

Protezione e Sicurezza nei Sistemi Operativi: Denial of Service

Ozalp Babaoglu

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA

Denial of Service

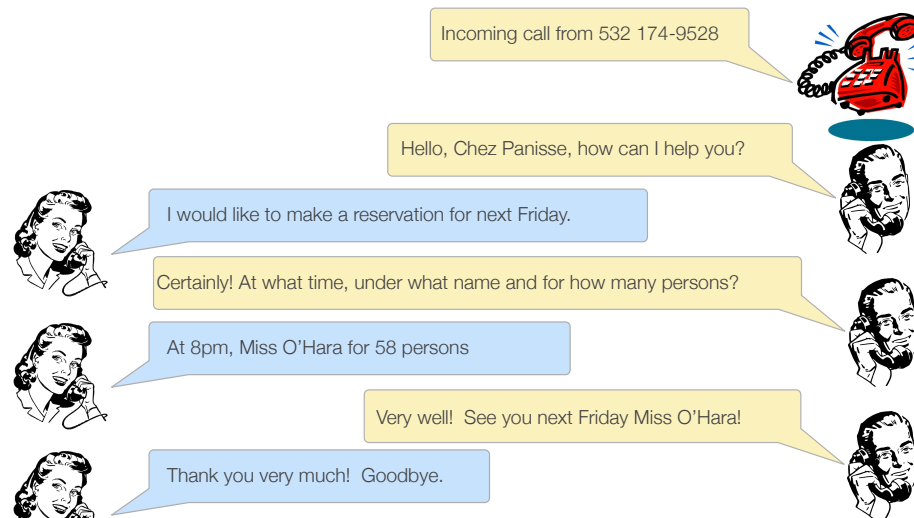
- *Availability* refers to the ability to use a desired information, resource or service
- A *Denial of Service attack* is an attempt to make that information (resource or service) unavailable to legitimate users
- The most common attacks are aimed at Internet hosts, whose services are temporarily denied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

© Babaoglu

Sicurezza

2

A metaphor: Denial-of-Dinner Attack

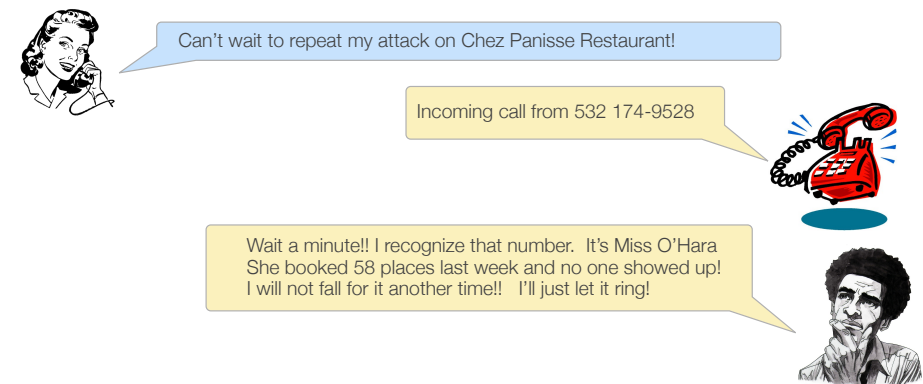


© Babaoglu

Sicurezza

3

Denial-of-Dinner Attack 2

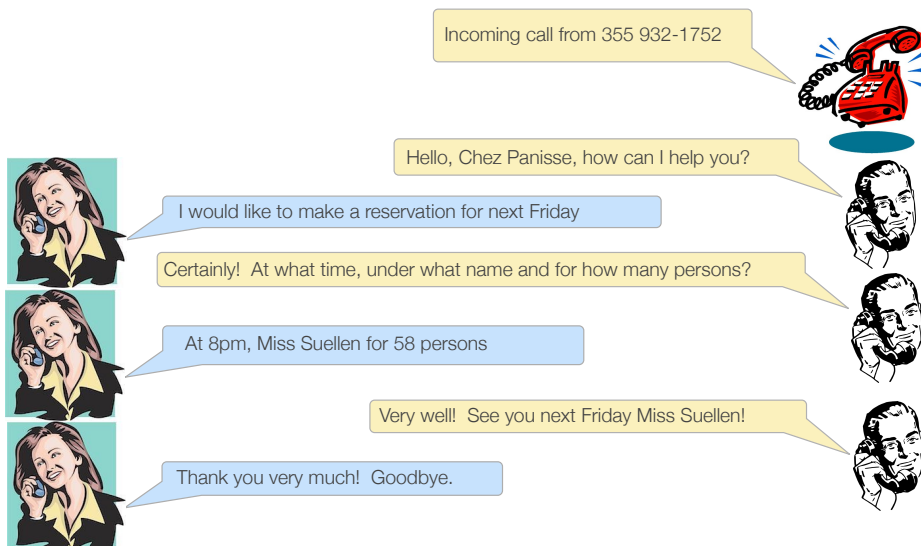


© Babaoglu

Sicurezza

4

Denial-of-Dinner Attack 3

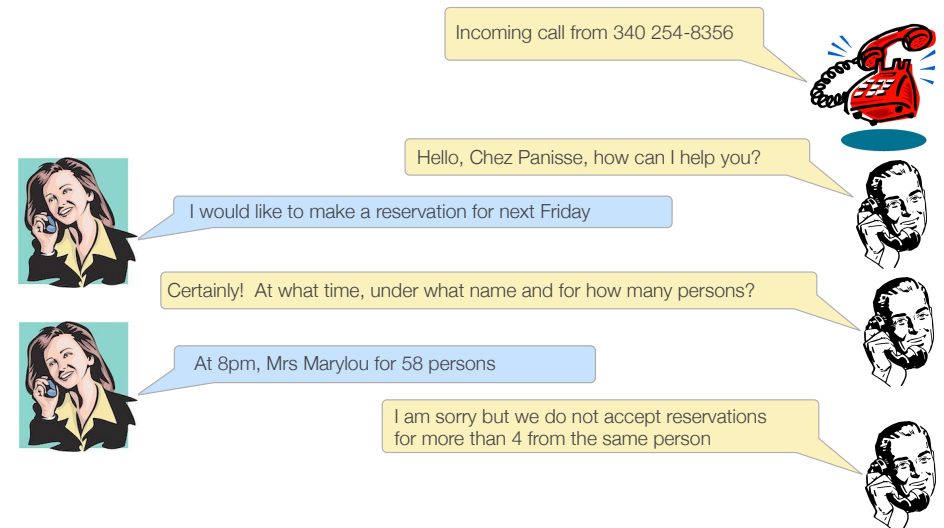


© Babaoglu

Sicurezza

5

Denial-of-Dinner Attack 4

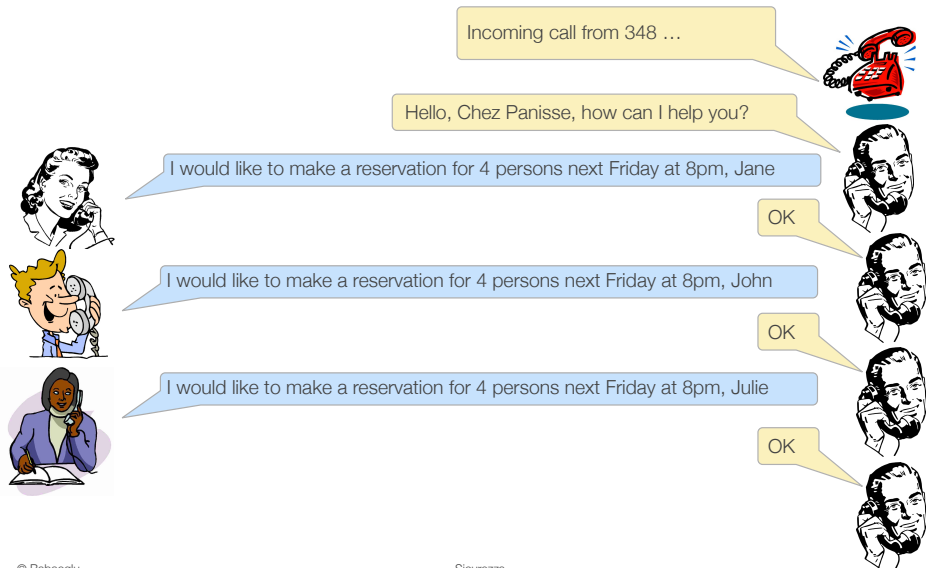


© Babaoglu

Sicurezza

6

Distributed Denial-of-Dinner Attack 5



© Babaoglu

Sicurezza

7

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
- Achieve both by asking for a credit card when reserving

© Babaoglu

Sicurezza

8

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 (in time) and intelligence required to make a request — CAPTCHA
- Can be effective by limiting service requests to human beings, e.g., creating accounts, directory look-up, image or document conversion by excluding bots

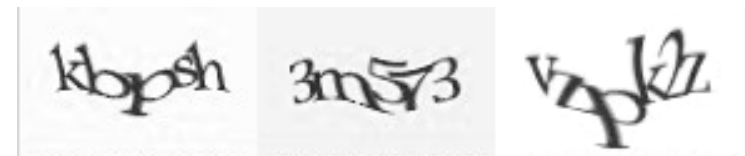
CAPTCHA

- **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



Word Verification:

Type the characters you see in the picture below.

munkr


Letters are not case-sensitive

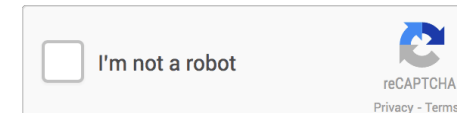
reCAPTCHA

- 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 text in scanned books or photographs by sending those 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.)

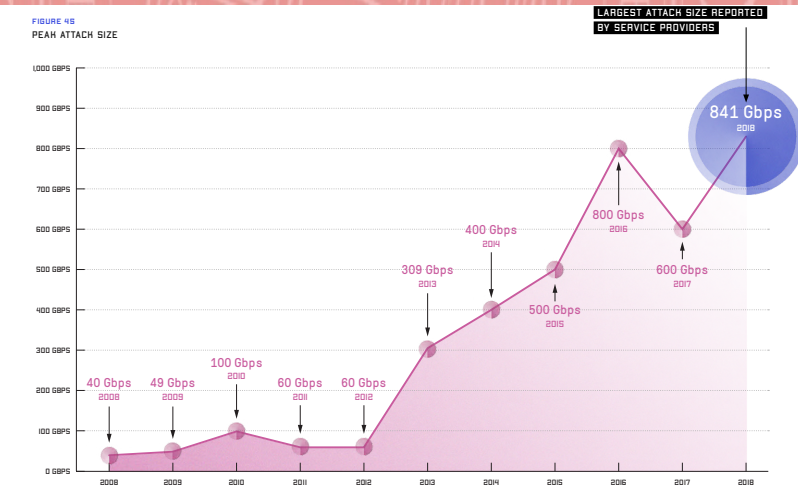
DoS manifestations

- 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* or *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

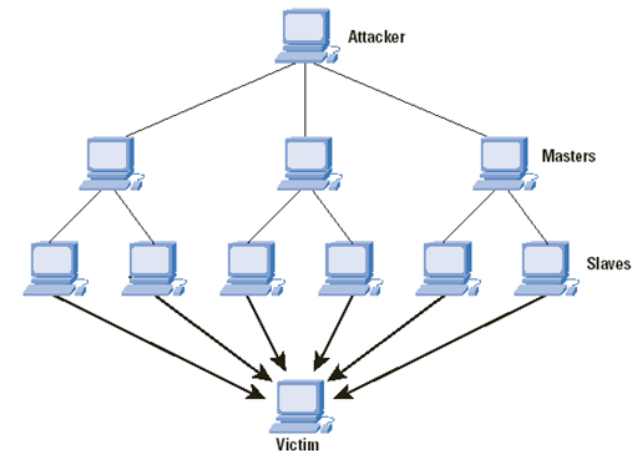
DDoS over the years



Peak Attack Size (Gbps)

Source: Arbor Networks 13th Annual Worldwide Infrastructure Security Report

Anatomy of a DDoS attack



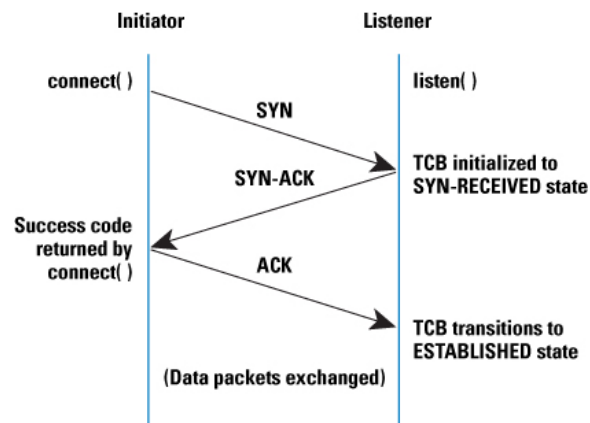
IP Spoofing

- 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

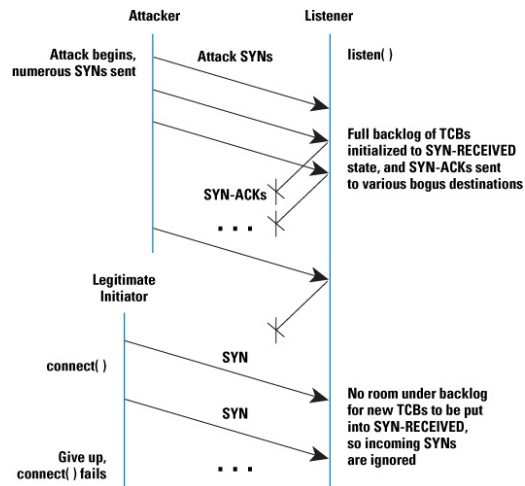
DoS attacks: SYN Flooding TCP 3-way handshake



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



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/>

