CSCI 7000-001 — Principles of Intrusion Detection

December 4, 2001

1 Motivations

So far, we have studied (mostly) *prevention* mechanisms.
In other words, we have studied how to make it difficult, or impossible, for bad things to happen (or for bad guys to cause bad things to happen).
However, we must keep in mind a fundamental principle about computer and network security:

- security requires an *engineering approach*
- security must be dealt with on may fronts and at many levels
- different fronts and levels have different associated costs and benefits

In other words, *bad things happen!*
No matter how smart your prevention mechanism is. Also, your smart prevention mechanism may have be very expensive (in terms of resources, inconveniences, etc.).

2 Intrusion detection

Intrusion detection is a second line of defense. Or not even that:

- intrusion detection can recognize a threat (e.g., an intruder) before it can actually cause any damage

- even if it doesn't prevent bad things to happen it may put pressure on attackers and in any case facilitate the identification of attackers, and therefore act as a deterrent
- in any case, it provides useful information for forensic analysis, and ultimately feeds back to strengthen the prevention mechanisms

3 Basic principles

- the goal is to *detect* intrusions or intrusion attempts
- the hypothesis is that intruders behave in a way that is *noticeably different* from legitimate users

4 Basic approaches

As usual, there are two complementary approaches

- define good behaviors, and detect when something or someone doesn't follow those rules;
- define bad behavior, and detect when something or someone starts following those rules

5 How to define behaviors

Traditionally two classes of methods:

- statistical models of behavior
- rule-based or state-transition models

6 Auditing

The primary enabling technology for intrusion detection is *auditing*.

- a trace of security-sensitive events in a system

- an event of interest is logged with an *audit record*
- the object that emits audit records is usually referred to as *probe* or *sensor*
- the concept of "event" is broadly defined: events range from OS activities (e.g., user X reads file F) to network activities (e.g., host H receives message M on interface I)

7 Audit records: classical definition

As originally defined by [Denning 87], audit records describe elementary actions in the context of a single computer. Therefore, at a minimum, an audit record should contain:

- *subject*: entity executing the recorded action (e.g., user, process, terminal)
- *action*: operation executed (e.g., login, read-file, execute, logout, etc.)
- *object*: entity receiving the action (file, process, user, terminal, etc.)
- *exceptions*: possible abnormal results or error conditions caused by the action
- *resource usage*: annotation of specific resources consumed by the action (e.g., CPU time, bytes read or written, etc.)
- *time-stamp*

8 Audit records: current approaches

Intrusion detection has been a very active research field in the past ten years. Many different intrusion detection and response systems have been developed, and along with them, many audit record formats have been proposed.
The *Intrusion Detection Message Exchange Format (IDMEF)* is a recent effort aimed at unifying audit records as well as intrusion detection alarms into a common format.

9 Statistical behavior definition

The idea is to measure activities using simple metrics:

- *counters*: e.g., number of consecutive failed (or successful) logins
- *gauges*: e.g., number of active processes or active sessions for a specific user
- *durations*: e.g., time interval between two login for the same user
- *resource utilization*: e.g., CPU time used per day

Once we have those measures, we can distinguish good vs. bad behavior by defining

- thresholds (ranges)
- mean value and standard deviation
- other distributions
- time progressions

This class of techniques is typically used to define acceptable behaviors. Therefore, intrusion (anomaly) detection reduces to the identification of behaviors that exceed prescribed ranges or distributions.

10 Rule-based or state-transition models

Another idea is to identify *signatures*. That is, to identify characteristic sequences of events that identify a specific activity. These models are more frequently used to describe bad behaviors (intruders). There are two flavors of these models:

- rule bases derived from historical observations
- rule bases derived from analysis and knowledge of specific vulnerabilities

With these systems, intrusion detection boils down to "parsing" audit logs looking for the appropriate signatures.

11 Combined approaches

As usual, the most effective approach is to mix and match different techniques. Typically, statistical description of positive behaviors are combined with signature-based descriptions of negative behaviors.

12 Distributed intrusion detection

A natural extension of system- or host-based intrusion detection systems consists in monitoring several hosts across a network.
The essence of this approach is conceptually identical to traditional intrusion detection. The difference is that audit traces of several hosts are merged in a single audit trace.
Keep in mind that the focus here remains on system-specific attacks. This technique is also called *distributed intrusion detection*.

13 Network-based intrusion detection

Another rather different approach is to extend the idea of intrusion detection to networks. The focus here is on network-wide and network-specific attacks, or attacks that make use of the network. For example:

- buffer-overflow attacks on network applications
- spoofing attacks
- distributed denial-of service attacks

This technique is called *network-based intrusion detection*. Although the term "intrusion" is probably too restrictive.
The fundamental ingredients of network-based intrusion detection are:

- model of the attacks (same as model of bad behavior in traditional intrusion detection)
- model of the network, including:
  - topological model

- service model (what network-sensitive application or what network service is running on what host)

The difference with respect to traditional or distributed intrusion detection is that network-based intrusion detection requires and develops an additional fundamental technique to place probes in the appropriate place, depending on the combination of topology and services of the observed network, and based on the models of observed attacks.