Network Security

Chapter 0

Attacks and Attack Detection

Attacks and Attack Detection

Have you ever been attacked (in the IT security sense)?

What kind of attacks do you know?

What can happen?

- Part 0: Attacks
- Part I: Attack Prevention
- Part II: Attack Detection
- Part III: Response Mechanisms

Attacks by Impact

Disruptive:

- The goal is to fully deny the victim‘s service to its clients

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.

Leakage of data

- Confidential data, passwords, password files, keys, …

Control

- Being able to command a machine (may not interfere with normal operation)

Scans and Port Scans

Scans

- A scan is an active attack to obtain information about a network and its systems. The attacker contacts machines and requests information in a systematic way and analyzes the result.
- Port Scan: scan is to see which ports are open on a machine

Can leak info about

- Network Topology
- Operating System
- Applications and Application Versions
- …

Used to

- Use information for subsequent attacks

Denial of Service attacks

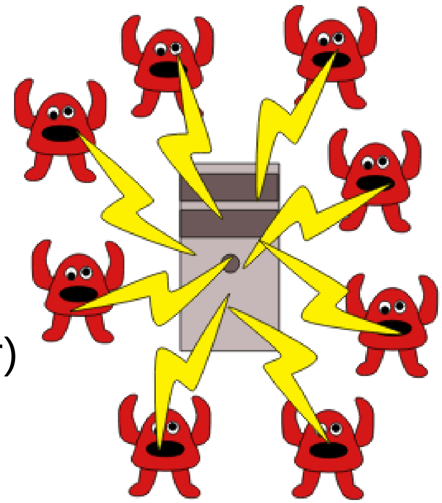
What is Denial of Service?

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

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)*

Resource Destruction via unforeseen error cases (ancient examples)

Ping-of-Death:

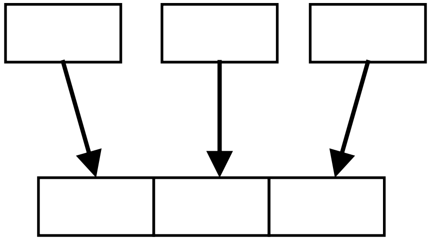
Maximum size of TCP/IP packet is 65536 bytes

Oversized packet may crash, freeze, reboot system

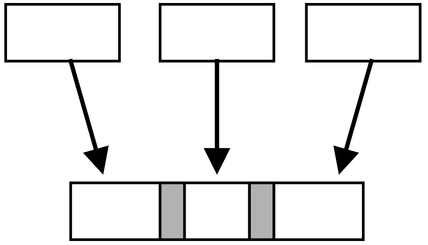
Teardrop:

Fragmented packets are reassembled using the Offset field.

Overlapping Offset fields might cause system to crash.



Normal Behavior



Teardrop Attack

Take-Home Message:

Only a few packets can be sufficient to bring down a system.

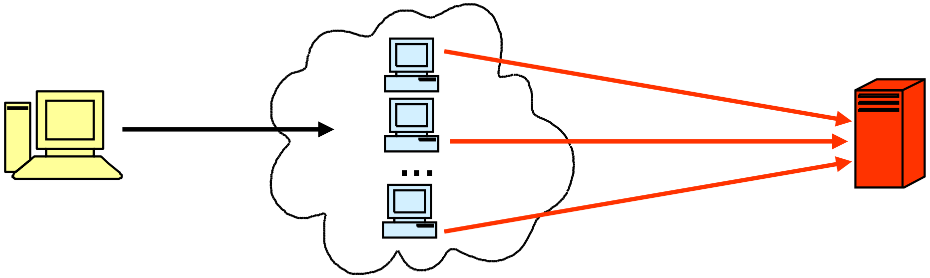
Resource Depletion Example 1: Abusing Multicast or Broadcast

Here with ICMP:

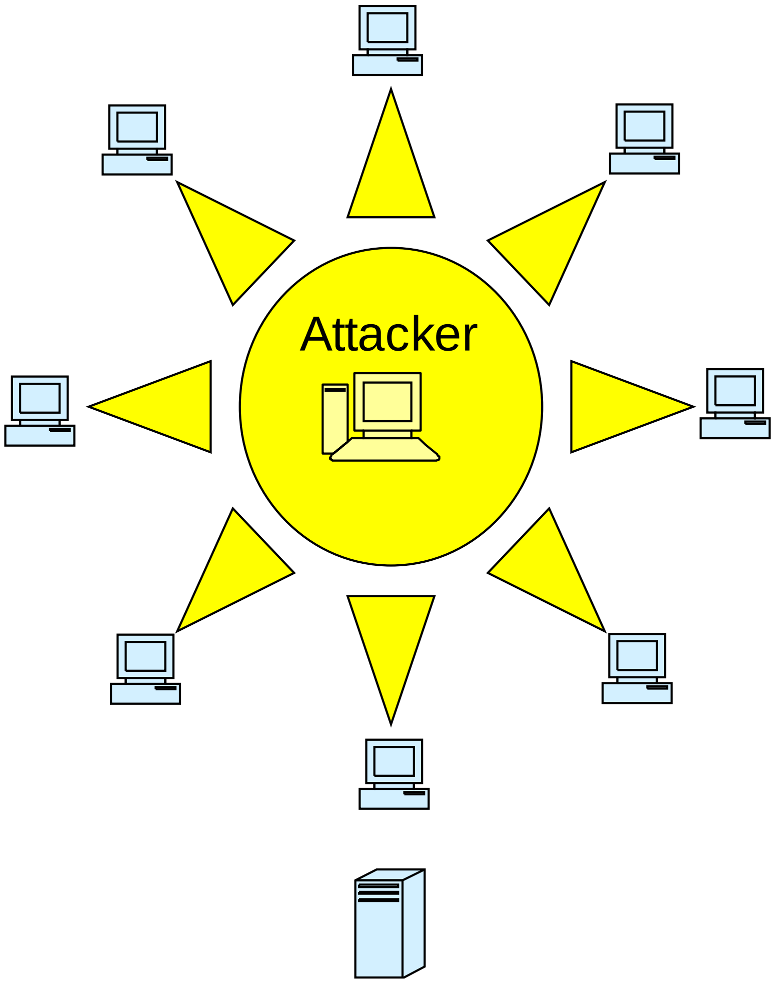
- It may be addressed to broadcast addresses
- Routers respond to it

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:*

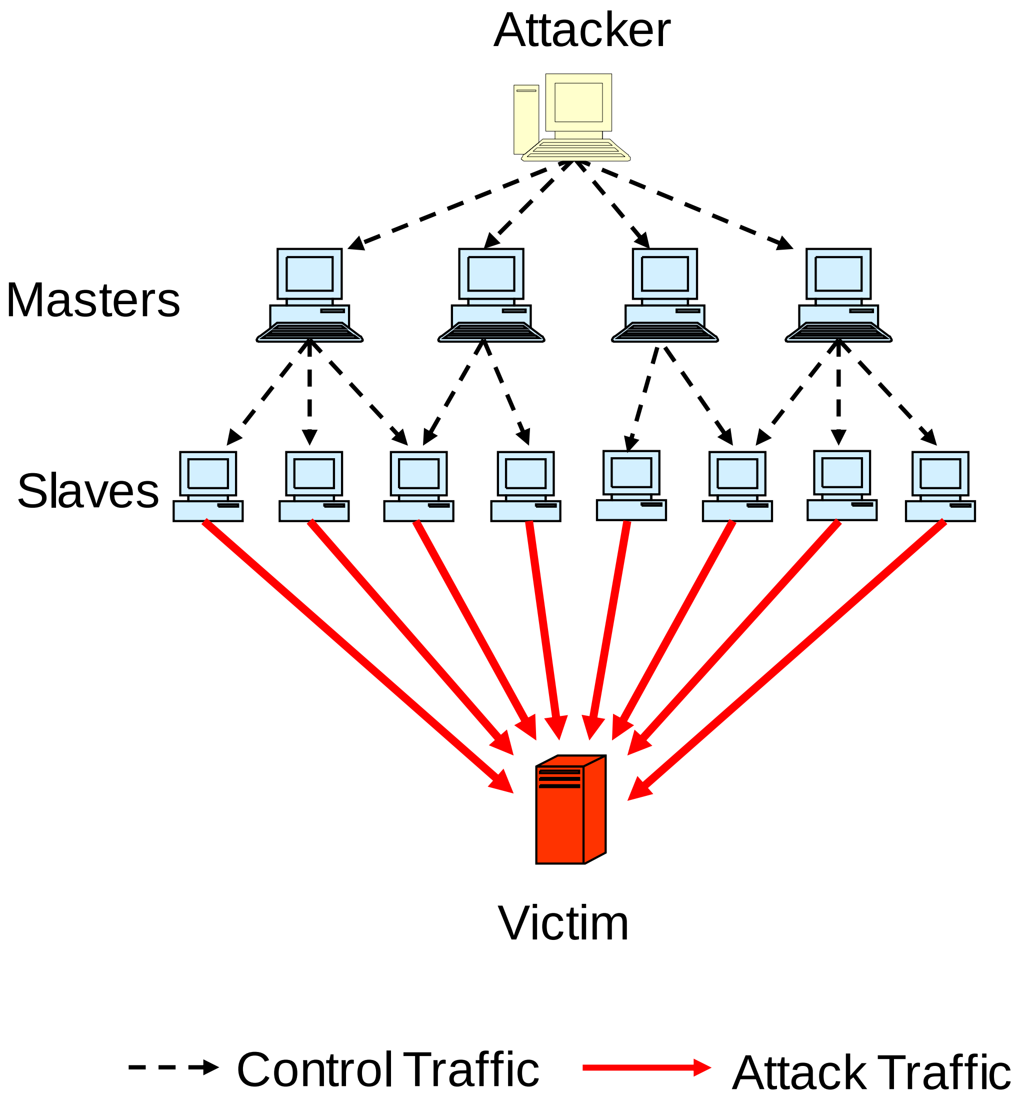


Resource Depletion with Distributed DoS (1)



- 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

Resource Depletion with Distributed DoS (2)

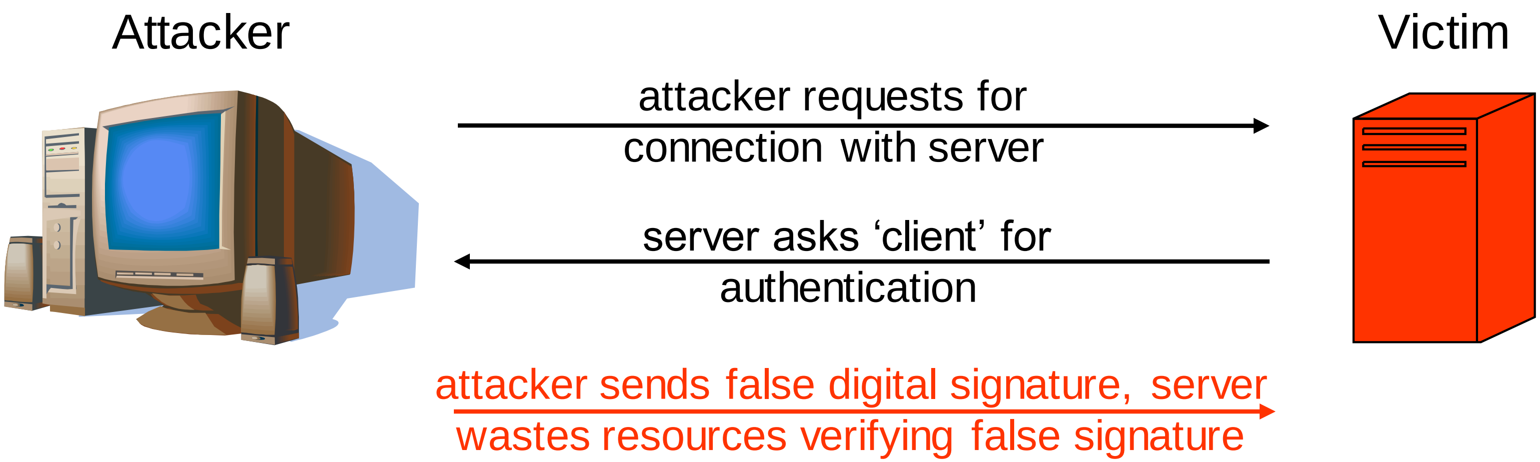


- The attacker classifies the compromised systems in:
  - Master systems
  - Slave systems
- Master systems:
  - Receive command data from attacker
  - Control the slaves
- Slave systems:
  - Launch the proper attack against the victim
- During the attack there is no traffic from the attacker

Resource Depletion with CPU Exhaustion

Category *CPU exhaustion by causing expensive computations:*

Here: attacking with bogus authentication attempts



- 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!!!

Part I: Attack Prevention

- Part 0: Attacks
- Part I: Attack Prevention
- Part II: Attack Detection
- Part III: Response Mechanisms

Attack Prevention

Covered by all chapters of the course except this one.

Attack Prevention

Prevention:

All measures taken in order to avert that an attacker succeeds in realizing a threat

Examples:

- Cryptographic measures: encryption, computation of modification detection codes, running authentication protocols, etc.
- Firewall techniques: packet filtering, service proxying, etc.

➡ Preventive measures are by definition taken *before an attack takes place*

Attention: it is generally impossible to prevent every potential attack!

- Part 0: Attacks
- Part I: Attack Prevention
- Part II: Attack Detection
- Part III: Response Mechanisms

Introduction

Host IDS vs. Network IDS

Knowledge-based Detection

Anomaly Detection

Attack Detection: What can we do?

Detection where?:

- Host-based Intrusion Detection Systems (HIDS)
- Network- based Intrusion Detection Systems (NIDS)

Detection how?:

- Knowledge-based detection
- Anomaly detection
- Hybrid attack detection

Introduction

Host IDS vs. Network IDS

Knowledge-based Detection

Anomaly Detection

Host Intrusion Detection Systems (HIDS)

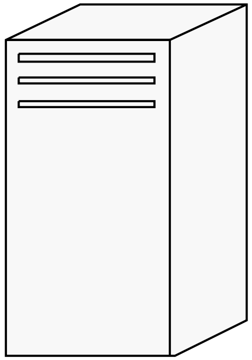
Use information available on a system, e.g. OS-Logs, application-logs, timestamps

Can easily detect attacks by insiders, as modification of files, illegal access to files, installation of Trojans or root kits

Drawbacks:

- Has to be installed on every system.
- The attack packets can not be detected before they reach the victim

  $\Rightarrow$ Host-based IDS are helpless against bandwidth saturation attacks.



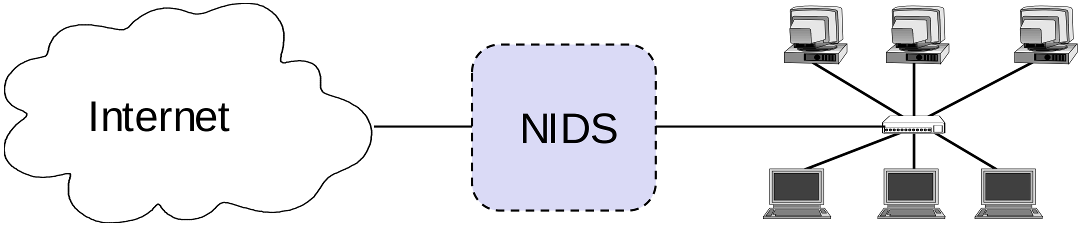
Network Intrusion Detection Systems (NIDS)

Use information provided by the network, mainly packets sniffed from the network layer.

Often used at the edges of the (sub-)networks (ingress/egress points)

Can detect known attack signatures, port scans, invalid packets, attacks on application layer, DDoS, spoofing attacks

Uses signature detection (stateful), protocol decoding, statistical anomaly analysis, heuristical analysis



Knowledge-based Attack Detection (1)

Idea:

- Store signatures of attacks in a database
- Each communication is monitored
  and compared with database entries
  to discover occurrence of attacks.

The database is occasionally updated with new signatures.

Hand detected
➔ human

Advantage:

- Known attacks can be reliably detected. Hardly “false positives” (see below for the definition of “false positives”)
- Drawbacks:
  - Only known attacks can be detected.
  - Slight variations of known attacks are not detected.

Different appellations for “Knowledge-based” attack detection in the literature

- “pattern-based” “signature-based” “misuse-based”.

Knowledge-based Attack Detection (2)

Patterns can be specified at each protocol level

- Network protocol (e.g. IP, ICMP)
- Transport protocol (e.g. TCP, UDP)
- Application protocol (e.g. HTTP, SMTP)

Example of a rule in the IDS Snort (http://www.snort.org/)

```
alert tcp $HOME_NET any -> any 9996 \

(msg:"Sasser ftp script to transfer up.exe"; \
content:"|5F75702E657865|"; depth:250; flags:A+; classtype: misc-activity;
\ sid:1000000; rev:3)
```

5F75702E657865

Fragment of Sasser located
➔ Sasser

Anomaly Detection (1)

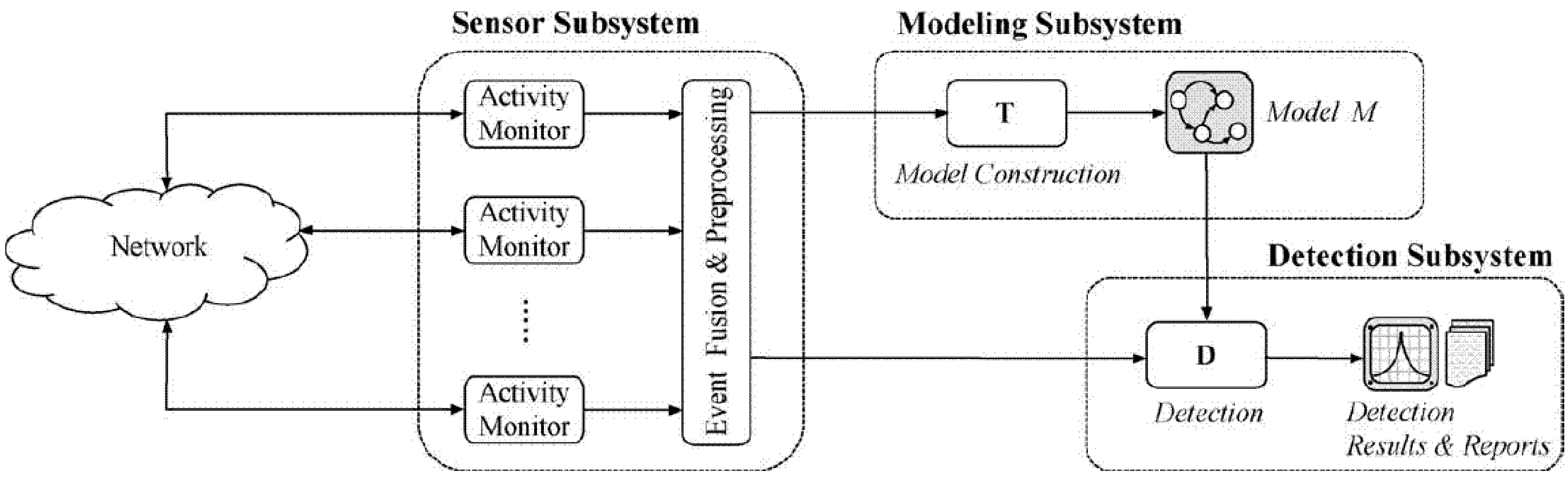
- Anomaly detection systems include a model of “normal system behavior” such as:
  - normal traffic dynamics
  - expected system performance
- The current state of the network is compared with the models to detect anomalies.
- If the current state differs from the normal behavior by a threshold then an alarm is raised.
- Anomalies can be detected in
  - Traffic behavior
  - Protocol behavior
  - Application behavior

Anomaly Detection (2)

A formal definition: [Tapidor04]

An anomaly detection system is a pair $\delta = (M, D)$, where:

- *M* is the model of normal behavior.
- *D* is similarity measure that allows obtaining, giving an activity record, the degree of deviation (or likeness) that such activities have with regard to the model *M*.



Source: [Tapiador04]

Simple Anomaly Detection

Performance Metrics of your system

E.g. number of requests

Define a normal operational interval for the metric.

Anomaly if metric outside of interval („fixed threshold“).

E.g. number of requests > 200 requests per second

Cons:

- Legitimate change of system over time, e.g. usage increases over the years (→ success is no attack)
- No inclusion of periodic changes (e.g. daily and weekly changes in use) and trend changes (usage increases 8 % in year) as above

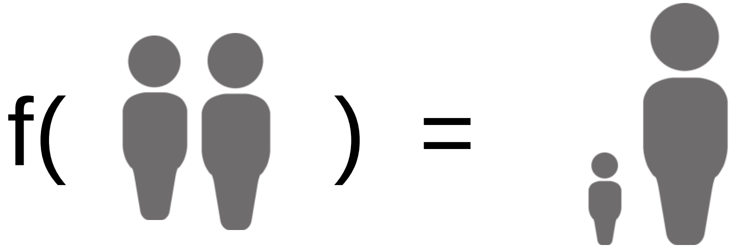
Other options

Time series (of performance metrics) → Change detection in time series

- The assumption is that an attack changes the system comparably rapidly.
- A resource depletion attacker will not slowly increase bandwidth for a year until succeeding.

Change detection

- Ignore single outliers
- Respond quickly once multiple values indicate change
- Basis usually a function that amplifies the change.



CUSUM

CUSUM (cumulative sum) is a change detection function

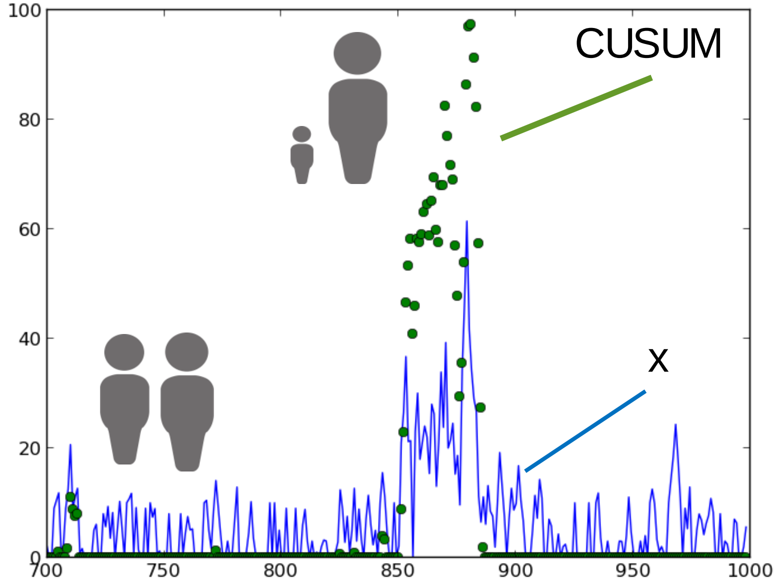
S(0) = 0

$S(t) = \max(0, S(t-1) + x(t) - m - k\, s)$

with x input stream and m a mean and s a standard deviation and k a factor.

The consequence is that

- S = 0 whenever average or small values
- S small whenever single or few large values occur
- S large whenever many large values occur at some moment in time



Detection if S(t) > threshold h

- h can be adaptable to a mean + k2 std dev where k2>k

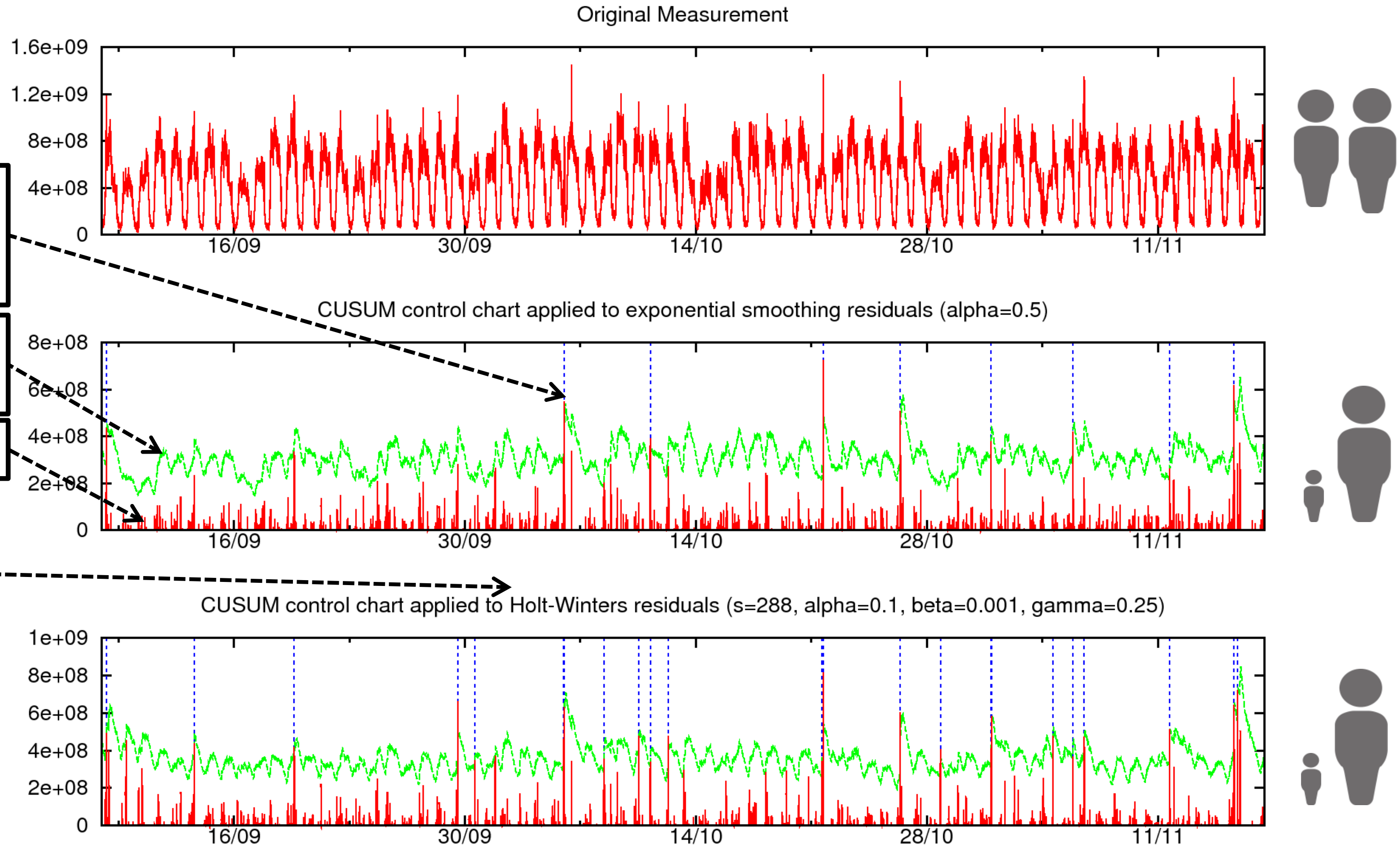
CUSUM Example (Bytes in an ISP network)

Blue dots: Change detected

Green: threshold

Red: CUSUM

Holt-Winters takes into account periodic and seasonal effects



From Gerhard Münz. *Traffic Anomaly Detection and Cause Identification Using Flow-Level Measurements*. PhD thesis, Technische Universität München, June 2010.

Anomaly Detection (2)

Pros

Might recognize some unknown attacks as well

Cons

False-positive (see definition below) rate might be high

Definitions:

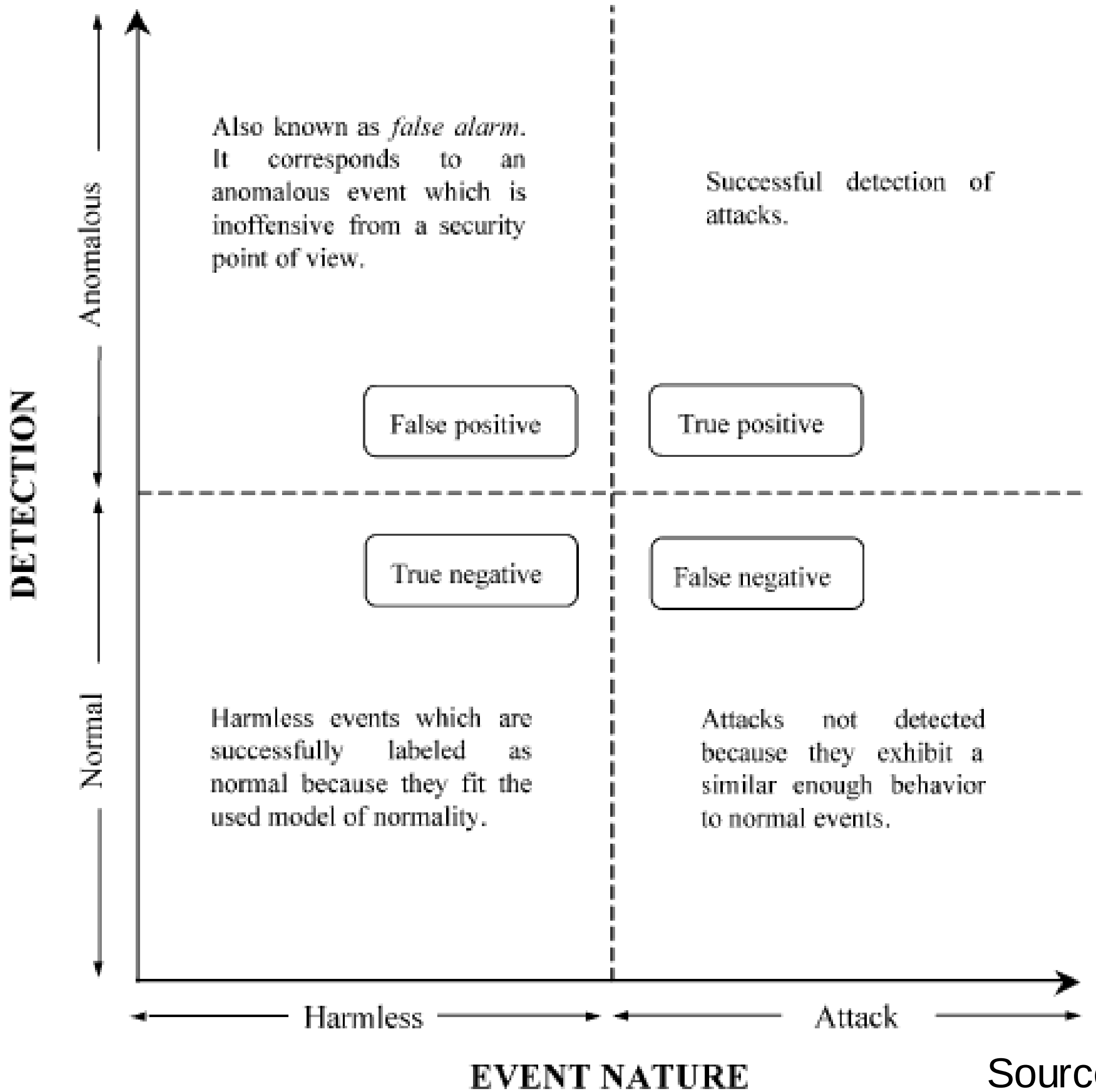
A *false positive* means the attack detection system raises an alarm while the behavior is legitimate.

A *false negative* means that an attack happens while it is classified by the attack detection system as normal behavior.

$\Rightarrow$ If the threshold for raising an alarm is set too low, the false positive rate is too high.

If the threshold is set too high, the attack detection system is insensitive.

Detection Quality



Source: [Tapiador04]

Anomaly Detection (3)

Challenges

- Modeling Internet traffic is not easy
- Data collection issues
  - Collection is expensive, collecting the right information is important
- Anomalies can have different reasons

Network Operation Anomalies

- caused, e.g. by a link failure or a configuration change
- In modern data centers, migration of a virtual machine

Flash Crowd Anomalies

- rapid rise in traffic flows due to a sudden interest in a specific services (for instance, a new software path in a repository server or a highly interesting content in a Web site)

Network Abuse Anomalies

- such as DoS flood attacks and port scans

- Part 0: Attacks
- Part I: Attack Prevention
- Part II: Attack Detection
- Part III: Response Mechanisms

Attack Response

Covered mostly in the chapter on Firewalls / Packet Filtering.

Additional answers with respect to DoS attacks:

- Content Distribution Networks (or Content Delivery Networks)
    - Large distributed network of servers in many networks and data centers
    - E.g. provided by companies like Akamai
- IP Anycast
    - Different machine contacted depending on location → restrict attack to certain area

Issues with Packet Filtering and Attack Response

Attack packets are filtered out and dropped.

Challenges

- How to distinguish between legitimate packets (the „good“ packets) and illegitimate packets (the „bad“ packets).
- Attacker’s packet might have spoofed source addresses

Filterable attacks

- If the flood packets are not critical for the service offered by the victim, they 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 to both attackers and legitimate users.