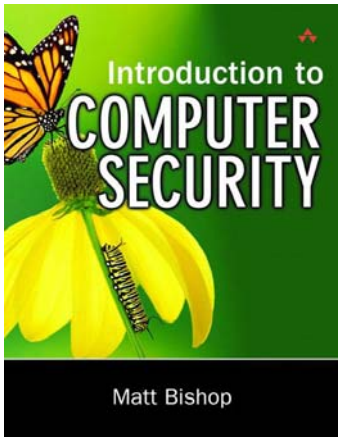
These Slides are prepared from
Matt Bishop slides and book "Introduction to Computer Security"
Benefiting from the Slides posted by Ahmad Al-Mulhem

# Intrusion Detection

Chapter 22



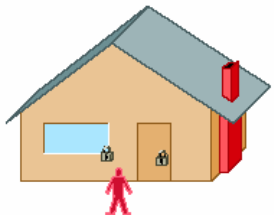

Adnan Gutub
*gutub@kfupm.edu.sa*

*Computer Engineering Department*
*King Fahd University of Petroleum & Minerals*
*Dhahran, Saudi Arabia*



# Chapter 22: Intrusion Detection

- Introduction
  - History
  - Goals
- Types
  - Misuse
  - Anomaly
  - Specification
- Source of Data
  - Network
  - Host
- Agents
  - Comparisons



## Need for Detection

- Someone trying to go in !!
- House door & window are locked …..
  - Why install an alarm?
  - What if you forget to lock?
  - What if you forget to update your firewall?

# Intrusion Detection Systems (IDS)

Definition: (Intrusion Detection Systems)

– An intrusion detection system (IDS) is a *system* that monitors computing resources (a single host or an entire network) in order to detect attacks and/or anomalous activities.

- Requires reliable and complete data about the target system
- Don not detect intrusions, instead identify evidence (manifestation) of intrusions!
- Similar to a smoke detector (detect smoke, not fire!)



# Violations Example

Goal: insert a back door into a system

– violates #1 Intruder will modify system configuration file or program

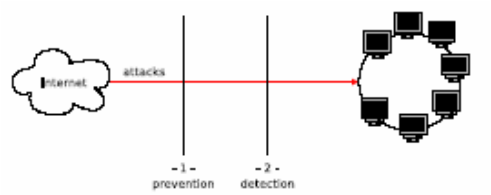
– violates #2 Requires privilege; attacker enters system as an unprivileged user and must acquire privilege

- Nonprivileged user may not normally acquire privilege (violates #1)
- Attacker may break in using sequence of commands that violate security policy (violates #2)
- Attacker may cause program to act in ways that violate program's specification

# Where does IDS fit?

Lines of defenses in network security model

- Prevention: Firewalls, authentication, access control
- Detection: IDS






# Some History: Intrusion Detection

Originally, system admins performed intrusion detection by sitting in front of a console and monitoring user activities

- e.g. a vacationing user login locally, a never used printer is active
- ad hoc and not scalable

'70s and early '80s, admins print audit logs and search them every week

- time consuming
- forensic tool after the fact
- no hope to catch ongoing intrusions

80', as storage became cheaper, audit logs moved online

- researchers developed programs to analyze the data
- but, analysis was slow and often computationally intensive, usually run at night

'90s, researchers developed real-time intrusion detection systems

- in some cases, attack preemption
- distributed intrusion detection

# Goals of IDS

Detect wide variety of intrusions
- Previously known and unknown attacks
- Suggests need to learn/adapt to new attacks or changes in behavior

Detect intrusions in timely fashion
- May need to be be real-time, especially when system responds to intrusion
  - Problem: analyzing commands may impact response time of system
- May suffice to report intrusion occurred a few minutes or hours ago

Present analysis in simple, easy-to-understand format
- Ideally a binary indicator
- Usually more complex, allowing analyst to examine suspected attack
- User interface critical, especially when monitoring many systems

Be accurate
- Minimize false positives, false negatives
- Minimize time spent verifying attacks, looking for them



# Models of Intrusion Detection (IDS Types)

Source of Data:

Host-based IDS
- Analyze logs
- Check file-system integrity

Network-based IDS
- Analyze traffic

Detection Method:

Misuse-based (signature-based) detection
- What is bad, is known
- What is not bad, is good

Anomaly-based detection
- What is usual, is known
- What is unusual, is bad

Specification-based detection
- What is good, is known
- What is not good, is bad

# Misuse-based (signature-based) detection

Signature-based IDS

- An IDS which detects attacks by matching against a database of known attacks.
  - Similar to anti-virus software
  - Have a database of attack signatures
  - If a a rule matches, an alert fires
  - Simple and effective
  - New attacks are not detected!



# Anomaly Detection

Anomaly-Based IDS

- An IDS which builds a model of "normal" system behavior, and alerts when a deviation from the model is detected.
  - Classify into either normal or anomalous
  - How do you define normal? (e.g. a user normal activities)
  - Mathematical Techniques: Threshold metrics, statistics (mean, std)
    - Example: Counts number of events that occur
      - Assume events expected to occur are between *m* and *n* (inclusive)
      - If number falls outside this range, anomalous

# Specification-based detection

Determines whether execution of sequence of instructions violates specification

Only need to check programs that alter protection state of system



# Signature-based vs Anomaly-Based

Signature-based

•Advantages

– Simple and effective

– Easy to administer

Anomaly-based

•Advantages

– Detects new attacks

•Disadvantages

- Specification-based vs. Misuse (signature-based):
  - spec assumes if specifications followed, policy not violated;
  - misuse assumes if policy as embodied in rule-sets followed, policy not violated

# IDS Architecture

Basically, a sophisticated audit system

- *Agent* like logger; it gathers data for analysis
- *Director* like analyzer; it analyzes data obtained from the agents according to its internal rules
- *Notifier* obtains results from director, and takes some action
  - May simply notify security officer
  - May reconfigure agents, director to alter collection, analysis methods
  - May activate response mechanism



# Network-Based vs Host-Based

- Network-based (NIDS)
  - Advantages
    - Easy to deploy
    - Monitors many hosts
  - Disadvantages
    - Difficulty processing in high-speed networks
    - Difficulty with encrypted protocols
    - No indicator of attack success
- Host-based (HIDS)
  - Advantages
    - Detects at application layer
    - No trouble with encryption
  - Disadvantages
    - Harder to manage, correlate data
    - IDS on host may be attacked and/or disabled

# Agents

Obtains information and sends to director

May put information into another form

- Preprocessing of records to extract relevant parts

May delete unneeded information

Director may request agent send other information

IDS uses failed login attempts in its analysis

- Example: Agent scans login log every 5 minutes, sends director for each new login attempt:
  - Time of failed login
  - Account name and entered password
- Director requests all records of login (failed or not) for particular user
  - Suspecting a brute-force cracking attempt



# Host-Based Agent

Obtain information from logs

- May use many logs as sources
- May be security-related or not
- May be virtual logs if agent is part of the kernel
  - Very non-portable

Agent generates its information

- Scans information needed by IDS, turns it into equivalent of log record
- Typically, check policy; may be very complex

# Network-Based Agent

Detects network-oriented attacks
- Denial of service attack introduced by flooding a network

Monitor traffic for a large number of hosts

Examine the contents of the traffic itself

Agent must have same view of traffic as destination
- TTL tricks, fragmentation may obscure this

End-to-end encryption defeats content monitoring
- Not traffic analysis, though

Focus is usually on intruders entering network
- If few entry points, place network agents behind them
- Does not help if inside attacks to be monitored



# Agents Pros & Cons

Advantages
- No single point of failure
  - *All* agents can act as director
  - In effect, director distributed over all agents
- Compromise of one agent does not affect others
- Agent monitors one resource
  - Small and simple
- Agents can migrate if needed
- Approach appears to be scalable to large networks

Disadvantages
- Communications overhead higher, more scattered than for single director
  - Securing these can be very hard and expensive
- As agent monitors one resource, need many agents to monitor multiple resources
- Distributed computation involved in detecting intrusions
  - This computation also must be secured

# Counterattacking

Use legal procedures

- Collect chain of evidence so legal authorities can establish attack was real
- Check with lawyers for this
  - Rules of evidence very specific and detailed
  - If you don't follow them, expect case to be dropped

Technical attack

- Goal is to damage attacker seriously enough to stop current attack and deter future attacks



# Consequences

May harm innocent party

- Attacker may have broken into source of attack or may be impersonating innocent party

May have side effects

- If counterattack is flooding, may block legitimate use of network

Antithetical to shared use of network

- Counterattack absorbs network resources and makes threats more immediate

May be legally actionable

# Example: Counterworm

Counterworm given signature of real worm

- Counterworm spreads rapidly, deleting all occurrences of original worm

Some issues

- How can counterworm be set up to delete *only* targeted worm?
- What if infected system is gathering worms for research?
- How do originators of counterworm know it will not cause problems for any system?
  - And are they legally liable if it does?



# Key Points

Intrusion detection is a form of auditing

Anomaly detection looks for unexpected events

Misuse (signature-based) detection looks for what is known to be bad – Anti Viruse

Specification-based detection looks for what is known not to be good

Intrusion response requires careful thought and planning