5.1 INTRUSION DETECTION

INTRUDER

An intruder refers to an unauthorized person or entity attempting to gain access to a computer system, network, or data without permission. Generally referred to as hacker or cracker.

Three classes of intruders are as follows:

1. **Masquerader** – an individual who is not authorized to use the computer and whopenetrates 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 misuse 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.

The masquerader is likely to be an outsider; the misfeasor generally is an insider; and the clandestine user can be either an outsider or an insider.

Intrusion Techniques

The objective of the intruders is to gain access to a system or to increase the range of privileges accessible on a system.

The password files can be protected in one of the two ways:

One-way encryption – The system stores only an encrypted form of user's password.

Access control – Access to the password file is limited to one or a very few accounts.

The following techniques are used for learning passwords.

1.Try default passwords used with standard accounts that are shipped with the system.

Many administrators do not bother to change these defaults.

2. Exhaustively try all short passwords.

3. Try words in the system's online dictionary or a list of likely passwords.

4. Collect information about users such as their full names, the name of their spouse and children, pictures in their office and books in their office that are related to hobbies.

5. Try user's phone number, social security numbers and room numbers.

6. Try all legitimate license plate numbers.

7. Use a torjan horse to bypass restriction on access.

8. Tap the line between a remote user and the host system.

Two principle countermeasures:

1. Detection – concerned with learning of an attack, either before or after its success.
2. Prevention – challenging security goal

INTRUSION DETECTION

Intrusion detection is based on the assumption that the behaviour of the intruder differs from that of alegitimate user in ways that can be quantified. Although the typical behaviour of an intruder differs from the typical behaviour of an authorized user, there is an overlap in these behaviours.

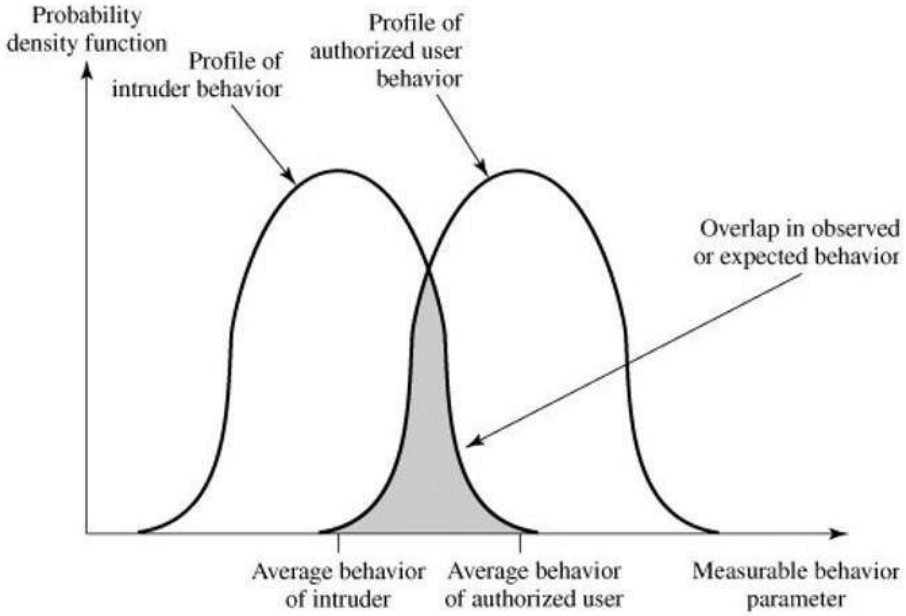

Figure: Profiles of Behavior of Intruders and Authorized Users

Approaches to intrusion detection

1. Statistical anomaly detection
2. Rule-based detection:

3. Distributed Intrusion Detection
4. Honeypot

1. Statistical anomaly detection:

Involves the collection of data relating to the behaviour of legitimate users over a period of time.

Then statistical tests are applied to observed behaviour to determine with a high level of confidence whether that behaviour is not legitimate user behaviour.

a) Threshold detection: This approach involves defining thresholds, independent of user,for the frequency of occurrence of various events.

b) Profile based: A profile of the activity of each user is developed and used to detect changes in the behaviour of individual accounts.

2. Rule-based detection:

Involves an attempt to define a set of rules that can be used to decide that a given behaviour is that of an intruder.

a) **Anomaly detection**: Rules are developed to detect deviation from previous usage patterns.

b) **Penetration identification**: An expert system approach that searches for suspicious behaviour.

A fundamental tool for intrusion detection is the **audit record.** Some record of ongoing activity by users must be maintained as input to an intrusion detection system.

Basically, two plans are used:

1.**Native audit records**: Virtually all multiuser operating systems include accounting software that collects information onuser activity. The advantage of using this information is that no additional collection software is needed.

2.**Detection-specific audit records**: A collection facilitycan be implemented that generates audit records containing only that information required by the intrusion detection system. The disadvantage is the extra overhead involved in having, in effect, two accounting packages running on a machine.

Each audit record contains the following fields:

• Subject: Initiators of actions. A subject is typically a terminal user but might also be a process acting on behalf of users or groups of users.

• Action: Operation performed by the subject on or with an object; for example, login, read, perform I/O, execute.

• Object: Receptors of actions. Examples include files, programs, messages, records, terminals, printers, and user- or program-created structures

• Exception-Condition: Denotes which, if any, exception condition is raised on return.

• Resource-Usage: A list of quantitative elements in which each element gives the amount used of some resource

• Time-Stamp: Unique time-and-date stamp identifying when the action took place.

3. Distributed Intrusion Detection

Traditional focus is on single systems. But typically have networked systems. More effective defense has these working together to detect intrusions

- Dealing with varying audit record formats
- Integrity & confidentiality of networked data
- Centralized or decentralized architecture

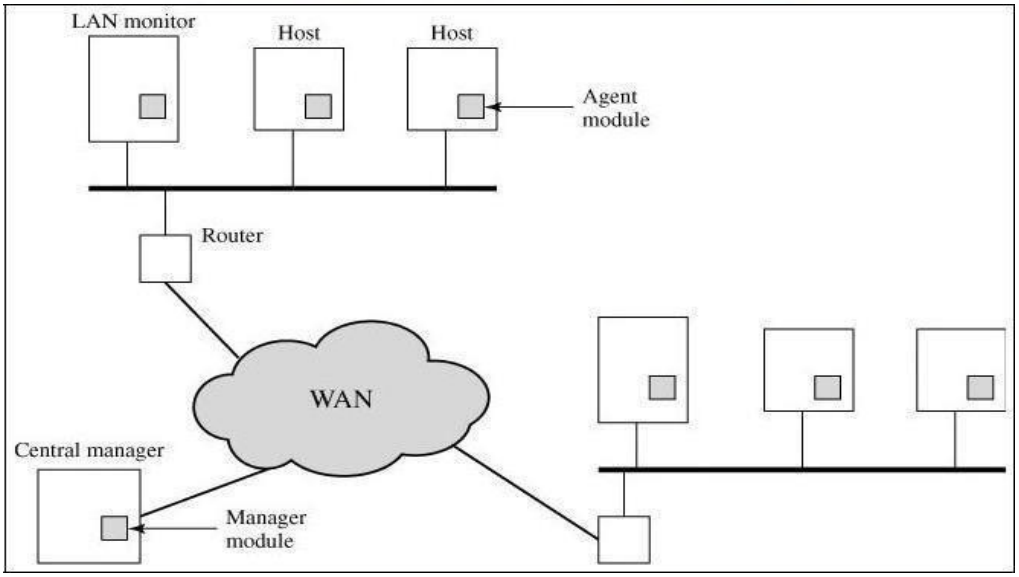

Figure. Architecture for Distributed Intrusion Detection

Three main components

1. **Host agent module**: An audit collection module operating as a background process on a monitored system. Its purpose is to collect data on security-related events on the host and transmit these to the central manager.

2. **LAN monitor agent module**: Operates in the same fashion as a host agent module except that it analyses LAN traffic and reports the results to the central manager.

3. **Central manager module**: Receives reports from LAN monitor and host agents and processes and correlates these reports t2o detect intrusion.

Agent Architecture

The agent captures each audit record produced by the native audit collection system.

1. Filter is applied that retains only those records that are of security interest.
2. These records are then reformatted into a standardized format referred to as the host audit record (HAR).
3. Next, a template-driven logic module analyzes the records for suspicious activity.
4. At the lowest level, the agent scans for notable events that are of interest independent of any past events.
5. At the next higher level, the agent looks for sequences of events, such as known attack patterns (signatures).
6. Finally, the agent looks for anomalous behaviour of an individual user based on a historical profile of that user, such as number of programs executed, number of files accessed, and the like.
7. When suspicious activity is detected, an alert is sent to the central manager.
8. The central manager includes an expert system that can draw inferences from received data.
9. The manager may also query individual systems for copies of HARs to correlate with those from other agents.
10. The LAN monitor agent also supplies information to the central manager.

11. The LAN monitor agent audits host-host connections, services used, and volume of traffic.

12. It searches for significant events, such as sudden changes in network load, the use of security-related services, and network activities such as rlogin.

4. Honeypots

Honeypots are decoy systems that are designed to lure a potential attacker away from critical systems. Honeypots are designed to

- Divert an attacker from accessing critical systems
- Collect information about the attacker's activity
- Encourage the attacker to stay on the system long enough for administrators to respond