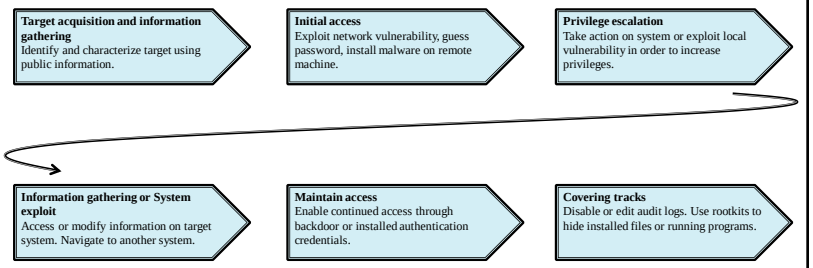
Intrusion Detection and Firewalls

Attackers/intruders can be divided into

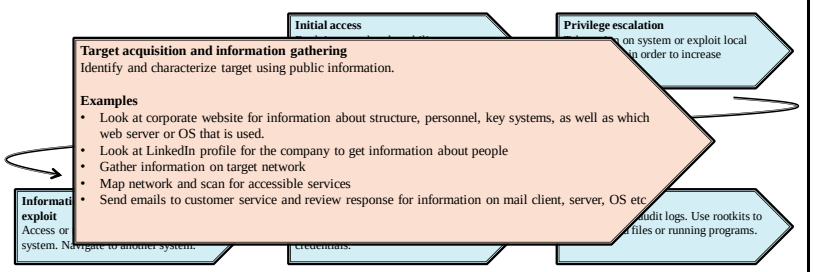
- **Cyber criminals**
  - Goal is financial reward (ransomware, cryptojacking)
- **Activists**
  - Social or political causes. Looking for publicity (web site defacement, DoS, data theft that results in negative publicity for target)
- **State-sponsored organizations**
  - Espionage and sabotage (Stuxnet)
- **Others**
  - Motivated by technical challenge, reputation, street cred. (Reports new vulnerabilities, often responsibly)

Intrusion Attack Components



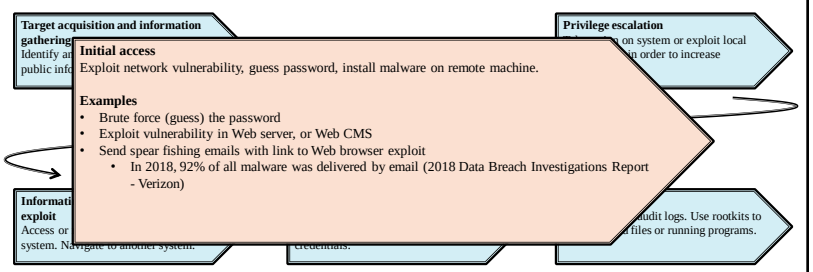
Actual attacks can of course use only a subset of these components

Intrusion Attack Components



Actual attacks can of course use only a subset of these components

Intrusion Attack Components



Actual attacks can of course use only a subset of these components

Intrusion Attack Components

Target acquisition and information gathering
Identify ar
public info

Initial access

Privilege escalation
Take action on system or exploit local vulnerability in order to increase privileges.

Examples
- Scan system for applications with local exploit
- Exploit vulnerable applications to gain elevated privileges
- Install sniffers or keyloggers to capture administrator passwords
- Use captured administrator passwords to access privileged information

Informati exploit
Access or
system. Navigate to another system.

audit logs. Use rootkits to
files or running programs.

Actual attacks can of course use only a subset of these components

Intrusion Attack Components

Target acquisition and information gathering
Identify ar
public info

Initial access

Privilege escalation
on system or exploit local
in order to increase

Information gathering or System exploit
Access or modify information on target system. Navigate to another system.

Examples
- Scan files for desired information
- Transfer larger number of documents to external repository
- Use guessed or captured passwords to access other servers on the network

audit logs. Use rootkits to
files or running programs.

Actual attacks can of course use only a subset of these components

Intrusion Attack Components

Target acquisition and information gathering
Identify ar
public info

Initial access

Privilege escalation
on system or exploit local
in order to increase

Maintain access
Enable continued access through backdoor or installed authentication credentials.

Examples
- Install remote administration tool or rootkit with backdoor for later access
- Use administrator password to later access network
- Modify or disable antivirus or IDS running on system

Informati exploit
Access or
system. Navigate to another system.

audit logs. Use rootkits to
files or running programs.

Actual attacks can of course use only a subset of these components

Intrusion Attack Components

Target acquisition and information gathering
Identify ar
public info

Initial access

Privilege escalation
on system or exploit local
in order to increase

Covering tracks
Disable or edit audit logs. Use rootkits to hide installed files or running programs.

Examples
- Use rootkit to hide files installed on system
- Edit logfiles to remove entries generated during the intrusion

Informati exploit
Access or
system. Navigate to another system.

Actual attacks can of course use only a subset of these components

Intrusion Detection

- Cryptography and protocols focus on preventing attacks
- Systems designed to detect attacks are called Intrusion Detection Systems (IDS)

Security Intrusion
Unauthorized act of bypassing the security mechanisms of a system.

Intrusion Detection
A hardware or software function that gathers and analyzes information from various areas within a computer or a network to identify possible security intrusions.

Intrusion Detection (another definition)
Sensing and analyzing system events for the purpose of noticing (i.e., becoming aware of) attempts to access system resources in an unauthorized manner.

RFC 4949, Internet Security Glossary

Motivation and IDS Components

Reasons to have intrusion detection

1. If we detect an intrusion quickly enough, the intruder can be identified and ejected before damage is done
2. An effective IDS can serve as a deterrent and help prevent intrusion
3. An IDS can help collect information about intrusion techniques and help making prevention stronger

Logical components of an IDS

1. Sensor – Collects data. Takes e.g., log files or network packets as input. Forwards information to analyzer.
2. Analyzer – Takes input from sensors and determines if there has been an intrusion. Outputs indication of intrusion and evidence.
3. User Interface – Lets a user view output and control behaviour.

IDS Classifications

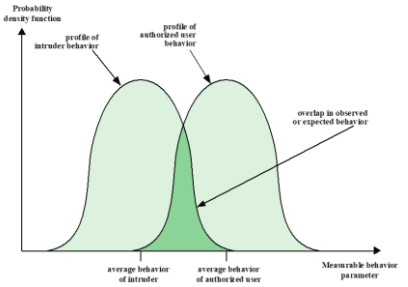
Host-based IDS (HIDS)
Monitors events within a single host and the events occuring within that host. Examples are process identifiers and the system calls that are made by the processes.

Network-based IDS (NIDS)
Monitors network traffic for particular network segments or devices. Analyzed network, transport and application protocols.

Distributed or hybrid IDS
Combines information from several sensors, both from host and network based.

Assumptions and Errors

- **Assumption**: Intruder behavior differs from legitimate behavior in a quantifiable way
- Introduces errors due to overlaps between behaviour



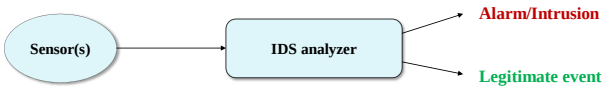
<u>Interpret intruder behavior</u>

Loose interpretation
- Catch more intruders
- Identify more legitimate users as intruders

Tight interpretation
- Legitimate users are not identified as intruders
- Many intruders not identified

False Alarms

- All alarms must be (manually) investigated
  - Time consuming
- Keep number of false alarms low
  - Difficult when the probability for an event to be an intrusion is very low



Base Rate Fallacy

- Intrusions are not very common compared to legitimate uses of system

Example

- Probability that an event is an intrusion: **1/1000**
- Probability that we detect (raise Alarm for) an actual intrusion: **0.98**
- Probability that we raise Alarm when we have legitimate use: **0.01**

What is the probability of **no attack** (Legitimate use) when we have **raised alarm?**

Some Probability

Conditional probability

Probability of event A, assuming event B has occurred.

$$\Pr[A|B] = \frac{\Pr[AB]}{\Pr[B]}$$

Probability of event A and B

Probability of event B

Example

What is the probability of getting 8 when rolling two dice, assuming at least one die is even?

A: "Getting 8"
B: "At least one is even"

$\Pr[AB] = \Pr[\text{"Getting 8" and "At least one is even"}] = \frac{3}{36} \qquad ((2,6);(4,4);(6,2))$
$\Pr[B] = \Pr[\text{"At least one is even"}] = \frac{3}{4}$

$\Rightarrow \Pr[A|B] = \Pr[\text{"Getting 8"}|\text{"At least one is even"}] = \frac{\frac{3}{36}}{\frac{3}{4}} = \frac{1}{9}$

Bayes Theorem

Law of total probability

E_i are mutually exclusive events that together cover all outcomes

$$\Pr[A] = \sum_{i=1}^{n} \Pr[A|E_i]\Pr[E_i]$$

Bayes Theorem

$$\Pr[E_i|A] = \frac{\Pr[A|E_i]\Pr[E_i]}{\Pr[A]} = \frac{\Pr[A|E_i]\Pr[E_i]}{\sum_{j=1}^{n}\Pr[A|E_j]\Pr[E_j]}$$

Base Rate Fallacy (Again)

- Assume that intrusions are not very common compared to legitimate uses of system

- Probability that an event type is an intrusion: **Pr[I] = 1/1000**
- Probability that an event type is legitimate use: **Pr[L] = 999/1000**
- Probability that we detect (raise Alarm for) an actual intrusion: **Pr[A | I] = 0.98**
- Probability that we raise Alarm when we have legitimate use: **Pr[A | L] = 0.01**

- Probability of no attack (Legitimate use) when we have raised alarm **(False alarm): Pr[L | A]**

$$\Pr[L|A] = \frac{\Pr[A|L]\Pr[L]}{\sum_{j=1}^{n}\Pr[A|\text{Type}_j]\Pr[\text{Type}_j]} = \frac{\Pr[A|L]\Pr[L]}{\Pr[A|L]\Pr[L]+\Pr[A|I]\Pr[I]} = \frac{0.01\cdot 0.999}{0.01\cdot 0.999+0.98\cdot 0.001} = 0.911$$

Analysis Approaches

Two main approaches to detect intrusions

Anomaly Detection
Collect data over time related to legitimate behavior. Use this to define what is "normal". Determine if current behavior is that of a legitimate user or of an intruder.

Signature or Heuristic Detection
Use known malicious usage patterns (signatures) or attack rules (heuristics) and compare with current behavior. Directly defines malicious or unauthorized behavior.

Anomaly Detection

Also called behavior-based IDS

- Uses statistics – Determine what is normal (baseline) and detect variations from normal
  - Baseline might be dynamically updated
  - Can typically only train on legitimate data

Problems with baseline

- Attacks are not necessarily anomalies
  - Miss attacks that do not deviate from normal
- Legitimate users may do legitimate things that still deviate from normal
  - Raise alarms when we have legitimate use

Main advantage

- Can detect zero-day attacks (previously unknown attacks)

Anomaly Detection

Common metrics

- *Counter* – Can e.g., be number of logins/hour, number of times a command is executed/login, number of password failures
- *Gauge* – Can e.g., be number of connections to application or server
- *Interval timer* – Length between two related events e.g., logins to an account
- *Resource utilization* – Amount of resources used during some period e.g., pages printed, total time of program execution

Finding anomalies

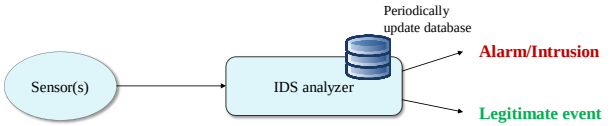
- *Mean and standard deviation*
- *Multivariate* – Correlation between two or more variables
- *Markov process* – Transition between states e.g., commands
- *Time series* – A sequence of events that happen too rapidly or too slowly
- *Operational model* – What is normal?

Signature or Heuristic Detection

Also called misuse detection or knowledge-based IDS

- Looks for attack signatures – examine network traffic or log files, e.g., failed logins
- Uses a database of signatures which has to be kept up to date
- Works well against attacks with a fixed bahaviour
- A race between attackers and signature developers
- Heuristic rules can also be used
  - Users should not read files in other users' personal directories
  - Users must not write to other users' files
  - User do not open devices directly, but instead through other programs
  - Users should not be logged in more than once to the same system
  - Users do not make copies of system programs

Not required for anomaly detection



Host-Based Intrusion Detection

- Can detect both **external** and **internal** attacks
- Can be both signature/heuristic based or based on anomaly detection

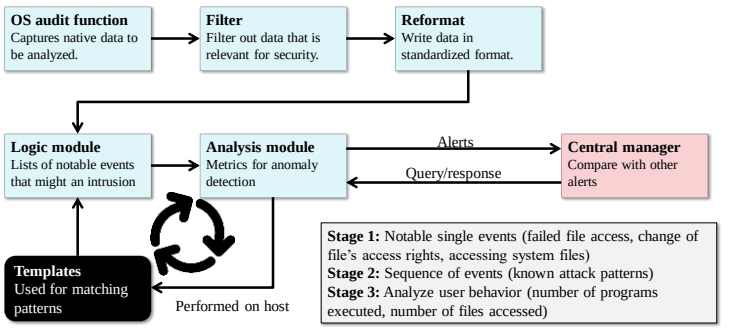
Data sources and sensors

- System call traces: analyze the sequence of system calls made by processes
  - Provide much information about a process's activity
  - Most common
- Audit (log file) records: Analyze the log files that are maintained by the underlying system
  - Does not require additional software for collecting data
- File integrity checksums: Detect modifications of files by saving a hash of the file content
  - Must save hashes somewhere
  - Can incur overhead
  - Files do change for legitimate reasons
- Registry access: Monitor access to Windows registry
  - Limited to Windows

Distributed HIDS

- Having one HIDS on each computer is possible, but centralized analysis can
  - Reduce workload
  - Use combined information
- But…
  - Different sensor formats may have to be supported
  - Data sent across network require integrity protection, and often confidentiality protection
  - Centralized analyzer can be a bottleneck and is a single point of failure
  - May require high bandwidth if much information is collected
- Use agents on hosts and send data to a central manager

Distributed HIDS, Example Approach



Stage 1: Notable single events (failed file access, change of file's access rights, accessing system files)
Stage 2: Sequence of events (known attack patterns)
Stage 3: Analyze user behavior (number of programs executed, number of files accessed)

Network-based Intrusion Detection

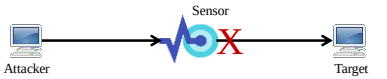
- Monitor traffic on a network, i.e., traffic directed to a potentially vulnerable computer
- Often incorporated into a firewall
- Analyze both traffic content and traffic patterns

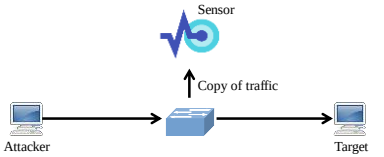
- **Main problem today:** Traffic is encrypted on the network

Network Sensors

Inline sensor

- Traffic passes through sensor.
- Often combined with firewall or switch.
- Can perform IPS



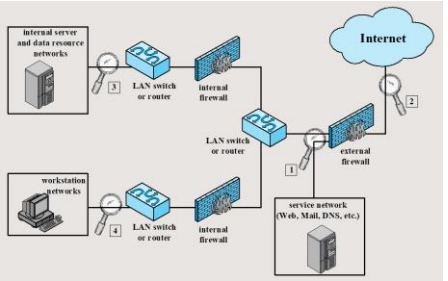


Passive sensor

- Monitors copy of traffic.
- More efficient since it does not add extra handling which causes delay.
- Easy deployment

Sensor Deployment

Possibilities for sensor placement



1. Just inside external firewall
   - Sees outside attacks that are able to bypass firewall
   - Highlights problems with firewall
   - Sees attacks that target web server, mail server etc
   - Can see outgoing traffic that might be result of an attack
2. Between external firewall and Internet
   - Documents attacks originating from Internet (type and number)
3. Just outside internal servers
   - Monitors a large amount of network's traffic
   - Can detect unauthorized activity by authorized users within the security perimeter
4. Just outside critical subsystems
   - Detects attacks targeting critical systems and resources
   - Puts focus on network assets that are considered most critical

Honeypot

- A security resource who's value lies in being probed, attacked or compromised
- Filled with information that seem useful, but is useless to owner.
- No legitimate reason to interact with the honeypot
  - Any access to honeypot is by definition an attack
  - Outbound connections are results of compromise
- Goals
  - Divert attackers from critical systems
  - Collect information about attacker's activity
  - Encourage the attacker to stay on the system long enough to be identified
- Classifications based on interaction

Types of Honeypots

Low interaction honeypot

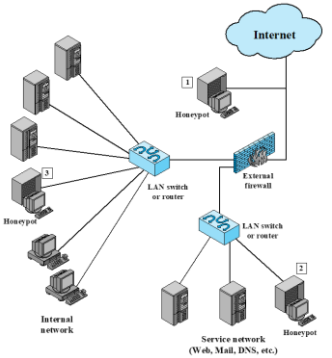
- Emulates services up to the point of initial interaction
  - Less realistic
  - Can identify intruders in the early stages of attacks

High interaction honeypot

- Real system with real services and applications
  - More realistic
  - Requires more resources
  - Might be used to initiate attacks on other systems

More advanced variants can simulate a complete network of computers

Honeypot Deployment



1. Outside external firewall
   - Track attacks to unused IP-addresses owned by company
   - No risk for internal network
   - Reduces alarms by IDS that is inside the external firewall
   - Can not detect internal attackers
2. In service network (DMZ)
   - Other systems in DMZ must be secured
   - Firewall must be configured to allow traffic to honeypot
3. Fully internal
   - Can catch internal attacks
   - Can detect misconfigured firewall
   - If honeypot is compromised, other systems are at risk
   - Firewall must be configured to allow traffic to honeypot

Firewalls

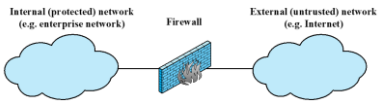
> **Firewall**
> An internetwork gateway that restricts data communication traffic to and from one of the connected networks (the one said to be "inside" the firewall) and thus protects that network's system resources against threats from the other network (the one that is said to be "outside" the firewall).
>
> *RFC 4949, Internet Security Glossary*

Design Goals

- All traffic must pass through the firewall
  - Egress filtering – Filter outgoing traffic
  - Ingress filtering – Filter incoming traffic
- Only authorized traffic is allowed to pass
- The firewall itself is secure from attacks

Characteristics for policies

- IP addresses and protocol values (port, direction)
- Application protocol (email, web requests)
- User identity (based on authentication)
- Network activity (time, request rate)

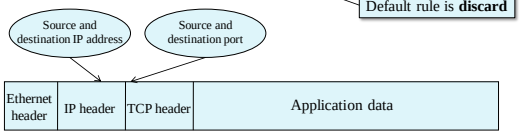


Packet Filtering Firewall

- Working at OSI levels 3 (IP) and 4 (TCP/UDP)
- Packets examined individually
- Base filter (rules) on
  - IP address (source or destination)
  - Port (source or destination)
  - Transport protocol
  - Interface
- **Negative filter** (Permissive policies) – allow all except....
- **Positive filter** (Restrictive policies) – block all except....

Default rule is **forward**

Default rule is **discard**



Packet Filter Example

Goal: Allow inbound and outbound email traffic (port 25), but block all other traffic.

Rule	Direction	Src address	Dest addresss	Protocol	Dest port	Action
1	In	External	Internal	TCP	25	Permit
2	Out	Internal	External	TCP	>1023	Permit
3	Out	Internal	External	TCP	25	Permit
4	In	External	Internal	TCP	>1023	Permit
5	Either	Any	Any	Any	Any	Deny

1. Allow connections to internal SMTP server
2. Allow responses to the inbound connection
3. Allow connections to external SMTP servers
4. Allow external SMTP server to respond
5. Deny everything else

Problem: Rule 4 allows any inbound connections to ports >1023
Remedy: Add src port = 25. Then only SMTP servers can connect

Problem 2: Attacker can send packets with source port 25. SMTP is just the default!
Remedy: Only allow TCP packets with ACK flag. Then only responses are valid

Stateful Packet Filters (Stateful Inspection Firewall)

- **Limitation of packet filter:** Must allow all incoming ports >1023 in order to allow e.g., SMTP, FTP, HTTP
- **Instead:** Keep track of all connections in a table. Allow incoming packet on port >1023 only if it is in table.

Source address	Source port	Destination address	Destination port	Connection state
192.168.1.100	1055	66.249.93.104	80	Established
192.168.1.105	1254	68.250.190.8	21	Established
192.168.1.120	1034	72.21.203.1	80	Established
192.168.1.122	5787	66.135.194.100	80	Established
192.168.1.134	3854	207.46.193.254	80	Established

Application Level Proxy

- Relays application-level traffic
- Sets up its own connection to remote host
  - Implements the protocol
  - Can filter data at application level, e.g., remove email attachments
  - Address of proxy seen outside, not address of client
  - Can also be used to anonymize, fake source country etc.
- Can audit and log at application level
- Slower than packet filters, higher cost

Firewall Limitations

- Less protection against insider threats
- Tunneling through open ports is still possible
- Encrypted protocols can not be examined at application level
- All portable devices will bypass the firewall
- Wireless access points behind firewall can be used