**Chapter 11**

**Types 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
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
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

**Examples of Intrusion**

• 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 • Dialing into an unsecured modem and gaining internal network access • Posing as an executive, calling the help desk, resetting the executive’s email password, and learning the new password • Using an unattended, logged-in workstation without permission

**Hackers**

• Can be benign (consume resources) or malignant (damages sensitive data)

• Intrusion detection systems (IDSs) and intrusion prevention systems (IPSs) are designed to counter hacker threats

• In addition to using such systems, organizations can consider restricting remote logons to specific IP addresses and/or use virtual private network technology

• CERTs (Computer emergency response teams)

• These cooperative ventures collect information about system vulnerabilities and disseminate it to systems managers

• Hackers also routinely read CERT reports

• It is important for system administrators to quickly insert all software patches to discovered vulnerabilities

**Criminal Hackers**

• Organized groups of hackers

• Usually have specific targets, or at least classes of targets in mind

• Once a site is penetrated, the attacker acts quickly, scooping up as much valuable information as possible and exiting

• IDSs and IPSs can be used for these types of attackers, but may be less effective because of the quick in-and-out nature of the attack

**Insider Attacks**

• Among the most difficult to detect and prevent • Can be motivated by revenge or simply a feeling of entitlement

**Countermeasures**

1. Enforce least privilege, only allowing access to the resources employees need to do their job.
2. Set logs to see what users access and what commands they are entering.
3. Protect sensitive resources with strong authentication.
4. Upon termination, delete employee’s computer and network access.
5. Upon termination, make a mirror image of employee’s hard drive before reissuing it (used as evidence if your company information turns up at a competitor

**Intrusion techniques**

• Objective of the intruder is to gain access to a system or to increase the range of privileges accessible on a system

• Most initial attacks use system or software vulnerabilities that allow a user to execute code that opens a backdoor into the system

• Ways to protect a password file:

A screenshot of a computer

Description automatically generated with low confidence

**Intrusion detection**

A system’s second line of defense

• Assumes that the behavior of the intruder differs from that of a legitimate user in ways that can be quantified

• Considerations:

• If an intrusion is detected quickly enough, the intruder can be identified and ejected from the system before any damage is done or any data are compromised

• Intrusion detection enables the collection of information about intrusion techniques that can be used to strengthen the intrusion prevention facility

Diagram

Description automatically generated

**Approaches to Intrusion Detection**

**• Statistical anomaly detection**

• Involves the collection of data relating to the behavior of legitimate users over a period

• Then statistical tests are applied to observed behavior to determine whether that behavior is not legitimate user behavior

• Threshold detection - This approach involves defining thresholds, independent of user, for the frequency of occurrence of various events

• Profile based - A profile of the activity of each user is developed and used to detect changes in the behavior of individual accounts

**• Rule-based detection**

• Involves an attempt to define a set of rules or attack patterns that can be used to decide that a given behavior is that of an intruder

• Often referred to as signature detection

**Audit Records**

Diagram, text

Description automatically generated

**• Threshold detection**

• Involves counting the number of occurrences of a specific event type over an interval of time

• If the count surpasses what is considered a reasonable number that one might expect to occur, then intrusion is assumed

• By itself is a crude and ineffective detector of even moderately sophisticated attacks

**• Profile-based**

• Focuses on characterizing the past behavior of individual users or related groups of users and then detecting significant deviations

• A profile may consist of a set of parameters, so that deviation on just a single parameter may not be sufficient to signal an alert

• Techniques detect intrusion by observing events in the system and applying a set of rules that lead to a decision regarding whether a given pattern of activity is or is not suspicious

**• Rule-based anomaly detection**

• Is similar in terms of its approach and strengths to statistical anomaly detection

• Historical audit records are analyzed to identify usage patterns and to automatically generate rules that describe those patterns

• Current behavior is then observed, and each transaction is matched against the set of rules to determine if it conforms to any historically observed pattern of behavior

• For this approach to be effective, a rather large database of rules will be needed

**• Rule-based penetration identification**

• Typically, the rules used in these systems are specific to the machine and operating system

• The most fruitful approach to developing such rules is to analyze attack tools and scripts collected on the Internet

• These rules can be supplemented with rules generated by knowledgeable security personnel

**• USTAT**

• A model independent of specific audit records

• Deals in general actions rather than the detailed specific actions recorded by the UNIX auditing mechanism

• Implemented on a SunOS system that provides audit records on 239 events

**Base-Rate Fallacy**

• To be of practical use, an intrusion detection system should detect a substantial percentage of intrusions while keeping the false alarm rate at an acceptable level

• If only a modest percentage of actual intrusions are detected, the system provides a false sense of security

• If the system frequently triggers an alert when there is no intrusion, then either system managers will begin to ignore the alarms, or much time will be wasted analyzing the false alarms

• Because of the nature of the probabilities involved, it is very difficult to meet the standard of high rate of detections with a low rate of false alarms

• If the actual numbers of intrusions is low compared to the number of legitimate uses of a system, then the false alarm rate will be high unless the test is extremely discriminating

**Distributed Intrusion Detection**

• Traditional systems focused on single-system stand-alone facilities

• The typical organization, however, needs to defend a distributed collection of hosts supported by a LAN or internetwork

• A more effective defense can be achieved by coordination and cooperation among intrusion detection systems across the network

Diagram

Description automatically generated

Diagram, schematic

Description automatically generated

**Agent Architecture**

Diagram

Description automatically generated

**Honeypots**

Graphical user interface, text, application

Description automatically generated

**Example of Honey deployment**

Diagram

Description automatically generated

**Intrusion Detection exchange format**

• To facilitate the development of distributed intrusion detection systems that can function across a wide range of platforms and environments, standards are needed to support interoperability

• IETF Intrusion Detection Working Group

• Purpose of the group is to define data formats and exchange procedures for sharing information of interest to intrusion detection with response systems and to management systems that may need to interact with them

• Have issued the following RFCs:

• Intrusion Detection Message Exchange Requirements (RFC 4766)   
• The Intrusion Detection Message Exchange Format (RFC 4765)

• The Intrusion Detection Exchange Protocol (RFC 4767)

**Model for Intrusion Detection exchange format**

Diagram

Description automatically generated

**Password Management**

• Front line of defense against intruders

• Virtually all multiuser systems require that a user provide not only a name or identifier (ID) but also a password

• Password serves to authenticate the ID of the individual logging on to the system

• The ID provides security by:

• Determining whether the user is authorized to gain access to a system

• Determining the privileges accorded to the user

• Used in discretionary access control

**Attack Strategies and Countermeasures**

1. **Workstation hijacking -** • The attacker waits until a logged-in workstation is unattended • The standard countermeasure is automatically logging the workstation out after a period of inactivity
2. **Exploiting user mistakes -** • Attackers are frequently successful in obtaining passwords by using social engineering tactics that trick the user or an account manager into revealing a password; a user may intentionally share a password to enable a colleague to share files; users tend to write passwords down because it is difficult to remember them • Countermeasures include user training, intrusion detection, and simpler passwords combined with another authentication mechanism
3. **Offline dictionary attack -** • Determined hackers can frequently bypass access controls and gain access to the system’s password file • Countermeasures include controls to prevent unauthorized access to the password file, intrusion detection measures to identify a compromise, and rapid reissuance of passwords should the password file be compromised
4. **Specific account attack -** • The attacker targets a specific account and submits password guesses until the correct password is discovered • The standard countermeasure is an account lockout mechanism, which locks out access to the account after several failed login attempts
5. **Electronic monitoring** - • If a password is communicated across a network to log on to a remote system, it is vulnerable to eavesdropping • Simple encryption will not fix this problem, because the encrypted password is, in effect, the password and can be observed and reused by an adversary
6. **Password guessing against single user** - • The attacker attempts to gain knowledge about the account holder and system password policies and uses that knowledge to guess the password • Countermeasures include training and enforcement of password policies that make passwords difficult to guess
7. **Exploiting multiple password use** - • Attacks can become much more effective or damaging if different network devices share the same or a similar password for a given user • Countermeasures include a policy that forbids the same or similar password on network devices
8. **Popular password attack** - • Attack is to use a popular password and try it against a wide range of user IDs • Countermeasures include policies to inhibit the selection by users of common passwords and scanning the IP addresses of authentication requests and client cookies for submission patterns

Diagram

Description automatically generated

**Unix implementations**

**• Crypt(3)**

• Was designed to discourage guessing attacks • This implementation is now considered inadequate • Despite its known weaknesses, this UNIX scheme is still often required for compatibility with existing account management software or in multivendor environments

**• MD5 secure hash algorithm**

• The recommended hash function for many UNIX systems, including Linux, Solaris, and FreeBSD • Far slower than crypt(3)

**• Bcrypt**

• Developed for OpenBSD • Probably the most secure version of the UNIX hash/salt scheme • Uses a hash function based on the Blowfish symmetric block cipher • Slow to execute • Includes a cost variable

Text, letter

Description automatically generated

**• Impact:** – the consequence of a security attack where a vulnerability has been successfully exploited. – Impact is derived from the value of the assets being protected.

• **Exploitability** – a measure of how easy it is for an attacker to successfully exploit a vulnerability. – For example, it is easier to correctly guess a weak password than it is to guess a strong password. – Exploitability is derived from the vulnerabilities in the system.

**• Key Characteristics of a vulnerability:**

– Product(s): This is the set of products/software system where the vulnerability is applicable or that are affected by this vulnerability e.g., Window 10 Home Edition, Acrobat Reader Version 5.6.2, etc.

– Exploitability(P) Score – same as defined in the previous slide

– Impact Score(I) – see previous slide

– Threats: This is the set of threats posed by the vulnerability e.g., a weak password for your Blackboard account poses the threat of Information Disclosure(I).

Text

Description automatically generated with low confidence

**Stride threat modelling framework**

Table

Description automatically generated

Diagram

Description automatically generated

Table

Description automatically generated

Diagram

Description automatically generated with medium confidence

Table

Description automatically generated

Table

Description automatically generated