UNIVERSITE DE RENNES 1

IRISA

INRIA

Survivre aux dénis de service - DoS survivability

Bernard Cousin

Outline

- General Presentation of DoS
- Denial of Service
  - Host DoS
  - Distributed Service or Network DoS
- Distributed Denial of Service
- Some Solutions
- Framework against DDoS
- Conclusion

Denial of Service

- Denial of Service
  - The goal of a denial of service attack is to deny legitimate users access to a particular resource.
  - An incident is considered an attack if a malicious user intentionally disrupts service to a computer or network resource.
  - Resource exhaustion
    - Consumption of all network bandwidth, server processor, memory or disk spaces, etc.

Example of real DoS

- E-commerce sites
  - In February 2000, high traffic sites were faced with the task of handling huge amounts of spoofed traffic.
  - eBay/Amazon/Yahoo/CNN/Buy.Com/Datek/ZDNet
- Search engines
  - In July 2004 : Google, Yahoo (Akamai)
- Corporate sites
  - Recently, there have been attacks on Cisco which resulted in considerable downtime.
  - Microsoft : January 2001 (500 M$), 05/2003, 06/2004 (Akamai)
- Network Infrastructures
  - The first major attack involving DNS servers as reflectors occurred in January 2001.
    - The target was Register.com.
    - This attack, which forged requests for the MX records of AOL.com (to amplify the attack) lasted about a week before it could be traced back to all attacking hosts and shut off. It used a list of tens of thousands of DNS records that were a year old at the time of the attack.
  - First attack targeting routers : 01 September 2001
  - October 2002: massive attack against the 13 DNS root servers
  - July 2004 : large-scale attack that did affect the Internet name service managed by Akamai.
    - MSN.com, Microsoft.com and Yahoo.com outsource DNS services to Akamai for fast content delivery. By content caching, Akamai provides accelerated, dynamic and personalized web service.
- Some public blacklists have also been targeted by spammers and taken out of business.

Types of DoS attacks

- There are three general categories of attacks:
  - Against services
  - Against clients
  - Against networks
- Twofold nature of attacks:
  - Flooding
  - Exploit of vulnerabilities

DoS against Hosts

- Local DOS against client or service hosts
  - Processor exhaustion, consuming kernel memory
    - fork() bomb
  - Intentionally generate errors to fill logs
  - Consuming memory or disk space
  - Crashing
  - The power switch!

Host DoS: Countermeasures

- Countermeasures
  - Partition disks and segment memory
  - Limit user resource
    - Disk quotas
    - Set process limits
  - User identification and authorization
  - (File) access control
  - Monitoring of system activity/CPU/Disk Usage
  - Apply last software patches
  - Physical Security

Distributed Service and Network DoS

- TCP/IP level
  - TCP/IP stack attack
  - TCP connection attack
  - UDP bombing
- ICMP level
- MAC level
- DNS
- Email level
- Router level

TCP/IP Layer Attacks

- TCP/IP stack attack
- TCP connection attack
  - SYN flooding
  - RST attack
- UDP bombing

TCP/IP stack attack

- Exploit the weakness (bug) in TCP/IP stack.
  - Teardrop, NewTear, Newtear2, Bonk, and Boink, Land, Ping of Death
- The attacker sends the victim a pair of malformed IP/UDP/TCP fragments which get reassembled into an invalid IP/UDP/TCP packet.
- Upon receiving the invalid fragments, the victim host freezes (“blue-screens”) or reboots (stopping service and adding delay)
- Countermeasure: Apply vendor patches

TCP/IP stack attack

- Teardrop
  - IP fragmentation reassembly code do not handle properly overlapping IP fragment
    - 2 fragments with one of the fragment too small
    - Incorrect IP header fragment offset field
- New Tear or New Tear2
  - UDP/IP stack does not handle properly misformed UDP header information
    - UDP length > size of IP packet
- Land
  - TCP SYN packet with source address and port identical to destination address and port (i.e. spoofed)
- Bonk
  - Manipulates the fragment offset field in UDP/TCP packet to make a too big packet
  - Uses DNS port only (53)
- Boink
  - Same as a multiport Bonk

TCP SYN Flooding

- Also referred to as the TCP “half-open”attack
- To establish a legitimate TCP connection:
  - the client sends a SYN packet to the server
  - the server sends a SYN-ACK back to the client
  - the client sends an ACK back to the server to complete the three-way handshake and establish the connection

TCP SYN Flooding

- The attack occurs
  - The attacker initiating a TCP connection to the server with a SYN. (using a legitimate or spoofed source address)
  - The server replies with a SYN-ACK
  - The client then doesn't send back a ACK, causing the server to allocate memory for the pending connection and wait. (If the client spoofed the initial source address, it will never receive the SYN-ACK)

TCP SYN Flooding

- Results
  - The half-open connections buffer on the victim server will eventually fill.
  - The system will be unable to accept any new incoming connections until the buffer is emptied out.
  - There is a timeout associated with a pending connection, so the half-open connections will eventually expire.
  - The attacking system can continue sending connection requesting new connections faster than the victim system can expire the pending connections.

TCP SYN Flooding: Countermeasures

- Apply vendor's patches.
  - Most OS vendors have minimized the risks in newer OS releases and have patches for older releases.
- Install Ingress/Egress router filters to prevent some IP spoofing locally.
- Delegate the management of the establishment phase of TCP connections to front ends
- TCP cookies.

TCP Reset Attack

- TCP reset attack falsely terminates an established TCP connection.
- For instance:
  - An established TCP connection from host A to host B.
  - A third host, C,
    - spoofs a packet that matches the source port and IP address of host A, the destination port and IP address of host B, and the current sequence number of the active TCP connection between host A and host B.
    - sets the RST bit on the spoofed packet, so when received by host B, host B immediately terminates the connection.

TCP Reset Attack

- This results in a denial of service
  - Until the connection can be reestablished. However, the severity of such an attack is different from application to application.
    - BGP is very vulnerable as it relies on a persistent TCP session being maintained between BGP peers. If the connection gets terminated, it then takes time to rebuild routing tables and remote hosts may perform "route flapping".
- Counter-measure:
  - TCP utilizes sequence numbers:
    - To reassemble valid but out of order packets.
    - To ignore potentially spoofed packets.

UDP Service Attack

- UDP bombing
  - The culprit sends a large amount of UDP echo traffic all of it having a spoofed source address of a victim. This multiplies the traffic by the number of hosts.
- Improvement using broadcast address
  - E.g. *Fraggle*
  - Combine spoofing and reflection
  - This is the cousin of the *smurf* attack. This attack uses UDP echo packets with broadcast address.

UDP Service DoS: Countermeasures

- Verify the disabling of *echo*, *chargen* and all other unused services whenever possible, such /etc/inetd.conf on Unix, and “no udp smallservices” on Cisco IOS.

```
chargen stream tcp nowait root internal
chargen dgram udp wait root internal
```

- Filter UDP traffic at the firewall level.
  - Only allow legitimate traffic such as UDP port 53 (DNS)
  - Filter UDP port 7 (echo) and 19 (chargen)

Any Bombing

- Any type of protocol packet can be used to bomb any type of target:
  - For instance:
    - Ethernet data frames (individual or broadcast), B_PDU (bridging control frames) , IP (any type of address), UDP, TCP, ICMP, IGMP, DNS, etc.
- Countermeasure:
  - Traffic monitoring
  - Deny IP broadcast traffic onto your network

ICMP level

- Ping of death
- Smurf attack
- Host unreachable attack
- Redirect attack

Ping of Death

- Similar to TCP/IP stack attack but on ICMP driver
- The TCP/IP specification allows for a maximum packet size of 65,536 octets.
- The ping of death attack sends oversized ICMP datagrams (encapsulated in IP packets) to the victim.
- Some systems, upon receiving the oversized ICMP packet, will crash, freeze, or reboot, resulting in denial of service.
- Countermeasures:
  - Most systems are now immune, but apply vendor patches if needed.

Smurf attack

- A *smurf* attack consists of a host sending an ICMP echo request (*ping* command) to a network broadcast address. (usually network addresses with the host portion of the address having all **1**s)
- Every host on the network receives the ICMP echo request and sends back an ICMP echo response inundating the initiator with network traffic.

Smurf attack

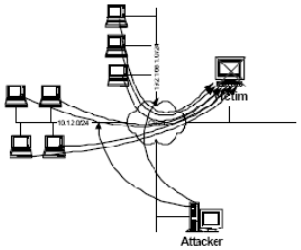
- There are 3 players in the smurf attack
  - the attacker, the intermediary (which can also be a victim) and the victim
- The attacker spoofs the IP source address as the IP of the intended victim to the intermediary network broadcast address.
- Every host on the intermediary network replies, flooding the victim and the intermediary network with network traffic.
- Result: Performance may be degraded such that the victim and intermediary networks become congested and unusable

Smurf Attack: an Example

- 1. Attacker sends ICMP echo request with spoofed source IP

  [IP: Victim =>10.1.2.255 (ICMP echo req)]
- 2. Victim is flooded with ICMP echo responses

  [IP: 10.1.2.3 => Victim (ICMP echo resp)]
  [IP: 10.1.2.7 => Victim (ICMP echo resp)]
  [IP: 10.1.2.17 => Victim (ICMP echo resp)]
  [IP: 10.1.2.35 => Victim (ICMP echo resp)]
  etc.
- 3. Victim hangs!



Smurf: Countermeasures

- Configure routers to deny IP broadcast traffic onto your network from other networks. In almost all cases, IP-directed broadcast functionality is not needed.
- Configure hosts (via kernel variable) to NOT reply to a packet sent to a broadcast address
- Configure Ingress/Egress filters on routers to counteract IP address spoofing.

Unreachable Host Attack

- **Unreachable Host Attack**
  - An "Host Unreachable" ICMP message is sent to the target about a fake destination.
  - The target will drop all active sessions with the (fake) destination of the ICMP message
  - Easy and very low bandwidth requirement.

ICMP Redirect Attack

- **ICMP Redirect Attack**
  - A "Redirect" ICMP message is sent to the target about a destination to be redirected to fake/wrong router.
  - The target will sent all packets toward the destination via the wrong router.
  - The wrong router can be
    - the attacker's host, which can capture, drop or modify and retransmit the traffic.
    - Any fake address.

MAC Level Attacks

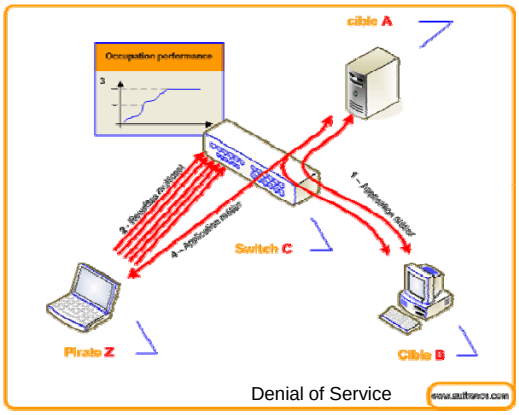
- MAC flooding
- Switch Saturation Attack
- ARP spoofing
- Spanning tree reconfiguration

MAC Flooding

- MAC flooding is a technique employed to slow traffic and compromise the security of network switches.
  - Switches maintain a translation table that maps individual MAC addresses on the network to the physical ports on the switch
  - A switch is flooded with packets, each containing different source MAC addresses. The intention is to consume the limited memory set in the switch.
  - The result of this attack causes the switch to enter a state in which all incoming packets are broadcast out on all ports, instead of just down the correct port as per normal operation
- The result of the attack may be:
  - A malicious user could then use a packet sniffer running in promiscuous mode to capture sensitive data from other computers
  - An increase of the collision rate, and thus in the network congestion and transmission delay
- Counter measures:
  - See ARP spoofing

Switch Saturation Attack

- Some similarity with MAC flooding, but
  - Uses multicast (*group*) frames which require more processing power
- Result
  - Switching performance is degraded



ARP Spoofing

- ARP spoofing (ARP poisoning) is a technique used to attack an local area network. It may allow an attacker to:
  - sniff data frames,
  - modify the traffic,
  - stop the traffic.
- The principle of ARP spoofing is to send fake ARP messages.
  - The aim is to associate the attacker's MAC address with the IP address of another node (such as the default gateway).
  - Any traffic meant for that IP address would be mistakenly sent to the attacker instead.
  - The attacker could then choose to forward the traffic to the actual default gateway (passive sniffing) or modify the data before forwarding it (man-in-the-middle attack).
  - The attacker could also launch a Denial of Service attack against a victim by associating a nonexistent MAC address to the IP address of the victim's default gateway.

ARP Spoofing : Countermeasures

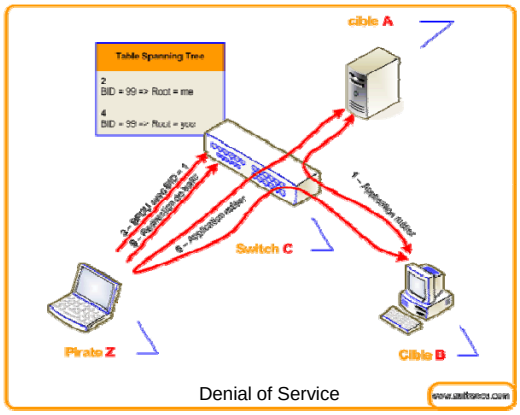
- Use of static, non-changing ARP entries:
  - The only method of completely *preventing* ARP spoofing
- Detections:
  - Listens for ARP replies on a network, and sends a notification (via email) when an ARP entry changes.
  - DHCP snooping
    - Only allow clients with specific IP/MAC addresses to have access to the network.
    - Improper ARP messages are dropped
    - It works with information from a DHCP server:
      - Track the physical location of hosts.
      - Ensure that hosts only use the IP addresses assigned to them.
      - Should ensure that only authorized DHCP servers are accessible.

LAN Countermeasures

- VLAN (IEEE 802.1Q)
- Traffic Filtering
  - Par port physique, @MAC, @IP, port number (TCP or UDP)
- Switch authentication
  - RFC 3580 (IEEE 802.1X) and RFC 3748 (EAP).
  - Based on Radius server, for instance

Spanning Tree Attack

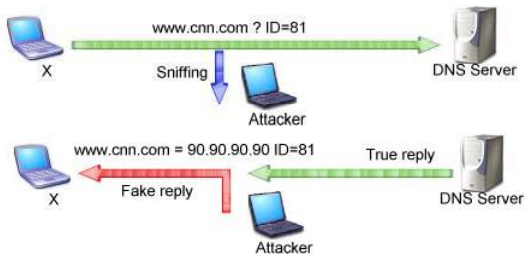
- The root of the spanning tree is elected. The lowest bidder wins.
  - The root can control the active port.
  - During LAN reconfiguration the traffic is interrupted



DNS Level Attacks

- DNS ID spoofing
- DNS cache poisoning

DNS ID Spoofing



- The attacker replies BEFORE the real DNS server
- In the example, the attacker runs a sniffer, intercepts the request and replies to his victim with the same ID number and with a reply of his choice
- Making the attack more accurate and efficient with
  - the Birthday Paradox
    - For 650 queries/fake replies, chances are about 0.960411
  - ARP cache poisoning

DNS Cache Poisoning

- DNS cache poisoning tricks a DNS server into believing it has received authentic information when, in reality, it has not.
  - Once the DNS server has been poisoned, the information is generally cached for a while, spreading the effect of the attack to the users of the server or indirectly by its downstream servers if applicable.
- To perform a cache poisoning attack, the attacker exploits a flaw in the DNS (Domain Name Server) software that can make it accept incorrect information.
  - If the server does not correctly validate DNS responses to ensure that they have come from an authoritative source, the server will end up caching the incorrect entries locally and serve them to users that make the same request.
- For example, an attacker poisons the IP address DNS entries for a target website on a given DNS server, replacing them with the IP address of a server he controls. Then he can:
  - sniff the service requests,
  - modify the traffic,
    - He creates fake entries for files on the server they control with names matching those on the target server. These files could contain malicious content, such as a worm or a virus. A user whose computer has referenced the poisoned DNS server would be tricked into thinking that the content comes from the target server and unknowingly download malicious content.
  - stop the traffic.

DNS Cache Poisoning

- To accomplish the attacks, the attacker must force the target DNS server to make a request for a domain controlled by one of the attacker's nameservers.
  - what are the address records for subdomain.example.com?
- Redirection of the target domain's nameserver
  - Assign an IP address specified by the attacker to the nameserver.
  - Example

```
DNS server's request:
   subdomain.example.com. IN A
Attacker's response:
      Answer:
      (no response)
      Authority section:
      example.com. 3600 IN NS ns.wikipedia.org.
      Additional section:
      ns.wikipedia.org. IN A w.x.y.z
```

  - A vulnerable server would cache the additional A-record (IP address) for ns.wikipedia.org, allowing the attacker to resolve queries to the entire wikipedia.org domain.
- Redirect the NS record of the target domain
  - Assign an IP address specified by the attacker to the nameserver of another domain unrelated to the original request
  - Example

```
DNS server's request:
   subdomain.example.com. IN A
Attacker's response:
      Answer:
      (no response)
      Authority section:
      wikipedia.org. 3600 IN NS ns.example.com.
      Additional section:
      ns.example.com. IN A w.x.y.z
```

  - A vulnerable server would cache the unrelated authority information for wikipedia.org's NS-record (nameserver entry), allowing the attacker to resolve queries to the entire wikipedia.org domain.

DNS Attacks: Countermeasures

- Countermeasures
  - DNS servers should ignore any DNS records which are not directly relevant to the query
  - Use of cryptographically-secure random numbers for selecting both the source port and the 16-bit nonce
  - DNSSEC, uses cryptographic electronic signatures signed with a trusted digital certificate to determine the authenticity of data.
    - DNSSEC can counter cache poisoning attacks, but as of 2006 is not widely deployed
  - Mitigated at the transport layer to perform end-to-end validation once a connection is set up to an endpoint.
    - A common example of this is the use of Transport Layer Security and digital signatures.

Router Level Attacks

- See Host attacks
  - Most involve either resource exhaustion or corruption of the router operating system runtime environment.
- Routing attacks
  - TCP SYN attack against BGP router
  - PIM join/leave flood attack against multicast router.
  - RIP redirection attacks
    - In routers, which implemented
      - the original RIP v1 (no router authentication)
      - the default setting of v2 (clear password).
    - The fake router claims to have a better route

Email flooding

- Email flooding, email bombing
  - Sending huge volumes of emails to a single user at any one time.
  - Spam is a major source of irritation. This is because the huge volumes of junk take a lot of time to be sifted through to be deleted in order to ensure that real useful emails are not deleted.
  - Modern computers have enough power and broadband networks provide enough bandwidth to allow sending of many emails at a time. Multiple parallel threads send the same email over and over again to the entered email address.
- Counter-measures
  - Black list, gray list, white list, etc
  - Anti spam tool
- Drawbacks :
  - List management
  - Spam blocking tools could trap legitimate emails

Some Email bombers

- Partial list of email bombers
  - Aenima
  - Avalanche
  - Euthanasia
  - Gatemail
  - Ghost Mail
  - HakTek
  - Kaboom
  - Serpent
  - The Unabomber
  - Mailbomber
  - Up Yours
  - Windows Email Bomber

Worming

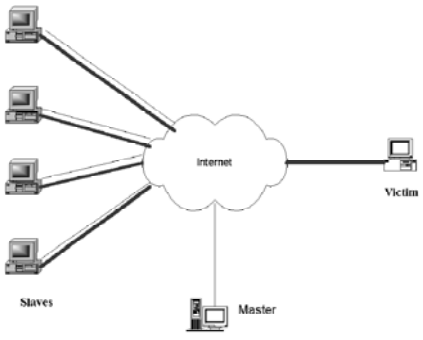
- The worm sends a large amount of data to remote servers. It then verifies that a connection is active by attempting to contact a website outside the network. If successful, an attack is initiated. This would be in conjunction with a mass-mailing of some sort.

Worm example

- Worm Attack on SSL Vulnerability
  - The vulnerability is in an older Microsoft protocol called PCT (Protected Communications Transport).
  - Microsoft's SSL library contains a buffer overrun flaw that enables attackers to run arbitrary code on vulnerable machines by sending specially designed PCT handshake packets. PCT is included in the SSL library, which is present in a number of products, including IIS and Exchange Server.
  - Microsoft Corp. has released a patch.
  - April 2004

Distributed Denial of Service Attacks (DDOS)

- Attacker logs into Master and signals Slaves to launch an attack on a specific target address (Victim).
- Slaves then respond by initiating TCP, UDP, ICMP, IP or any other DOS attack on victim.



DDoS Definition

- A Distributed Denial of Service (DDoS) is an attack on a network which is designed to bring it to a halt.
  - This is done by sending useless traffic to a specific service/port on a server.
  - The amount of traffic sent would overwhelm the service, so that legitimate traffic would be dropped or ignored.

Names of DDoS

- Trin00 (WinTrinoo)
- Tribe Flood Network (TFN) (TFN2k)
- Shaft
- Stacheldraht
- Mstream

- Use previous DoS, for instance :
  - Smurf attack
  - ICMP flood
  - SYN flood
  - UDP flood
  - All at once

Trin00

- Affects Windows and many Unix OS
- Attacker scans for exploits, gains root, and downloads Trin00 programs.
- Attacker->Master->Daemon…->Target hierarchy
  - (One -> More -> Many==>One)
- Attacker can telnet into a Master to initiate commands, which are distributed amongst its Daemons.

Trin00

- Communication between Master->Daemon through a password-protected cleartext UDP-based protocol.
- Daemons attack the target with a UDP or TCP packet bombardment.
- Used in the February 2000 attacks on eBay, Amazon, CNN, etc.

Example of real DDoS

```
4081 0.224610 119.226.89.96 -> poor.student.1.83 TCP 33081 > 60785 [SYN]
     Seq=3693150756 Ack=0 Win=32768 Len=0
4082 0.224610 poor.student.1.83 -> 223.144.66.65 TCP 52284 > 19586 [RST, ACK]
     Seq=0 Ack=423694111 Win=0 Len=0
4083 0.224610 3.41.60.116 -> poor.student.2.231 TCP 5594 > 40940 [SYN]
     Seq=2132997225 Ack=0 Win=32768 Len=0
4084 0.224610 poor.student.1.83 -> 50.180.94.71 TCP 33289 > 11952 [RST, ACK]
     Seq=0 Ack=1790973261 Win=0 Len=0
4085 0.224610 244.214.39.108 -> poor.student.2.231 TCP 38802 > 23759 [SYN]
     Seq=747020069 Ack=0 Win=32768 Len=0
4086 0.224610 poor.student.1.83 -> 198.183.172.81 TCP 57223 > 43146 [RST, ACK]
     Seq=0 Ack=3749566807 Win=0 Len=0
4087 0.224610 64.81.138.119 -> poor.student.1.83 UDP Source port: 1026
     Destination port: 24661
4088 0.224610 poor.student.2.231 -> 96.247.9.94 TCP 48931 > 50749 [RST, ACK]
     Seq=0 Ack=1188357973 Win=0 Len=0
4089 0.224610 103.227.64.42 -> poor.student.1.83 TCP 45715 > 63366 []
     Seq=3389528594 Ack=0 Win=16384 Len=0
4090 0.224610 poor.student.1.83 -> 211.107.218.23 TCP 12666 > 48183 [RST, ACK]
     Seq=0 Ack=2803931407 Win=0 Len=0
4091 0.224610 87.29.46.64 -> poor.student.1.83 TCP 17092 > 47365 [SYN]
     Seq=3446572548 Ack=0 Win=32768 Len=0
4092 0.224610 poor.student.1.83 -> 58.24.148.57 TCP 26667 > 9797 [RST, ACK]
     Seq=0 Ack=3710546447 Win=0 Len=0
4093 0.224610 8.116.40.43 -> poor.student.1.83 TCP 38367 > 32889 [SYN]
     Seq=1914703987 Ack=0 Win=32768 Len=0
4094 0.225448 poor.student.1.83 -> 68.132.173.125 TCP 64470 > 35524 [RST, ACK]
     Seq=0 Ack=1819819023 Win=0 Len=0
4095 0.225448 75.115.186.26 -> poor.student.1.83 TCP 4082 > 29772 [SYN]
     Seq=4245878839 Ack=0 Win=32768 Len=0
```


DDOS: some Countermeasures

- RID:
  - Sends out packets and listens for reply
  - Detects Trinoo, TFN, Stacheldraht
- find_ddos tool (from NIPC)
  - Runs on local system
  - Detects Trinoo, TFN, TFN2k
- Bindview's Zombie Zapper
  - Tells DDOS slave to stop flooding traffic

Conclusion

- DDoS attacks are very difficult to trace and stop.
- Many automated tools are available
  - Stacheldraht, Trinoo, TFN2K, Smurf, Fraggle, etc.
- New hardware appliances are being manufactured specifically for these types of attacks.
- Many dedicated server providers simply unplug the server that is being attacked until the attack has stopped. This is not a solution this is a careless and temporary fix. The culprit will still exist and has not been held accountable for their actions.
- Once an attack is detected hosts should immediately engage their upstream providers.

Some Solutions

- Firewall
- Rate limiter
- Cookie
- Security administration

Firewall

- **Firewalls**
  - Allow or deny protocols, ports or IP addresses.
  - Some DoS attacks are too complex for firewalls,
    - e.g. if there is an attack on port 80 (web service), firewalls cannot prevent that attack because they cannot distinguish good traffic from DoS attack traffic. Idem with 3-phase handshake TCP connection.
  - Modern firewalls are statefull
- Egress filtering
  - Examination of all packet headers leaving a subnet for address validity.
    - If the packet's source IP address originates inside the subnet that the router serves, then the packet is forwarded.
    - If not (the packet has an illegal source address) then the packet is simply dropped.
    - There is very little overhead involved, therefore there is no degradation to network performance.
  - You are working for the mankind. You are hoping that others are making the same for you.

Flow Analysis and Traffic Limitation

- Dedicated to mitigation technique
- Abnormal traffic detection
  - Port scanning
  - High increase in data volume coming from a source

Rate Limiter

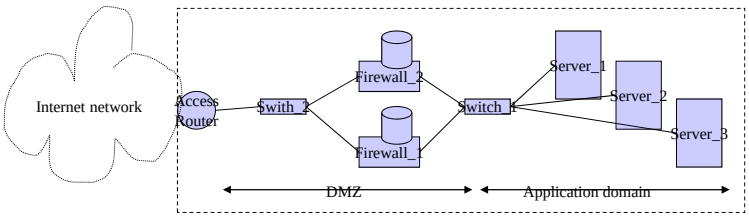
- Rate Limiting
  - Limit the response rate to specific requests
    - E.g. to TCP SYNs
  - Time out control of the connections
  - Queuing techniques
    - Class-based
  - Implemented into Switches or Routers

Cookies

- TCP SYN cookies
  - Reply with a SYN/ACK packet with a particular sequence number
    - hash of source IP, port number and time, for instance
  - SYN cookies modify the TCP protocol handling of the server by delaying allocation of resources until the client response (and address) has been verified

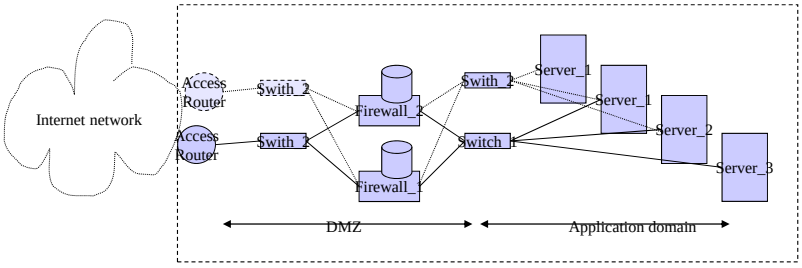
Firewall Architecture

- Multi-firewall can be required to have
  - Sufficient processing power for in-deep analysis of the traffic
  - State-full firewalls
- Request and the associated responses should passed through the same firewall on their way in and out



Redundancy Architecture

- Redundancy is required to have
  - Sufficient processing power for DoS resilience
  - Fault resilience



CERT

- World wide monitoring
- Computer Security Incident Response Teams (CSIRTs)
  - addresses risks at the software, system and network level
  - receives, reviews and responds to computer security incident reports and activity
  - focuses on identifying and addressing existing and potential threats
  - notifying system administrators of these threats, and coordinating with vendors and other CSIRTs
  - CERT/CC:
    - Computer Emergency Response Team/coordination center
  - *CERTA* est le CSIRT dédié au secteur de l'administration française
    - http://www.certa.ssi.gouv.fr/

Framework against DDoS

- Prevention
- Detection
- Traceback
- Mitigation
- Post-analysis

DoS Prevention

- At client's level: decrease system's vulnerabilities
  - Take into account security at service deployment
  - Regularly update systems and apply patches
  - Conduct security audit
- At network level: implement control mechanisms
  - Anti-spoofing, filtering
  - Access control, authentication

DoS Detection

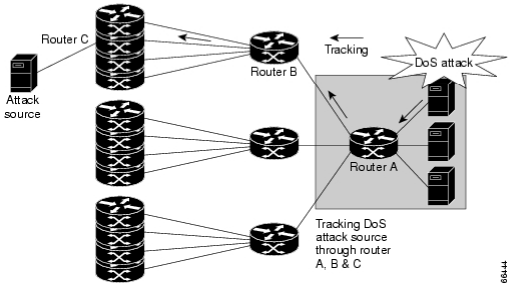
- Notification to the client as early as possible
- Signature detection
  - Usual attacking tools are recognized using signatures
    - For instance: Stacheldraht, etc.
  - Detects botnet creation (source), and DoS attacks (client)
- Anomaly detection
  - Flow modeling and deduction of traffic profiles
  - Detects even "unknown" attacks
  - At several locations: client (victim), backbone, peering points
- Future detection
  - Distributed,
  - High speed analysis,
  - Automatic creation and update of signatures and models

DoS Traceback

- Packets counting
  - Flow monitoring (e.g. SNMP, netflow, etc.)
- IP source back tracker
  - gathers information about the traffic that is flowing to a host that is suspected of being under attack. This feature easily traces back an attack to its entry point into the network.
- Packet marking solution
  - Sample packets are marked, and their transits are collected on some routers
  - Path of marked packets can be reconstructed
- Backscatter approach
  - Spoofed addresses generate Unreachable destination ICMP packets
  - Operators can detect the existence and the target of massive IP spoofing attacks
    - The source address of the packet which have generate the Un. Dest. ICMP packet !

Backtracker

- Counteracting a DoS attack involves intrusion detection, source tracking, and blocking.
- Source backtracking requires packet analysis, filtering and counting per ingress port
- To block attacks, committed access rate (CAR) and access control lists (ACLs) are used



DDoS mitigation

- Filtering
  - Firewalls, ACR and ACLs, advanced BGP filtering (network operators only)
- Resources limitation
  - Prioritize legitimate flows
- Hiding and re-configuration
  - Moving targets (change IP, domain name, etc.)
  - Use a Content Delivery Network
- Derivation of traffic targeted to the victims
  - Done by operators
  - Destruction of the traffic
  - Traffic is analyzed before destruction
  - Traffic laundry: traffic is cleaned (some are destroyed) and shaped (delayed) then re-injected

DDoS Post-mortem Analysis

- Analysis of collected attack traces
  - Extraction of new signatures to update IDS database
  - Update of behavior-based traffic models
  - Identification of vulnerabilities
- Report to the client
- Analysis of the security policy
  - Identification of critical points
  - Enhancement of procedures

Future Issues

- Future issues
  - Pulsing and cycling attacks
  - Simulation of legitimate traffic
  - Random request generation, if encryption is generalized

Bibliography

- Hervé Sibert. "Etat de l'art sur le DDoS". France Telecom, 2005
- Jelena Mirkovic, Sven Dietrich, David Dittrich, and Peter Reiher. "Internet Denial of Service. Attack and Defense Mechanisms". Prentice Hall 2005.
- P. Ferguson, D. Senie. "Network Ingress Filtering: Defeating Denial of Service Attacks which employ IP Source Address Spoofing", May 2000
- CERT. http.//www.cert.org
- CERTA. http://www.certa.ssi.gouv.fr/
- Un site français sur la sécurité des réseaux informatiques : http://www.uthsecu.com