ROYAL INSTITUTE OF TECHNOLOGY

Networked System Security

# (D)DoS Attacks and IDS

Module TAs:
**Syed Zubair and Hongyu Jin**

**Panos Papadimitratos**
**Networked Systems Security Group**
www.ee.kth.se/nss

## Outline

- (D)DoS attacks
  - Types of DoS
  - LAND attack
  - Smurf Flooding, SYN Flooding
  - DNS,NTP amplification attacks
  - SCTP (Recap)
- Intrusion Detection Systems
  - IDS Design policies
  - IDS Architecture
  - Attacks against IDS
    - Insertion vs. Evasion



## (D)Denial of Service attacks

- DoS attacks
  - Attack on the availability; aim at making the network unavailable to users or available with poor service
  - Distributed DoS deploys multiple machines to attain this Goal
  - Operates on the basis of work asymmetry; more expensive for the victim than the attacker
- Why is DoS possible
  - Security: Highly Interdependent
  - Power of Many Vs Power of one (or few)
  - Core protocols designed for Functionality (not Security)

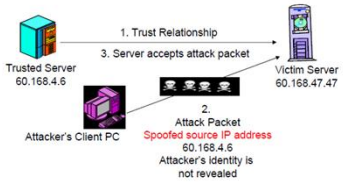


## Types of DoS attacks

- Protocol Implementations Bugs
  - Send Unusual traffic to application: Crash a host with a single attack packets
  - Examples: Ping-of-Death, LAND, Teardrop etc
  - New single message attacks appearing because developers rarely test their software for unusual patterns
- Resource/Bandwidth Exhaustion Attacks
  - Reflection and Amplification
  - SYN Flooding (try to open many connections with SYN segments)
  - Smurf Attack (ping a range of IPs with victimõs IP as source )

## IP address Spoofing

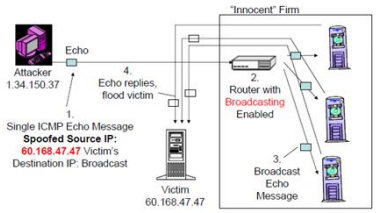
- IP address spoofing: Sending a message with a false IP address
  - Gives sender anonymity so that attacker cannot be identified
  - Can exploit trust between hosts if spoofed IP address is that of a host the victim trusts
  - Reflection
- LAND attack: send victim a packet with victimôs IP address in both source and dest address fields and the same port number for the source and destination




  - In 1997, many computers, switches, routers, and even printers, crashed when they received such a packets
  - Show how unexpected combination of parameters can create problems



## Smurf Flooding Attack

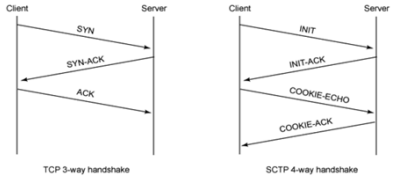



- Attacker sends tailored ICMP messages to the broadcast address of a network
  - Internet Control Message Protocol (ICMP) is for supervisory message at the Internet layer (network analysis and error messages)
  - Source IP of the packet is the IP of the victim
  - Receiving hosts will issue an ICMP reply and flood the victim



## SYN Flooding

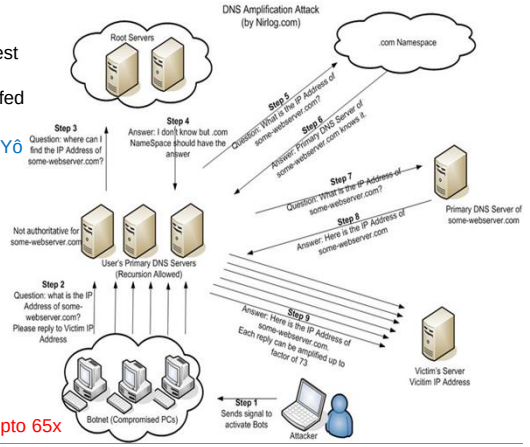
- Attacker sends flood of SYN segments
  - Victim allocates resources, sends SYN/ACK and waits for ACK
  - Connection table full: Victim cannot respond to new requests
  - Allocation of resources without client address authentication
  - SCTP more resilient






## DNS Amplification Attack

- Attacker
  - Sends DNS request to public DNS servers with spoofed source IP
  - Query of type ÁANYÔ
  - Returns all known info about a DNS zone
  - Amplification!!
  - 60 B--> 4KB




Amplification factor upto 65x

## NTP Amplification Attack

- Network Time Protocol
  - UDP based
  - MONLIST (or MON_GETLIST): Returns addresses of last 600 machines that NTP server has interacted with
  - Amplification (Response>>Request)
  - Combine this with Reflection
  - Distribute this to Bots

Amplification factor 19x

ntpdc ï c monlist 1xx.xxx.xxx.xx9

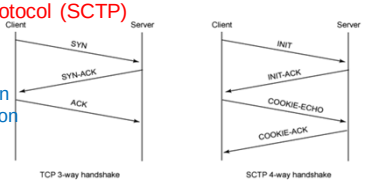
| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 665 | *REF* | 10.114.1.118 | 1[redacted]9 | NTP | 234 | NTP Version 2, private |
| 666 | 0.144916000 | 1[redacted]9 | 10.114.1.118 | NTP | 482 | NTP Version 2, private |
| 667 | 0.146839000 | 1[redacted]9 | 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 |

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



## SCTP

- Stream Control Transmission Protocol (SCTP)
  - 4-way handshake
  - Use of state cookies
  - Avoids retaining state information prior to client existence verification




- Cookie baking
  - Cookie carries TCB (Transmission Control Block) data
  - Creation Time
  - Lifetime
  - MAC is also calculated over the entire cookie and included
- Spoofing resilience
  - Client must return a valid cookie for connection to establish

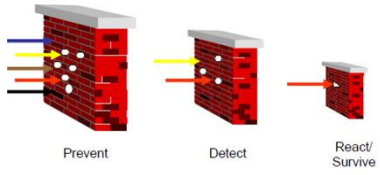


## Intrusion Detection Systems



## Why is it necessary

- Inevitably the best intrusion prevention system will fail. The next best line of defense is intrusion detection
  - If an intrusion is detected quickly enough, the intruder can be identified and stopped
  - An effective can serve as a deterrent
  - Enables the collection of information about intrusion techniques



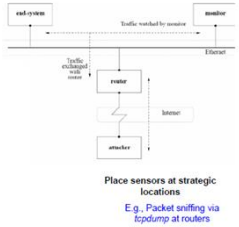
## Design Process

- Intrusion Detection Approaches
  - Statistical anomaly detection involves the collection of data relating to the behavior of legitimate users over a period of time (high false positive rate)
  - Rule-based detection attempts to define a set of rules that can be used to decide whether a given behavior is that of an intruder
- Intrusion Detection Policies
  - Misuse (or signature-based) detection
    - Observed behavior is compared with known attack patterns
  - Anomaly detection
    - Flags as intrusion attempts any activities varying from normal behavior
    - *What is normal behavior?*
- Intrusion Detection Architectures
  - Host-based (HIDS) vs. Network-based (NIDS)



## Network IDS

- NIDS are based on interpretation of raw network packets
  - Usually watch network passively and capture packets transmitted by other machines
  - Watch for violation of protocols and unusual connection patterns
  - Look into the data portions of the packets for malicious command sequences
- What information is relevant to IDS?
  - Names of the hosts being queried--responses
  - Contents of all TCP connections
- Logical target of attacks
  - Each components is point of vulnerability
- Possible attacks on their
  - Availability, Accuracy and Completeness
- Need to be reliable and robust





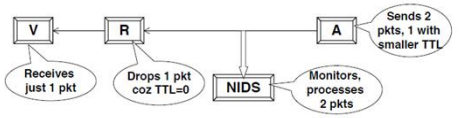

## Problems with NIDS

- Passive network monitors
  - Inherently ñfail-openò
  - Cease to provide protection when subverted
- Vulnerability to DoS
  - Process all flows to all protected end-systems
  - Being complex systems require lots of resources
- Insufficient information on the ñwireò
  - Not enough to correctly reconstruct the state of complex protocol transactions
- Diversity in protocol implementations
  - Packet processing differs across end-systems (ambiguous interpretations)
- Unknown internal network conditions
  - Topology, Router configs, Traffic congestion, etc.



## Eluding NIDS

- What the IDS sees may not be what the end system gets. This can lead to various types of attacks
  - Insertion and evasion attacks
  - IDS needs to perform full reassembly of the packets (BUT?)
- Insertion attack
  - NIDS accepts a packet that an end-system rejects or doesnô even receive
  - Data gets inserted into the NIDSôs packet stream



## Insertion Attack

- Attacker's Data Stream

| Seq# | 2 | 3 | 3 | 5 | 4 | 1 | 6 |
|---|---|---|---|---|---|---|---|
| Data | T | T | X | C | A | A | K |

- NIDS's Stream

Accepts 3rd packet which overwrites 2nd packet data
Interprets "**ATXACK**"

| Seq# | 1 | 2 | 3 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Data | A | T | T | X | A | C | K |

- End-System's Stream

Rejects 3rd packet for some reason, or does not receive it
Interprets "**ATTACK**"

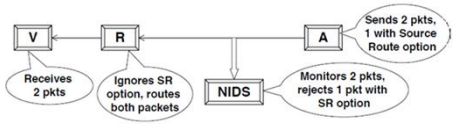
| Seq# | 1 | 2 | 3 | ~~3~~ | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Data | A | T | T | ~~X~~ | A | C | K |

- Occurs when NIDS is less strict in processing packets than internal network



## Evasion

- An end-system can accept a packet that a NIDS rejects
  - Data gets "slipped" past the NIDS




- Attacker's Data Stream

| Seq# | 2 | 3 | 3 | 5 | 4 | 1 | 6 |
|---|---|---|---|---|---|---|---|
| Data | T | X | T | C | A | A | K |

- NIDS's Stream

Rejects 3rd packet for some reason
Interprets "**ATXACK**"

| Seq# | 1 | 2 | 3 | ~~3~~ | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Data | A | T | X | ~~T~~ | A | C | K |

- End-System's Stream

Accepts 3rd packet which overwrites 2nd packet
Interprets "**ATTACK**"

| Seq# | 1 | 2 | 3 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| Data | A | T | X | T | A | C | K |

- Occurs when NIDS is more strict in processing packets than internal network



## Real Insertion/Evasion Attacks

- Mostly leverage on basic network and protocol ambiguities at NIDS
  - Ambiguous interpretation of header fields
  - Ambiguous handling of header options
  - Ambiguous fragment/segment reassembly
- Ambiguities can cause NIDS to accept/reject packets differently than the end-system

| Related Field | Ambiguity (Decision problem for NIDS) |
|---|---|
| TTL | Will the packet reach the end-system before TTL becomes 0? |
| Length, DF | Will all downstream links be able to transmit this big packet without fragmenting (DF bit set)? |
| IP Option(s) | Will the end-system/routers accept packet with this IP option(s)? E.g. (Strict) Source Route option |
| TCP option(s) | Will the end-system accept packet with this TCP option(s)? |
| Data | Will the end-system accept data in SYN packet? |
| ToS | Does the packet conform to all internal routers (DiffServ)? |
| IP Frag Offset | How will the end-system reassemble overlapping fragments? |
| TCP Seq No. | How will the end-system reassemble overlapping segments? |