Intrusion Detection

Unit 7 [4 LH]

Topics

- Intruders
- Intrusion Techniques
- Intrusion Detection
- Combining sources
- Director
- Notifier
- Organization of IDS
- Autonomous Agents
- Intrusion Response

Intruders and Intrusion Techniques

Background

- A significant security problem for networked systems is hostile, or at least unwanted, trespass by users or software.
- User trespass can take the form of unauthorized logon to a machine or, in the case of an authorized user, acquisition of privileges or performance of actions beyond those that have been authorized.
- Software trespass can take the form of a virus, worm, or Trojan horse.

Intruders and Intrusion Techniques

- One of the key threats to security is the use of some form of hacking by an ***intruder***, often referred to as a ***hacker*** or ***cracker, or interceptor.***
- **Intrusion** is a phenomenon that performs an activity that compromises a computer system by breaking the security or causing it to enter into an insecure state by an intruder.
- A set of attempts to compromise a computer or a computer network resource security is regarded as an intrusion.

Types of Intruders

- Anderson identified three classes of intruders:

1. **Masquerader**: An individual who is not authorized to use the computer and who penetrates a system's access controls to exploit a legitimate user's account. The masquerader is likely to be an outsider.
2. **Misfeasor**: A legitimate user who accesses data, programs, or resources for which such access is not authorized, or who is authorized for such access but misuses his or her privileges . The misfeasor generally is an insider.
3. **Clandestine user**: An individual who seizes supervisory control of the system and uses this control to evade auditing and access controls or to suppress audit collection. The clandestine user can be either an outsider or an insider.

Intruder Attacks

- Intruder attacks range from the benign to the serious.
- The following are some examples of intrusions:
  - Performing a remote root compromise of an e-mail server.
  - Defacing a Web server.
  - Guessing and cracking passwords.
  - Copying a database containing credit card numbers.
  - Viewing sensitive data, including payroll records and medical information, without authorization.
  - Running a packet sniffer on a workstation to capture usernames and passwords.
  - Using a permission error on an anonymous FTP server to distribute pirated software and music files.
  - Dialing into an unsecured modem and gaining internal network access.
  - Posing as an executive, calling the help desk, resetting the executive's e-mail password, and learning the new password.
  - Using an unattended, logged-in workstation without permission.

Class of Intruders

- **Cyber Criminals**
  - Goal: financial reward.
  - Activities: identity theft, theft of financial credentials, corporate espionage, data-theft, or data ransoming.
  - underground forums: like DarkMarket.org and theftservices.com to trade tips and data and coordinate attacks
- **Activist**
  - also known as hacktivists
  - Skill level: Low
  - Activities:
    - promote and publicize their cause, typically through website defacement,
    - denial of service attacks, or
    - the theft and distribution of data that results in negative publicity or compromise of their targets

Class of Intruders

- **State-Sponsored Organization**
  - groups of hackers sponsored by governments to conduct espionage or sabotage activities.
  - also known as Advanced Persistent Threats (APTs)
  - eg. information revealed by Edward Snowden
- **Others**
  - classic hackers or crackers who are motivated by technical challenge or by peer-group esteem and reputation.
  - responsible for discovering new categories of buffer overflow vulnerabilities.

Skill level class of intruders

- Apprentice
- Journeyman
- Master

Intrusion Detection

- **Security Intrusion**: A security event, or a combination of multiple security events, that constitutes a security incident in which an intruder gains, or attempts to gain, access to a system (or system resource) without having authorization to do so.
- **Intrusion Detection**: A security service that monitors and analyzes system events for the purpose of finding, and providing real-time or near real-time warning of, attempts to access system resources in an unauthorized manner.
- Intrusion detection is based on the assumption that the behavior of the intruder differs from that of a legitimate user in ways that can be quantified.

Intrusion Detection

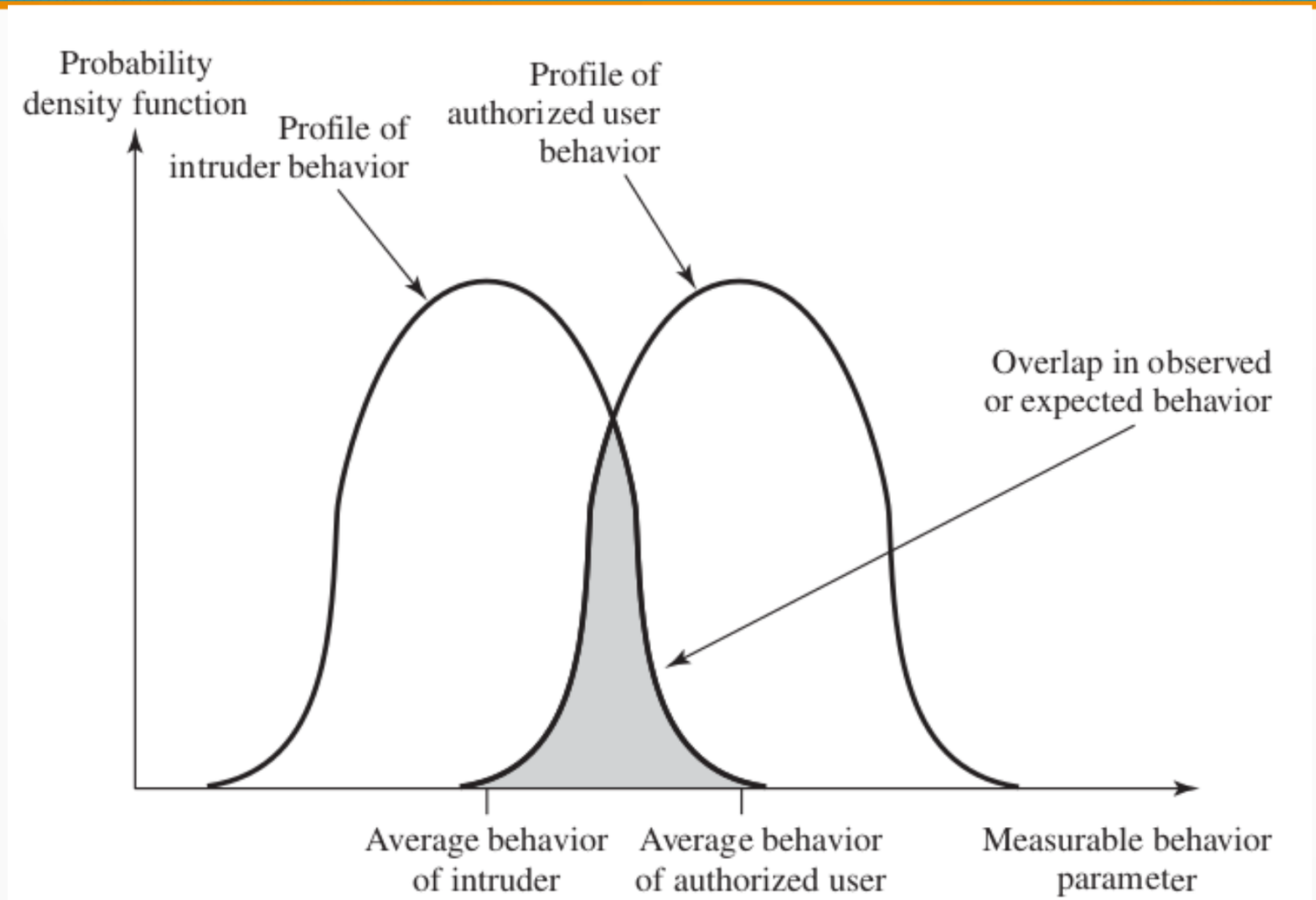

Figure: Profiles of Behavior of Intruders and Authorized Users

Goals of Intrusion Detection

- **Detect a wide variety of intrusions**.
  - Intrusions from within the site, as well as those from outside the site, are of interest.
- **Detect intrusions in a timely fashion**.
  - “Timely” here need not be in real time. Often, it suffices to discover an intrusion within a short period of time.
- **Present the analysis in a simple, easy-to-understand format.**
- **Be accurate**.
  - A false positive occurs when an intrusion detection system reports an attack, but no attack is underway. False positives reduce confidence in the correctness of the results as well as increase the amount of work involved. However, false negatives (occurring when an intrusion detection system fails to report an ongoing attack) are worse, because the purpose of an intrusion detection system is to report attacks

Intrusion Detection

An IDS comprises three logical components:

- **Sensors**: Sensors are responsible for collecting data. The input for a sensor may be any part of a system that could contain evidence of an intrusion. Types of input to a sensor includes network packets, log files, and system call traces. Sensors collect and forward this information to the analyzer.
- **Analyzers**: Analyzers receive input from one or more sensors or from other analyzers. The analyzer is responsible for determining if an intrusion has occurred. The output of this component is an indication that an intrusion has occurred. The output may include evidence supporting the conclusion that an intrusion occurred. The analyzer may provide guidance about what actions to take as a result of the intrusion. The sensor inputs may also be stored for future analysis and review in a storage or database component.
- **User interface:** The user interface to an IDS enables a user to view output from the system or control the behavior of the system. In some systems, the user interface may equate to a manager, director, or console component.

Intrusion Detection Model

- Intrusion detection systems determine if actions constitute intrusions on the basis of one or more models of intrusion.
- A model classifies a sequence of states or actions, or a characterization of states or actions, as “good” (no intrusions) or “bad” (possible intrusions).
- Model of Intrusion Detection
  1. Anomaly Modeling
  2. Misuse Modeling
  3. Specification Modeling

AnomalyModeling

- Anomaly detection analyzes a set of characteristics of the system and compares their behavior with a set of expected values.
- It reports when the computed statistics do not match the expected measurements.
- Anomaly detection uses the assumption that unexpected behavior is evidence of an intrusion.
- There are some set of metrics that characterize the expected behavior of a user or a process.
- There are three models:
  - Threshold Metrics
  - Statistical Moments
  - Markov Models
-

Threshold Metrics

- A minimum of ***m*** and a maximum of ***n*** events are expected to occur (for some event and some values m and n).
- If, over a specific period of time, fewer than ***m*** or more than ***n*** events occur, the behavior is deemed anomalous.
- Determining the threshold complicates use of this model.

Threshold Metrics

- EXAMPLE : Microsoft Windows systems allow the administrator to lock a user out after some number n of failed login attempts. This is an intrusion detection system using the threshold metric with the lower limit 0 and the upper limit n. The attempted logins are deemed anomalous after n failed attempts to log in.

Statistical Moments

- The analyzer knows the mean and standard deviation (first two moments) and possibly other measures of correlation (higher moments).
- If values fall outside the expected interval for that moment, the behavior that the values represent is deemed anomalous.
- The statistical moments model provides more flexibility than the threshold model. Administrators can tune it to discriminate better than the threshold model.
- But with flexibility comes complexity.

Markov Models

- A Markov model examine a system at some particular point in time. Events preceding that time have put the system into a particular state.
- When the next event occurs, the system transitions into a new state.
- Over time, a set of probabilities of transition can be developed.
- When an event occurs that causes a transition that has a low probability, the event is deemed anomalous.
- This model suggests that a notion of “state,” or past history, can be used to detect anomalies.
- The anomalies are now no longer based on statistics of the occurrence of individual events, but on sequences of events.
- This approach heralded misuse detection and was used to develop effective anomaly detection mechanisms.
- The effectiveness of Markov-based models depends on the adequacy of the data (i.e. training data) used to establish the model.

Misuse Modeling

- Misuse detection determines whether a sequence of instructions being executed is known to violate the site security policy being executed. If so, it reports a potential intrusion.
- In some contexts, the term “misuse” refers to an attack by an insider or authorized user. In the context of intrusion detection systems, it means “rule-based detection”.
- Modeling of misuse requires a knowledge of system vulnerabilities or potential vulnerabilities that attackers attempt to exploit. The intrusion detection system incorporates this knowledge into a rule set.
- When data is passed to the intrusion detection system, it applies the rule set to the data to determine if any sequences of data match any of the rules. If so, it reports that a possible intrusion is underway.

Misuse Modeling

- Misuse-based intrusion detection systems often use expert systems to analyze the data and apply the rule set.
- These systems cannot detect attacks that are unknown to the developers of the rule set.
- Previously unknown attacks, or even variations of known attacks, can be difficult to detect.
- Later intrusion detection systems used adaptive methods involving neural networks and Petri nets to improve their detection abilities.

Specification Modeling

- *Specification-based detection* determines whether or not a sequence of instructions violates a specification of how a program, or system, should execute. If so, it reports a potential intrusion.
- Anomaly detection has been called the art of looking for unusual states. Similarly, misuse detection is the art of looking for states known to be bad. Specification detection takes the opposite approach; it looks for states known not to be good, and when the system enters such a state, it reports a possible intrusion.
- For security purposes, only those programs that in some way change the protection state of the system need to be specified and checked. For example, because the policy editor in Windows changes security-related settings, it needs to have an associated specification.

Architecture of IDS

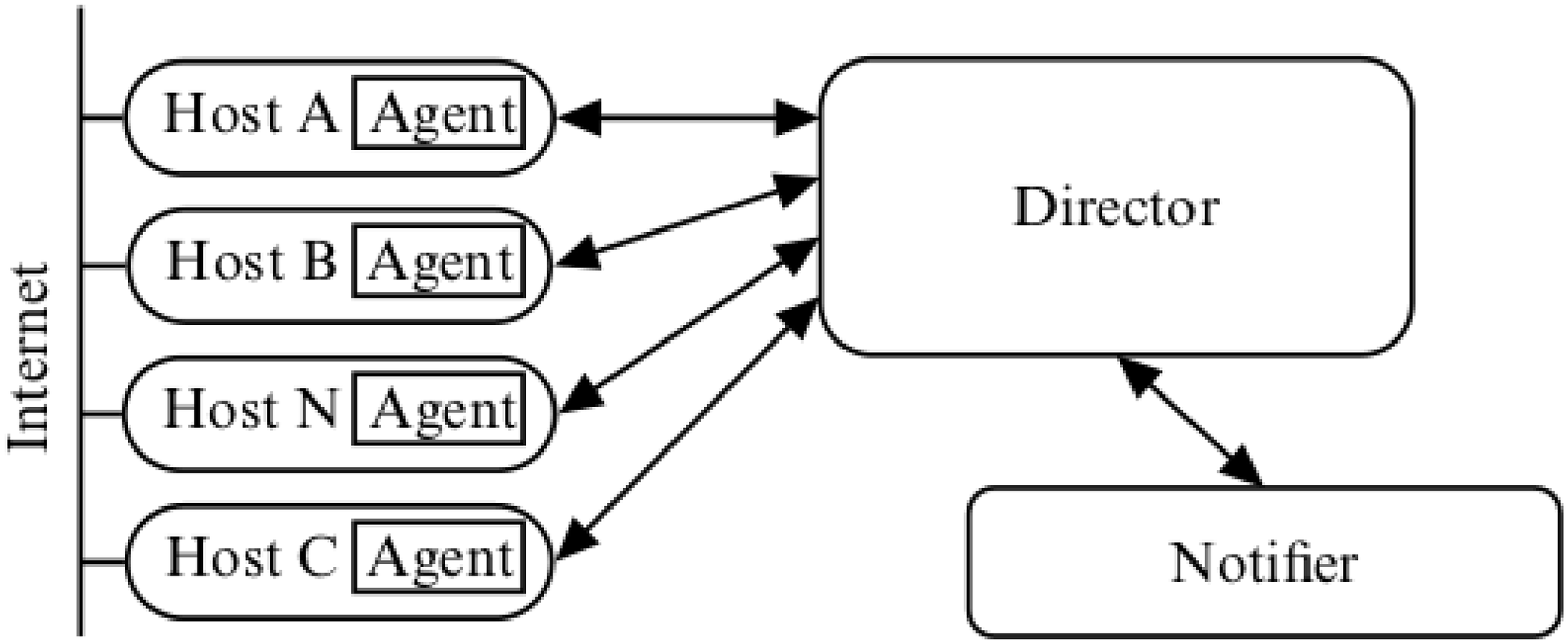

Figure: Architecture of an intrusion detection system. Hosts A, B, and C are general-purpose computers, and the agents monitor activity on them. Host N is designed for network monitoring, and its agent reports data gleaned from the Internet to the director.

Architecture of IDS

- It consists of three parts:
  - Agent
  - Director
  - Notifier
- The ***agent*** corresponds to the logger. It acquires information from a target (such as a computer system).
- The ***director*** corresponds to the analyzer. It analyzes the data from the agents as required (usually to determine if an attack is in progress or has occurred).
- The director then passes this information to the ***notifier***, which determines whether, and how, to notify the requisite entity. The notifier may communicate with the agents to adjust the logging if appropriate.

Agent

- An agent obtains information from a data source (or set of data sources).
- The source may be a log file, another process, or a network.
- The information, once acquired, may be sent directly to the director.
- Usually, however, it is preprocessed into a specific format to save the director from having to do this.
- Also, the agent may discard information that it deems irrelevant.

Agent

- The director may determine that it needs more information from a particular information source. In that case, the director can instruct the agent to collect additional data, or to process the data it collects differently. This help suspect the attack when it is underway.
- An agent can obtain information from a single host, from a set of hosts.
- The types of information that are available from each, and how they might be gathered are:
  - Host-Based Information Gathering
  - Network-Based Information Gathering
  - Combining Sources

Host Based Information Gathering

- Host-based agents usually use system and application *logs to obtain records of events*, and *analyze them to determine what to pass to the director*.
- The events to look for, and to analyze, are determined by the **goals** of the intrusion detection mechanism.
- The logs may be security-related logs or other logs such as accounting logs.

Network Based Information Gathering

- Network-based agents *use a variety of devices and software to monitor network traffic*.
- This technique provides information of a different types than host-based monitoring provides.
- It can detect network-oriented attacks, such as a denial of service attack introduced by flooding a network.
- It can monitor traffic for a large number of hosts.
- It can also examine the contents of the traffic itself (called content monitoring) .
- Network-based agents may use *network sniffing* to read the network traffic. In this case, a system provides the agent with access to all network traffic passing that host.

Combining Sources

- The goal of an agent is to provide the director with information so that the director can report possible violations of the security policy (intrusions).
- An aggregate of information is needed. However, the information can be viewed at several levels (For example, application log and system log)
- The difference between application and system views (which is, essentially, a problem of layers of abstraction) affects what the agent can report to the director and what the director can conclude from analyzing the information.
- The agent, or the director, must either obtain information at the level of abstraction at which it looks for security problems or be able to map the information into an appropriate level.

Director

- The director receives data from the agent and uses analysis engine to analyze if the attack is underway or has already occurred.
- Because the functioning of the director is critical to the effectiveness of the intrusion detection system, it is usually run on a separate system. This allows the system to be dedicated to the director's activity.
- It keeps the specific rules and profiles unavailable to ordinary users. Then attackers lack the knowledge needed to evade the intrusion detection system.
- Director uses the ***adaptive*** techniques to alter the set of rules that they use to make decisions.
- Directors rarely use only one analysis technique, because different techniques highlight different aspects of intrusions. The results of each are combined, analyzed and reduced, and then used.

Director

- EXAMPLE : A particular user logs in during the day to perform system maintenance functions. Occasionally she logs in during the late evening to write reports. One day, she apparently logs in during the late evening and begins altering the kernel (a system maintenance procedure). Agents provide information from both the log of login times and the log of commands executed. Neither set of data by itself will give an indication of a security problem. However, if the director correlates the two sets of data, the anomaly will be apparent.

Notifier

- The notifier accepts information from the director and takes the appropriate action.
- In some cases, this is simply a notification to the system security officer that an attack is believed to be underway.
- In other cases, the notifier may take some action to respond to the attack.
- Many intrusion detection systems use graphical interfaces.
- The notifier may contact the appropriate person (send email, SMS, etc) or make entries into the appropriate log files.

Organization of Intrusion Detection System

- An intrusion detection system can be organized in several ways.
  - Monitoring Network Traffic for Intrusions: NSM
  - Combining Host and Network Monitoring: DIDS
  - Autonomous Agents: AAFID

–Monitoring Network Traffic for Intrusions: NSM

- The Network Security Monitor develops a profile of expected usage of a network and compares current usage with that profile.
- It also allows the definition of a set of signatures to look for specific sequences of network traffic that indicate attacks.
- It runs on a local area network and assumes a broadcast medium.
- The monitor measures network utilization and other characteristics and can be instructed to look at activity based on a user, a group of users, or a service. It reports anomalous behavior.

–Monitoring Network Traffic for Intrusions: NSM

- The NSM monitors the source, destination, and service of network traffic.
- It assigns a unique connection ID to each connection.
- The source, destination, and service are used as axes for a matrix.
- Each element of the matrix contains the number of packets sent over that connection for a specified period of time, and the sum of the data of those packets.
- The NSM also generates expected connection data from the network.
- The data in the array is “masked” by the expected connection data, and any data not within the expected range is reported as an anomaly.

–Monitoring Network Traffic for Intrusions: NSM

- The NSM is important for two reasons.
  - First, it served as the basis for a large number of intrusion detection systems. Indeed, 11 years after its creation, it was still in use at many sites (although with an augmented set of signatures).
  - Second, it proved that performing intrusion detection on networks was practical.

Combining Host and Network Monitoring: DIDS

- The Distributed Intrusion Detection System (DIDS) combined the abilities of the NSM with intrusion detection monitoring of individual hosts.
  - neither network-based monitoring nor host-based monitoring was sufficient.
- An intruder attempting to log into a system through an account without a password would not be detected as malicious by a network monitor.
- Subsequent actions, however, might make a host-based monitor report that an intruder is present.
- Similarly, if an attacker tries to telnet to a system a few times, using a different login name each time, the host-based intrusion detection mechanism would not report a problem, but the network-based monitor could detect repeated failed login attempts.
- DIDS used a centralized analysis engine (the DIDS director) and required that agents be placed on the systems being monitored as well as in a place to monitor the network traffic.

Combining Host and Network Monitoring: DIDS

- The agents scanned logs for events of interest and reported them to the DIDS director.
- The DIDS director invoked an expert system that performed the analysis of the data.
- The expert system was a rule-based system that could make inferences about individual hosts and about the entire system (hosts and networks).
- It would then pass results to the user interface, which displayed them in a simple, easy-to-grasp manner for the system security officer.

Drawback of two system

- single point of failure.
  - If the director fails, the IDS will not function.
- Crosbie and Spafford suggest to partition the intrusion detection system into multiple components that function independently of one another, yet communicate to correlate information.

Autonomous Agents: AAFID

- An autonomous *agent* is a process that can act independently of the system of which it is a part.
- Crosbie and Spafford suggested developing autonomous agents each of which performed one particular monitoring function.
- Each agent would have its own internal model, and when the agent detected a deviation from expected behavior, a match with a particular rule, or a violation of a specification, it would notify other agents.
- The agents would jointly determine whether the set of notifications were sufficient to constitute a reportable intrusion.

Autonomous Agents: AAFID

- Advantages:
  - There a single point of failure. If one agent were compromised, the others can continue to function
  - Furthermore, if an attacker should compromise one agent, she has learned nothing about the other agents in the system or monitoring the network. Moreover, the director itself is distributed among the agents, so it cannot be attacked in the same way that an intrusion detection system with a director on a single host can be

Autonomous Agents: AAFID

- The drawbacks of autonomous agents lie in the overhead of the communications needed.
  - As the functionality of each agent is reduced, more agents are needed to monitor the system, with an attendant increase in communications overhead.
  - Furthermore, the communications must be secured, as must the distributed computations.

Intrusion Response

- Once an intrusion id detected, how can the system be protected?
- The field of intrusion response deals with this problem.
- Its goal is to handle the (attempted) attack is such a way that damage is minimized (as determined by the security policy).
- Some intrusion detection mechanisms may be augmented to thwart intrusions.
- Otherwise, the security officers must respond to the attack and attempt to repair any damage.

Intrusion Response

- **Techniques**:
  - Incident prevention
  - Intrusion handling
  - Containment phase
  - Eradication phase
  - Follow-up phase

Incident Prevention

- Incident prevention requires that the attack be identified before it completes.
- The defenders then take measures to prevent the attack from completing. This may be done manually or automatically.
- This typically involves closely monitoring the system (usually with an intrusion detection mechanism) and taking action to defeat the attack.

Intrusion Handling

- When an intrusion occurs, the security policy of the site has been violated.
- Handling the intrusion means restoring the sytem to comply with the site security policy and taking any actions against the attacker that the policy specifies.
- Intrusion handling consists of six phase:

1. ***Preparation*** for an attack: This step occurs before any attacks are detected. It establishes procedures for detecting and responding to attacks.
2. ***Identification*** of an attack: This triggers the remaining phases.

Intrusion Handling

3. ***Containment*** (confinement) of the attack. This step limits the damage as much as possible.
4. ***Eradication*** of attack. This step stops the attack an and blocks further similar attack.
5. ***Recovery*** from the attack. This step restores the system to a secure state (with respect to this site security policy).
6. ***Follow-up*** to the attack. This step involves taking action against the attacker, identifying problems in the handling of the incident, and recording lessons learned (or lesson not learned that should be learned)

Containment Phase

- Containing or confining an attack means limiting the access of the attacker to system Resources.
- The protection domain of the attacker is reduced as much as possible.
- There are two approaches:
  - passively monitoring the attack, and
  - constraining access to prevent further damage to the System.
- In this context, “damage” refers to any action that causes the system to deviate from a “secure” state as defined by the site security policy.

Containment Phase

- ***Passive monitoring*** simply records the attacker's actions for later use.
- The monitors do not interfere with the attack in any way.
- This technique is marginally useful.
  - It will reveal information about the attack and, possible, the goals of the attacker.
- However, not only is the intruded system vulnerable throughout, the attacker could attack other systems.

Containment Phase

- The other approach, in which, steps are taken to constrain the actions of the attackers, is considerably more difficult.
  - The goal is to to minimize the protection domain of the attacker while preventing the attacker from achieving his goal.
- But the system defenders may not know what the goal of the attacker is, and thus may misdirect the confinement so that the data or resources that the attacker seeks lie within the minimal protection domain of the attacker.

Eradication Phase

- Eradicating an attack means stopping the attack. The usual approach is to deny access to the system completely (such as by terminating the network connection) or to terminate the processes involved in the attack.
- An important aspect of eradication is to ensure that the attack does not immediately resume. This requires that attacks be blocked.
- A common method for implementing blocking is to place ***wrappers*** around suspected targets. The wrappers implement various forms of access control. Wrappers can control access locally on systems or control network access.

Follow-Up Phase

- In this follow-up phase, the systems take some action external to the system against the attacker.
- The most common follow-up is to pursue some form of legal action, either criminal or civil.
- The requirements of the law vary among communities, and indeed vary within communities over time.
- Counterattacking, or attacking the attacker, takes two forms.
  - Legal mechanism
    - Such as filling legal complaints
  - Technical attack
    - Goal: to damage the attacker seriously enough to stop the current attack and discourage future attacks.