase the cost of making a reservation — ask for a credit card

Economic model

- In the physical world, DoS attacks are very rare because almost everything has a cost — real, indirect or social
- The cost model of the Internet does not tax volume, so it costs (almost) the same to make one request or one million requests
- One way to increase the cost of a request is to increase the time it takes to complete it — CAPTCHA
- Can be effective in guarding services that involve human beings, e.g., creating accounts, directory look-up, image or document conversion

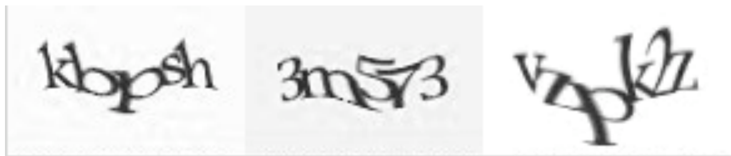
CAPTCHA

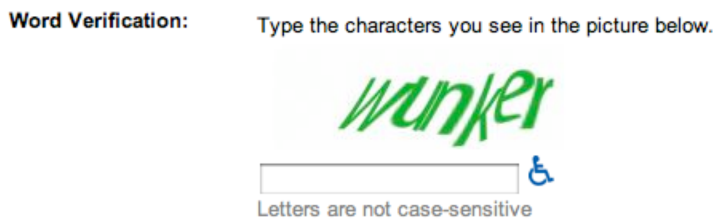
- **C**ompletely **A**utomated **P**ublic **T**uring test to tell **C**omputers and **H**umans **A**part
- Type of challenge-response test used in computing to determine whether the user is human
- CAPTCHA involves one computer (a server) which asks a user to complete a test
- The test can be *generated* and *graded* by a computer but a computer is not able to *solve* the test

CAPTCHA

- CAPTCHA requirements:
  - Most humans can solve easily
  - Current computers are unable to solve accurately
  - Do not rely on the attacker never having seen the given type of CAPTCHA before
  - Can be generated automatically but require artificial intelligence techniques to solve

CAPTCHA



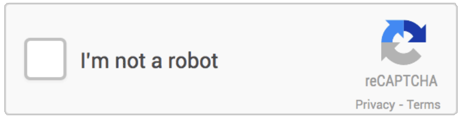


reCAPTCHA

- About 200 million CAPTCHAs are solved by humans around the world each day
- This amounts to more than 150,000 hours of work consumed each day
- reCAPTCHA improves the process of digitizing books by sending words that cannot be recognized by computers to the Web in the form of CAPTCHAs for humans to decipher

reCAPTCHA to noCAPTCHA

- Today, it is possible to distinguish humans from bots using sophisticated Machine Learning and AI techniques that take into account what a user does before and after ticking a simple checkbox



reCAPTCHA to noCAPTCHA

DoS types

- Two general strategies for attacks:
  - *Crash* the services
  - *Flood* the services
- Different ways of launching an attack:
  - Consumption of bandwidth
  - Consumption of host resources: RAM, disk space, CPU time
  - Disruption of configuration information (e.g., routing)
  - Disruption of state information (e.g., TCP sessions)
  - Disruption of information itself (cryptolocker)
  - Disruption of physical network components (LAN, WLAN, etc.)

DoS manifestations

- US-CERT defines symptoms of DoS attacks:
  - Unusually slow network performance (e.g., accessing web sites)
  - Inability to *provide* a service for remote access (web site)
  - Inability to *access* a remote service (web site)
  - Inability to *access* local information (files)
  - Increase in the number of spam emails received (email bomb)
  - Disconnection of a wireless or wired internet connection

Botnets, Zombies and DDoS

- Early DoS attacks were performed from a single host
- Today, "armies" of hosts are used to launch more effective "Distributed DoS" (DDoS) attacks: *botnets* of *zombies*
- "Zombie" refers to a compromised computer (infested by malware, virus, trojan horses, etc.) that can be used to perform malicious tasks, unbeknownst to its legitimate owner
- Botnets of zombies are remotely controlled by attackers

Some notable DoS attacks

- (1996) Attack against the New York City Internet Service Provider Panix (unavailable for one week, affected Internet Chess Club, NYT)
- (2000) Attack against Yahoo, eBay, Amazon, Datek, Buy, CNN, ETrade, ZDNet and Dell
- (2001) Code Red used 250.000 zombies to attack the White House
- (2013) Attack that brings down part of the Chinese Internet
- (October 2016) Hackers Used New Weapons to Disrupt Major Websites Across U.S.

October 2016 Attack

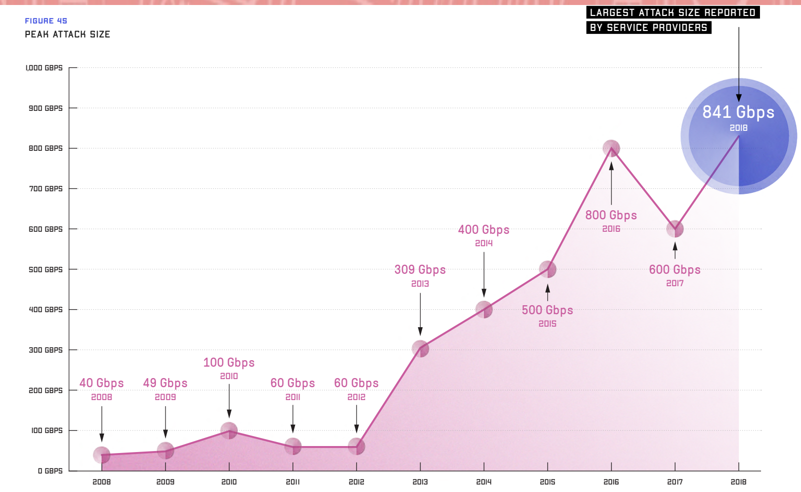
- Attack which took place over the weekend of October 21, 2016 caused problems in reaching several websites, including Twitter, Netflix, Spotify, Airbnb, Reddit, Etsy, SoundCloud and The New York Times
- Dyn, that hosts the Domain Name System (DNS), said it began experiencing what security experts called a distributed denial-of-service attack just after 7 a.m. Oct. 21
- The attack appears to have been highly distributed involving tens of millions of IP addresses from "IoT" devices like cameras, baby monitors and home routers that have been infected

October 2016 Attack

- The “Mirai” malware spreads to vulnerable devices by continuously scanning the Internet for IoT systems protected by factory default usernames and passwords

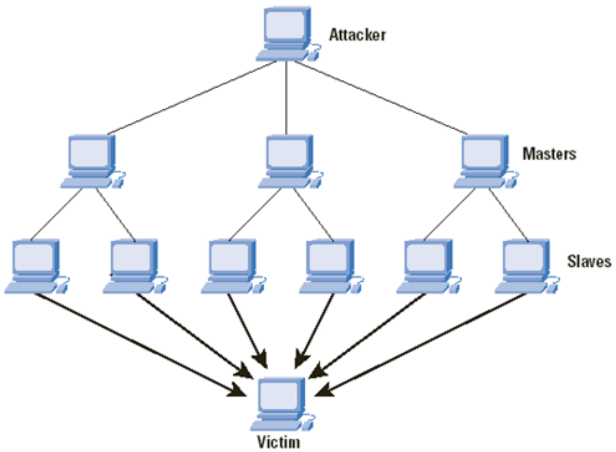
Username/Password	Manufacturer	Link to supporting evidence
admin/123456	ACTi IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/anko	ANKO Products DVR	http://www.cctvforum.com/viewtopic.php?f=3&t=44250
root/pass	Axis IP Camera, et. al	http://www.cleancss.com/router-default/Axis/0543-001
root/vizxv	Dahua Camera	http://www.cam-it.org/index.php?topic=5192.0
root/888888	Dahua DVR	http://www.cam-it.org/index.php?topic=5035.0
root/666666	Dahua DVR	http://www.cam-it.org/index.php?topic=5035.0
root/7ujMko0vizxv	Dahua IP Camera	http://www.cam-it.org/index.php?topic=9396.0
root/7ujMko0admin	Dahua IP Camera	http://www.cam-it.org/index.php?topic=9396.0
666666/666666	Dahua IP Camera	http://www.cleancss.com/router-default/Dahua/DH-IPC-HDW4300C
root/dreambox	Dreambox TV receiver	https://www.satellites.co.uk/forums/threads/reset-root-password-plugin.101146/
root/zlxx	EV ZLX Two-way Speaker?	?
root/juantech	Guangzhou Juan Optical	https://news.ycombinator.com/item?id=11114012
root/xc3511	H.264 - Chinese DVR	http://www.cctvforum.com/viewtopic.php?f=56&t=34930&start=15
root/hi3518	HiSilicon IP Camera	https://acassis.wordpress.com/2014/08/10/i-got-a-new-hi3518-ip-camera-modules/
root/klv123	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/klv1234	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/jvbzd	HiSilicon IP Camera	https://gist.github.com/gabonator/74cdd6ab4f733ff047356198c781f27d
root/admin	IPX-DDK Network Camera	http://www.ipxinc.com/products/cameras-and-video-servers/network-cameras/
root/system	IQinVision Cameras, et. al	https://ipvm.com/reports/ip-cameras-default-passwords-directory
admin/meinsm	Mobotix Network Camera	http://www.forum.use-ip.co.uk/threads/mobotix-default-password.76/
root/54321	Packet8 VOIP Phone, et. al	http://webcache.googleusercontent.com/search?q=cache:W1phozQZURUJ:community.freepbx.org/t/packet8-atas-phones/411
root/00000000	Panasonic Printer	https://www.experts-exchange.com/questions/26194395/Default-User-Password-for-Panasonic-DP-C405-Web-Interface.html
root/realtek	RealTek Routers	
admin/1111111	Samsung IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/xmhdipc	Shenzhen Anran Security Camera	https://www.amazon.com/MegaPixel-Wireless-Network-Surveillance-Camera/product-reviews/B00EB6FNDI
admin/smcadmin	SMC Routers	http://www.cleancss.com/router-default/SMC/ROUTER
root/ikwb	Toshiba Network Camera	http://faq.surveillixdvrsupport.com/index.php?action=artikel&cat=4&id=8&artlang=en
ubnt/ubnt	Ubiquiti AirOS Router	http://setuprouter.com/router/ubiquiti/airos-airgrid-m5hp/login.htm
supervisor/supervisor	VideoIQ	https://ipvm.com/reports/ip-cameras-default-passwords-directory
root/<none>	Vivotek IP Camera	https://ipvm.com/reports/ip-cameras-default-passwords-directory
admin/1111	Xerox printers, et. al	https://atyourservice.blogs.xerox.com/2012/08/28/logging-in-as-system-administrator-on-your-xerox-printer/
root/Zte521	ZTE Router	http://www.ironbugs.com/2016/02/hack-and-patch-your-zte-f660-routers.html

DDoS over the years



Peak Attack Size (Gbps)
Source: Arbor Networks 13th Annual Worldwide Infrastructure Security Report

Anatomy of a DDoS attack



IP Spoofing

- Most DDoS attacks rely on *spoofed source IP addresses*
  - the victim believes that the packet was sent by a machine other than the one that actually sent it
  - More effective if the spoofed IP address is of a host the victim trusts
- Exploits (corrupted) IP headers
- *IP Spoofing* has legitimate applications, for instance for simulating network load or traffic
- Can be exploited for DDoS since it:
  - makes it more difficult to trace back attackers (no accountability)
  - makes it more difficult to filter malicious traffic
  - allows errors and floods in network traffic

Some known attacks

- Ping of Death
- Teardrop
- SYN Flooding
- Reflector attack
- Smurf
- Slow HTTP DoS
- And many others

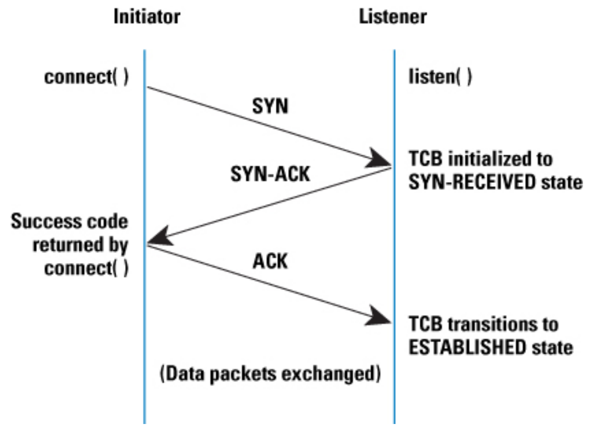
DoS attacks: Ping of Death

- The attacker creates IP packets containing more than 65,536 bytes, the limit defined in the IP protocol
- Malformed *ping* but can be generalized
- Exploits bugs in early implementations of TCP/IP when reassembling fragmented packets, causing a crash
- Today solved in most systems, can also be prevented with firewalls

DoS attacks: Teardrop

- Exploits IP packet fragmentation
  - Each fragmented packet identifies an offset that enables the entire packet to be reassembled
- The attacker sends malformed IP fragments with overlapping, over-sized payloads to the target machine, causing it to crash
- Affected mostly Windows systems, patched and no longer effective

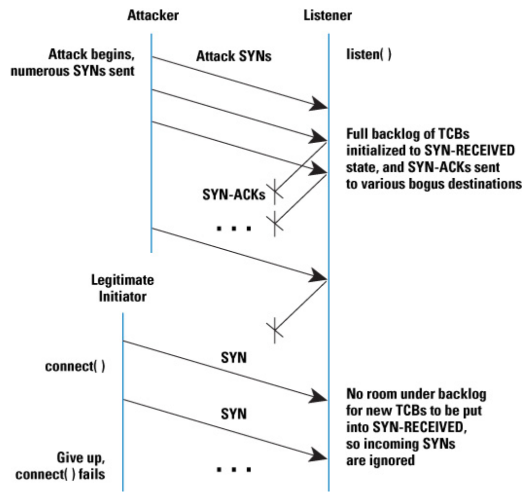
DoS attacks: SYN Flooding TCP 3-way handshake



DoS attacks: SYN Flooding

- Exploits vulnerabilities in the TCP three-way handshake through IP Spoofing
- The attacker (through the Botnet) initiates many TCP connection requests by sending SYNs to the victim host
- The victim initializes the connections in the *Transmission Control Block* (TCB), sends SYN-ACKs and waits for ACKs before declaring each connection ESTABLISHED
- Since the initial connection requests are spoofed, the SYN-ACK messages are lost and the ACKs never arrive
- The queue of incoming connections in the TCB is eventually exhausted and no more new connections can be accepted

DoS attacks: SYN Flooding

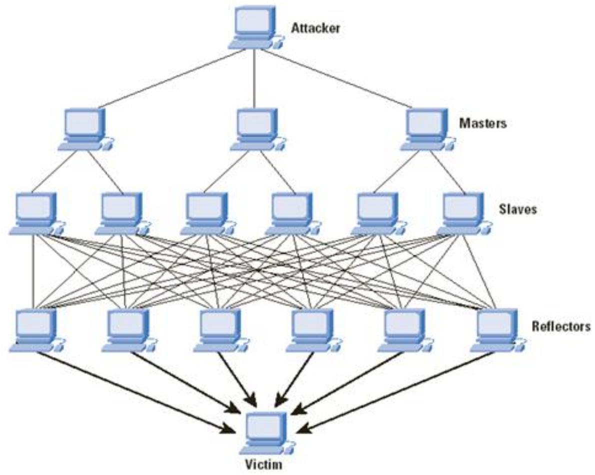


DoS attacks: Reflector

- Variation of the SYN Flood attack using the TCP three-way handshake with IP Spoofing
- The attacker (through the Botnet) initiates many TCP connection requests with many hosts (reflectors) where the (spoofed) source address is that of the victim
- Each of the reflectors sends its SYN-ACK message to the (spoofed) victim, flooding it

DoS attacks: Reflector

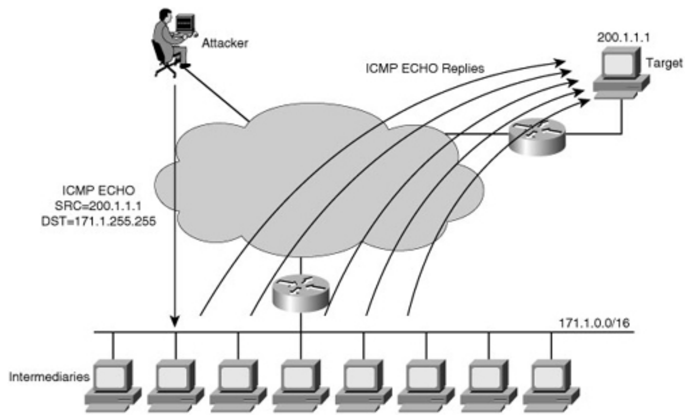
- Distributed Reflector DoS: more hosts, more distributed, more traffic



DoS attacks: Smurf

- Exploits vulnerabilities of *Internet Control Message Protocol* (ICMP), IP Spoofing and errors in network broadcast configurations
- The attacker sends many ICMP *echo-request* packets to the *broadcast address* of a subnet (useful for diagnostic purposes)
- These packets contain spoofed IP addresses set to that of the victim and are broadcast to all hosts in the subnet
- Every host responds by sending (a flood of) ICMP *echo-reply* packets to the victim

DoS attacks: Smurf



DoS attacks: Slow HTTP

- Exploits a vulnerability in thread-based web servers (like Apache) that wait for entire HTTP headers to be received before releasing the connection
- While servers typically make use of timeouts to end incomplete HTTP requests, the timeout, which is set to 300 seconds by default, is reset as soon as the client sends additional data
- By keeping the HTTP request open and feeding the server bogus data before the timeout is reached, the HTTP connection will remain open
- If an attacker succeeds in occupying all available HTTP connections on a web server, legitimate users would not be able to have their HTTP requests processed

Defenses

- DoS attacks cannot be prevented and there is no 100% effective defense
- Why is it so difficult to defend against DoS attacks?
  - Very difficult to distinguish between legitimate traffic and attacks
    - Filtering incoming flow might reject legitimate traffic
    - Filtering efficient only if detection is correct
  - Spoofed IP addresses make it very difficult to traceback the attacker
  - Heterogeneity of software and platforms

Defenses

- Three main defense strategies:
  - *Attack Prevention* (before the attack)
  - *Attack Detection and Filtering* (during the attack)
  - *Attack Source Traceback and Identification* (during and after the attack)
- A comprehensive solution should include all three lines of defense

Prevention

- Reduce the possibility of being a zombie
- Install security patches, antivirus, and intrusion detection systems
- Keep protocols and operating system up-to-date
- Install firewalls and configure network to filter input/output traffic
- Configure available resources
  - Alternate network paths
  - Load balancing
  - Additional servers/cloud-based resources

Detection

- Try to detect an attack as soon as possible and respond
  - Identification of ***statistical patterns*** of DDoS attacks and comparison of the same with live traffic
    - for known attacks, we can employ ***machine learning*** techniques
    - or search for signatures from a database of known attacks
    - effective for known attacks, but not for new ones
  - Identification of ***deviations from standard behavior*** of clients and usual network traffic (anomaly-based detection)
    - compare current network parameters with normal ones
    - effective against new attacks
    - keep the model of "normal traffic" updated
  - Hybrid approach combining both

Filtering

- Once detected, malicious traffic could be blocked by applying filters
- Where to apply filtering?
  - The closer to the attacker, the more effective the filter
  - The best solution would be to filter at the zombies (very difficult, often impossible)
- Preventive filters: try to reduce traffic with spoofed IP addresses on the network
  - The *source IP address* of outgoing traffic should belong to the originating subnetwork
  - The *source IP address* of incoming traffic should not

Filtering criteria

- *Source address*
  - Works if the attacker is known (but IP addresses are spoofed...)
  - Difficult to discover thousands of zombies/reflectors IP addresses
  - Difficult to deploy thousands of IP address filters
- *Service/port*
  - Works if the attack mechanism is known (UDP, TCP)
  - Not effective if the attacker used a common port or service
- *Destination address*
  - Works once the target is discovered
  - Legitimate traffic may be rejected
  - Useful to limit the consequences of an attack to other hosts served by the same ISP

Monitoring DDoS

- http://www.digitalattackmap.com/