Network Security

Chapter 10
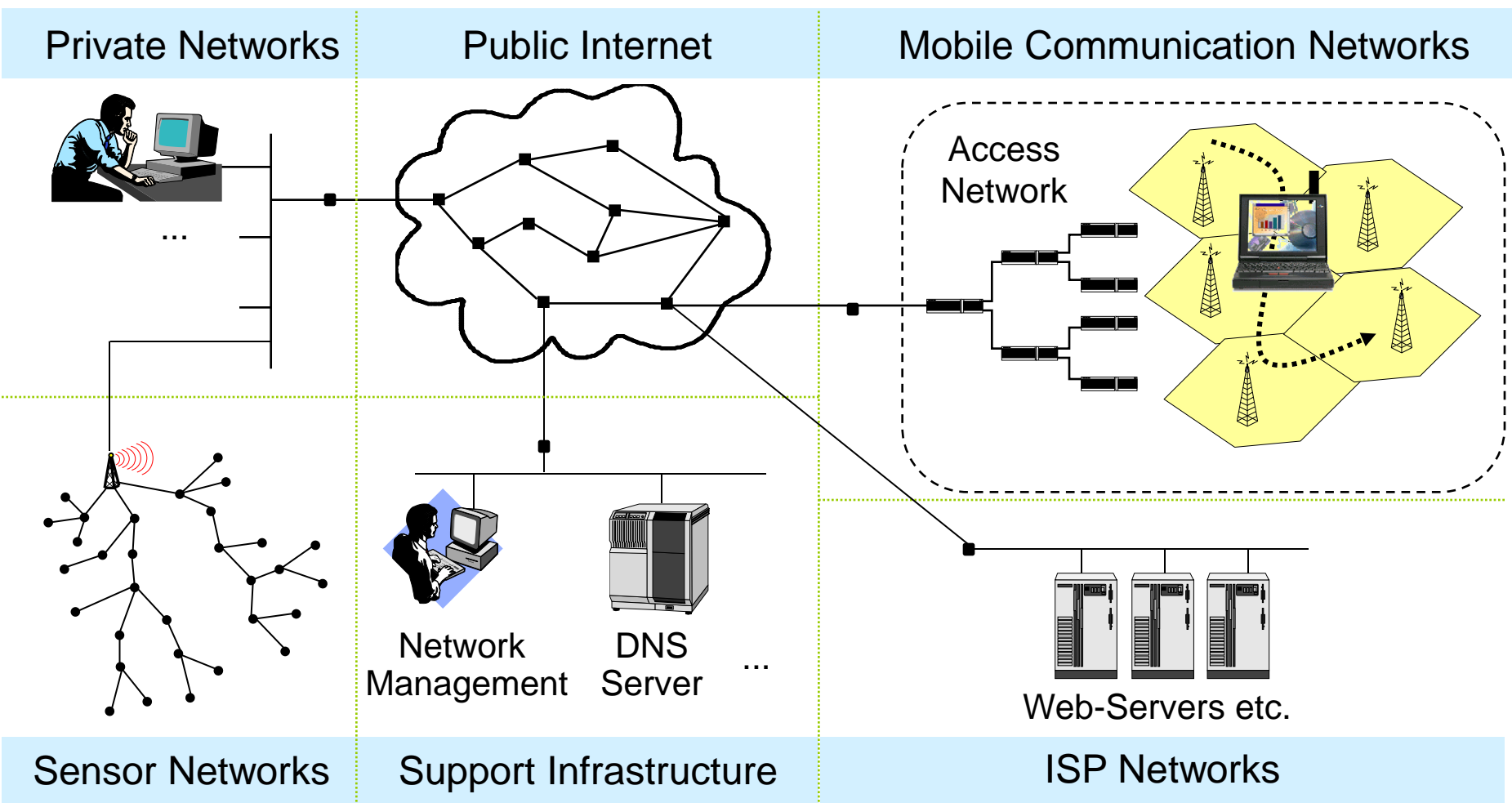
System Vulnerabilities and Denial of Service Attacks

A High Level Model for Internet-Based IT-Infrastructure

Private Networks

Public Internet

Mobile Communication Networks

Access Network

Sensor Networks

Support Infrastructure

Network Management

DNS Server

...

ISP Networks

Web-Servers etc.

Denial of Service

❑ What is Denial of Service?

  ▪ *Denial of Service (DoS) attacks aim at denying or degrading legitimate users' access to a service or network resource, or at bringing down the servers offering such services*

❑ Motivations for launching DoS attacks:

  ▪ Hacking (just for fun, by "script kiddies", ...)

  ▪ Gaining information leap ($\rightarrow$ 1997 attack on bureau of labor statistics server; was possibly launched as unemployment information has implications to the stock market)

  ▪ Discrediting an organization operating a system (i.e. web server)

  ▪ Revenge (personal, against a company, ...)

  ▪ Political reasons ("information warfare")
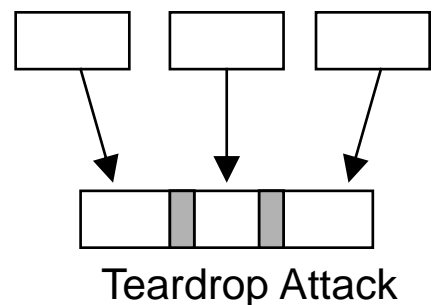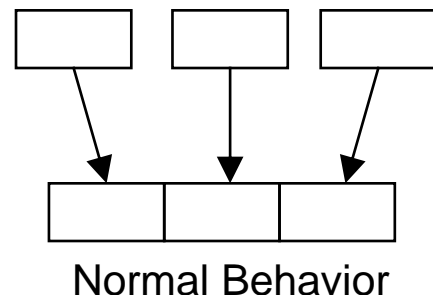
  ▪ ...

Denial of Service Attacking Techniques

❑ *Resource destruction* (disabling services):

- Hacking into systems
- Making use of implementation weaknesses as buffer overflow
- Deviation from proper protocol execution

❑ *Resource depletion* by causing:

- Storage of (useless) state information
- High traffic load (requires high overall bandwidth from attacker)
- Expensive computations ("expensive cryptography"!)
- Resource reservations that are never used (e.g. bandwidth)

❑ Origin of malicious traffic:

- Genuineness of source addresses: either genuine or forged
- Number of sources:
  - single source, or
  - multiple sources *(Distributed DoS, DDoS)*

Examples: Resource Destruction

❑ Ping-of-Death:
- Maximum size of TCP/IP packet is 65536 bytes
- Oversized packet may crash, freeze, reboot system

❑ Teardrop:
- Fragmented packets are reassembled using the Offset field.
- Overlapping Offset fields might cause system to crash.

Normal Behavior

Teardrop Attack

❑ Two main reasons make ICMP particular interesting for attackers:

- It may be addressed to broadcast addresses
- Routers respond to it

❑ The *Smurf* attack - ICMP echo request to broadcast:

- An attacker sends an ICMP echo request to a broadcast address with the source addressed forged to refer to the victim
    - local broadcast: 255.255.255.255;
    - directed broadcast: (191.128.0.0/24) 191.128.0.255
- Routers (often) allow ICMP echo requests to broadcast addresses
- All devices in the addressed network respond to the packet
- The victim is flooded with replies to the echo request
- With this technique, the network being abused as an (unaware) attack amplifier is also called a *reflector network:*
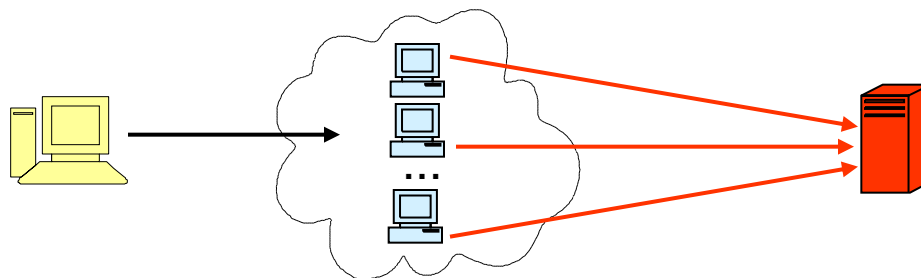
Resource Depletion Example 2: TCP-SYN Flood

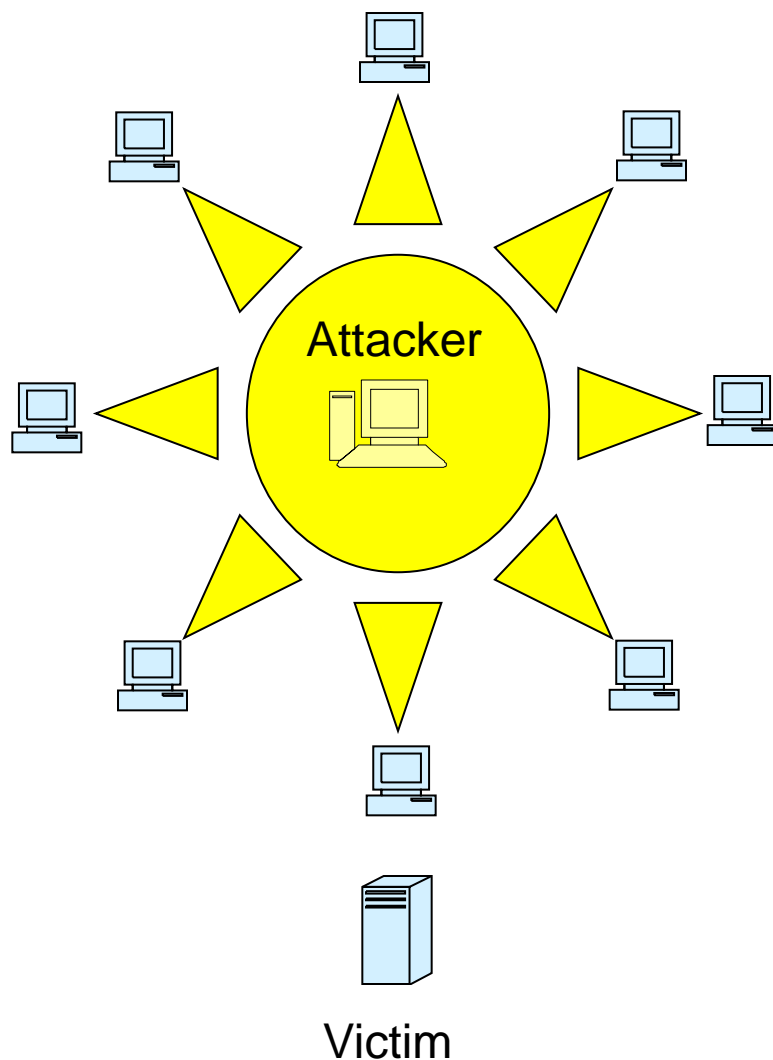❑ Category *Storage of useless state information:*
  ▪ Here: TCP-SYN flood attack
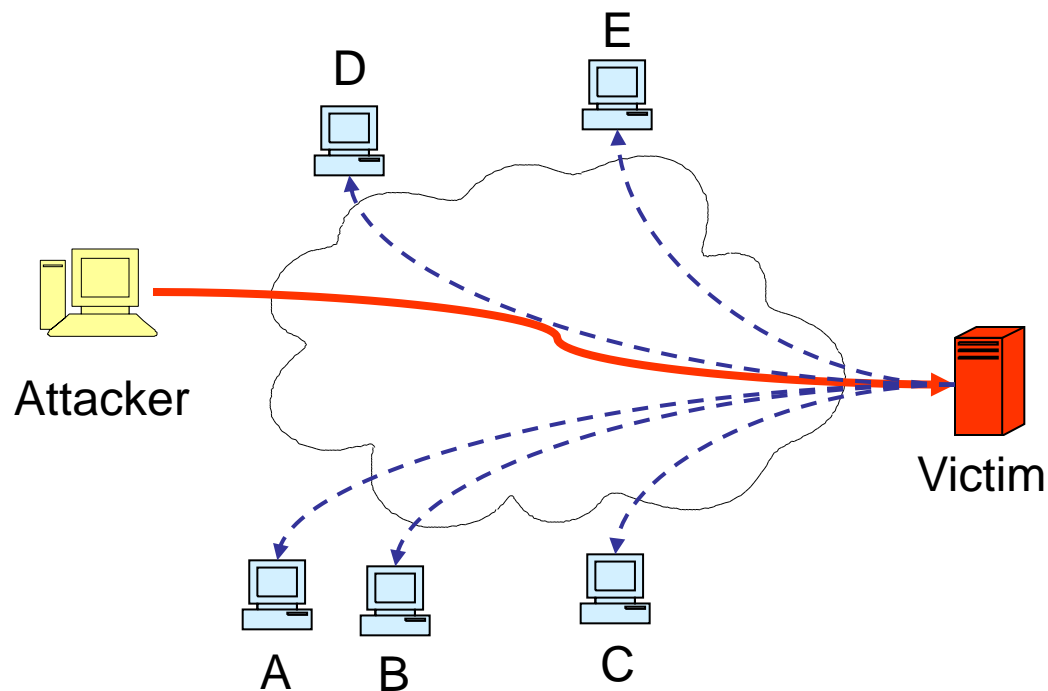
→ TCP SYN packets with forged source addresses ("SYN Flood")

- - → TCP SYN ACK packet to assumed initiator ("Backscatter")

Attacker

Victim

- ❑ Category *Overwhelming the victim with traffic*

- ❑ Attacker intrudes multiple systems by exploiting known flaws

- ❑ Attacker installs DoS-software:
  - ▪ „Root Kits" are used to hide the existence of this software

- ❑ DoS-software is used for:
  - ▪ Exchange of control commands
  - ▪ Launching an attack
  - ▪ Coordinating the attack

Attacker

Masters

Slaves

Victim

- - - -> Control Traffic ⟶ Attack Traffic
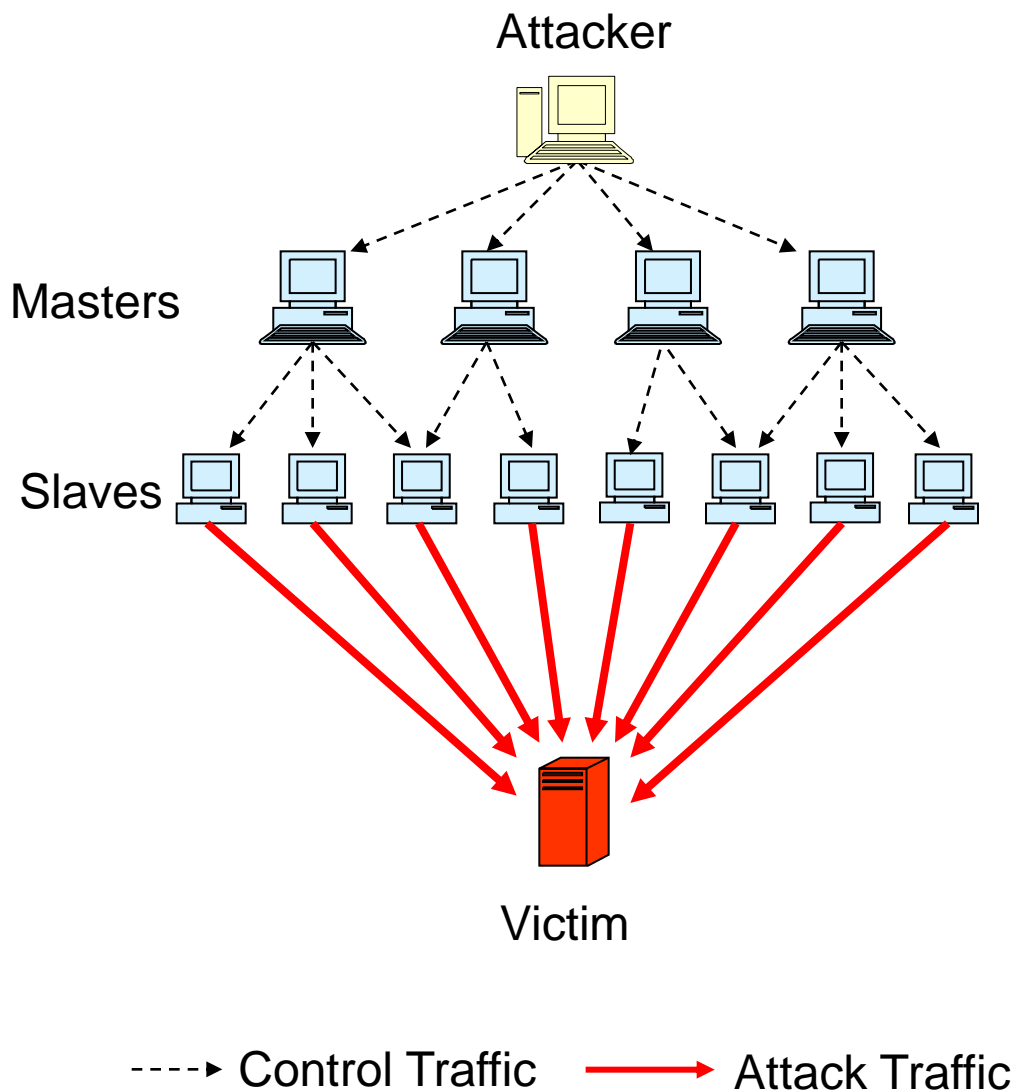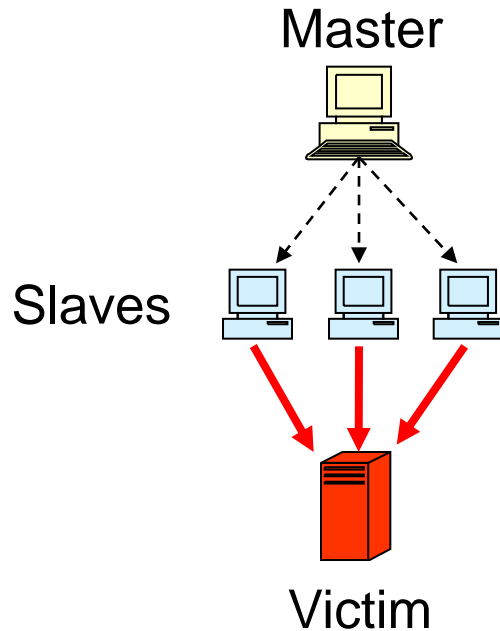
❑ The attacker classifies the compromised systems in:
- Master systems
- Slave systems

❑ Master systems:
- Receive command data from attacker
- Control the slaves

❑ Slave systems:
- Launch the proper attack against the victim

❑ During the attack there is no traffic from the attacker

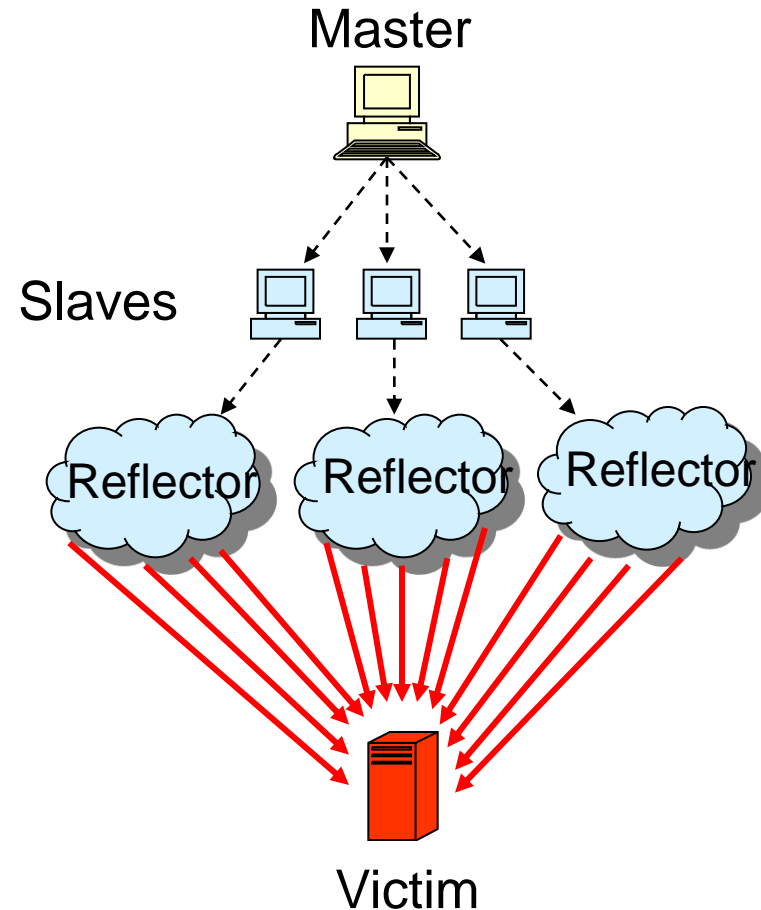❑ Different Attack Network Topologies

a.) Master-Slave-Victim   b.) Master-Slave-Reflector-Victim

Resource Depletion with CPU Exhaustion

❑ Category *CPU exhaustion by causing expensive computations:*

  ▪ Here: attacking with bogus authentication attempts

Attacker   Victim

attacker requests for
connection with server
→

server asks 'client' for
authentication
←

attacker sends false digital signature, server
wastes resources verifying false signature
→
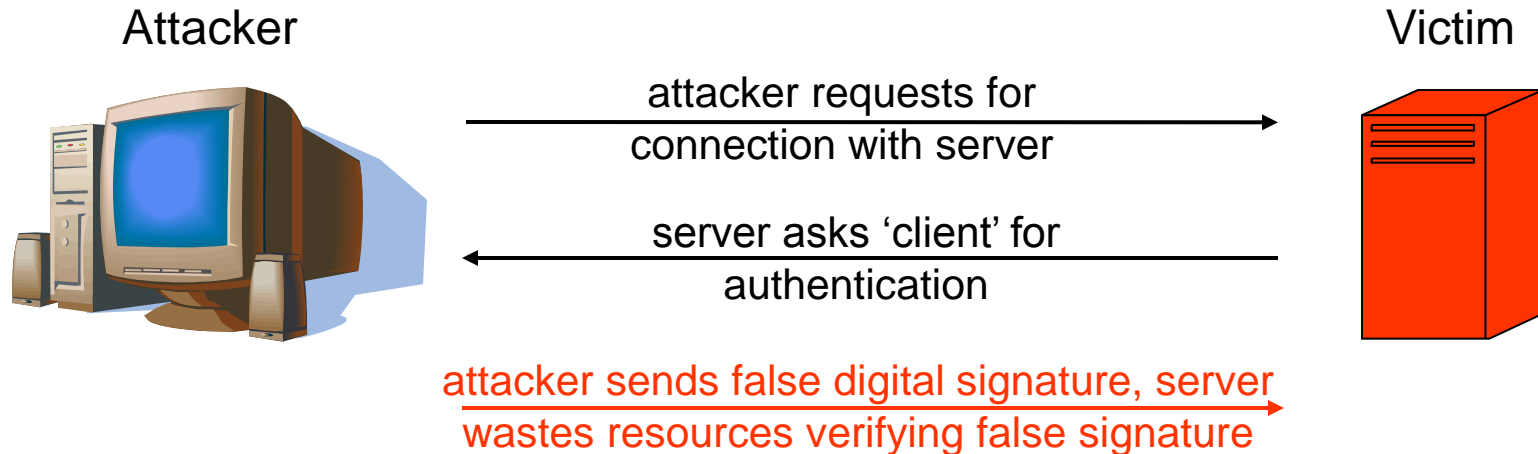
  ▪ The attacker usually either needs to receive or guess some values of the second message, that have to be included in the third message for the attack to be successful

  ▪ Also, the attacker, must trick the victim *repeatedly* to perform the expensive computation in order to cause significant damage

➡ Be aware of DoS-Risks when introducing security functions into protocols!!!

❑ Introduction and Threat Overview

❑ Denial of Service Threats

❑ DoS Attacks: Classification

❑ System Vulnerabilities

❑ Honeypots

❑ Upcoming Challenges

❑ Classification by exploited vulnerability

- ▪ Software vulnerability attacks
- ▪ Protocol attacks
- ▪ Brute-Force / flooding attacks

❑ Classification by attack rate dynamics:

- ▪ Continues rate
- ▪ Variable rate:
  - • Increasing
  - • Fluctuating

❑ Classification by impact:

- ▪ Disruptive
- ▪ Degrading

❑ Based on the vulnerability that is targeted during an attack, DoS attacks can be classified into:

- Software vulnerability attacks
- Protocol attacks
- Brute-Force / flooding attacks

❑ Some attacks can be classified into more than one of these categories. (see below)

❑ Software vulnerability attacks:

▪ Here, software bugs are exploited.

▪ Examples:

- Cisco 7xx attack: Some Cisco 7xx routers were crashed by connecting with "Telnet" and typing a very long password

  $\Rightarrow$ a password buffer overflow.

- Ping-of-Death
- Teardrop

- ❑ Protocol Attacks
  - ▪ Exploits a specific feature or implementation bug of the protocol.
  - ▪ Examples include:
    - • TCP SYN flood attacks
    - • Authentication server attacks
    - • Ping-of-death
    - • Teardrop

- ❑ Brute-force Attacks / Flooding attacks:
  - ▪ The victim is overwhelmed with a vast amount of seemingly legitimate transactions.
  - ▪ Brute-force attacks are further classified into two sub-categories:
    (see also next slide for more details)
    - • Filterable attacks
    - • Non-filterable attacks

❑ Filterable attacks:

- ▪ The flood packets are not critical for the service offered by the victim, and therefore can be filtered.
- ▪ Example: UDP flood or ICMP request flood on a web server.

❑ Non-filterable attacks:

- ▪ The flood packets request legitimate services from the victim.
- ▪ Examples include:
  - • HTTP request flood targeting a Web server
  - • CGI request flood
  - • DNS request flood targeting a name server
- ▪ Filtering all the packets would be an immediate DoS attack to both attackers and legitimate users.

❑ The victim might mitigate the effect of protocol attacks, by modifying the deployed protocol.

❑ However, the victim is helpless against brute-force attacks if they use legitimate services.

Classification of DoS Attacks by Attack Rate Dynamics

❑ Based on the attack rate dynamics that is targeted during an attack, DoS attacks can be classified into:

- Continuous Rate Attacks
- Variable Rate Attacks

❑ Continuous Rate Attacks:

- The most frequent kind of attack
- When the attack is launched, agent machines generate attack packets with a large constant rate.
- The sudden packet flood disrupts the victim's services quickly.
- The attack may be noticed quickly.

❑ Variable Rate Attacks:

- Vary the attack rate to avoid detection
- The attack rate might be increasing over a long time or even fluctuating, which makes detection even harder.

Classification by Impact

- Disruptive:
    - The goal is to fully deny the victim's service to its clients
    - The most common category of attacks
- Degrading:
    - A portion of the victim's resources (e.g. 30%) are occupied by the attackers.
    - Can remain undetected for a signification time period
    - Customers experience slow response times or now service during high load periods.
    - Customers go to an other Service Provider.

System Vulnerabilities: Basic Attacking Styles

❑ Origin of attacks:

- Remote attacks: attacker breaks into a machine connected to same network, usually through flaw in software

- Local attacks: malicious user gains additional privileges on a machine (usually administrative)

❑ Main attacking techniques:

- *Buffer overflow:*
  - Intentional manipulation of program state by causing an area of memory to be written beyond its allocated limits

- *Race condition:*
  - Exploiting non-atomic execution of a series of commands by inserting actions that were "unforeseen" by the programmer

- *Exploiting trust in program input / environment:*
  - It is often possible to maliciously craft input / environment variables to have deleterious side effects
  - Programmers are often unaware of this

Identifying Vulnerable Systems with Port Scans (1)

❑ Background

  ▪ Identification of vulnerable systems / applications in order to identify systems to compromise

  ▪ Automated distribution of worms

❑ Scan types

  ▪ Vertical scan: sequential or random scan of multiple (5 or more) ports of a single IP address from the same source during a one hour period

  ▪ Horizontal scan: scan of several machines (5 or more) in a subnet at the same target port from the same source during a one hour period

  ▪ Coordinated scan: scans from multiple sources (5 or more) aimed at a particular port of destinations in the same /24 subnet within a one hour window; also called distributed scan

  ▪ Stealth scan: horizontal or vertical scans initiated with a very low frequency to avoid detection

❑ A *Honeypot* is a resource, which pretends to be an attacked or compromised real target, but is a redundant or isolated resource where the attacker can not do any real damage.

❑ Motivation

   ▪ *Get to know the "enemy"!!*

❑ Low-Interaction Honeypots:

   ▪ Emulated services (e.g. FTP) and emulated operations systems

   ▪ Easier to deploy and maintain

   ▪ Can log only limited information

   ▪ Limited capture of activities

❑ High-Interaction Honeypots

   ▪ Involves real operation systems and real applications

   ▪ Can capture extensive amount of information

   ▪ Problem: Attackers can use this real operating system to attack non-honeypot systems.

Honeypots (2)

- Honeypots can capture unknown attacks.
- Honeypots can slow down or even stop the spread of worms.
  - Worms scan for vulnerabilities, and take over the system.
  - A honeypot can slow the scanning capabilities of the worm and eventually stop it.
    - scan unused IP spaces
    - TCP window size is zero.
- Real systems can not be taken offline for analysis.
  - They are often too critical.
  - They contain too much data pollution involved such as it is difficult to determine what the attacker actually did.
- Honeypots can quickly and easily be taken offline for a full forensic analysis.
- High-interaction honeypots are a very effective solution to prevent intrusion.
- They provide in-depth knowledge about the behavior of attackers.

❑ Introduction and Threat Overview

❑ Denial of Service Threats

❑ DoS Attacks: Classification

❑ System Vulnerabilities

❑ Honeypots

❑ **Upcoming Challenges**

Some Upcoming Challenges

❑ The introduction of Internet protocols in classical and mobile telecommunication networks also introduces the Internet's DoS vulnerabilities to these networks

❑ Programmable end-devices (PDAs, smart phones) may constitute a large base of possible slave nodes for DDoS attacks on mobile networks

❑ Software defined radio implementation may even allow new attacking techniques:

- Hacked smart phones answer to arbitrary paging requests
- Unfair / malicious MAC protocol behavior
- ...

❑ The ongoing integration of communications and automation ($\rightarrow$ sensor/actuator networks) may enable completely new DoS threats