

# Chapter 8

## Intrusion Detection

# Classes of Intruders – Cyber Criminals

- Individuals or members of an organized crime group with a goal of financial reward
- Their activities may include:
  - Identity theft
  - Theft of financial credentials
  - Corporate espionage
  - Data theft
  - Data ransomware
- Typically they are young, often Eastern European, Russian, or southeast Asian hackers, who do business on the Web
- They meet in underground forums to trade tips and data and

# Classes of Intruders – Activists

- Are either individuals, usually working as insiders, or members of a larger group of outsider attackers, who are motivated by social or political causes
- Also known as hacktivists
  - Skill level is often quite low
- Aim of their attacks is often to promote and publicize their cause typically through:
  - Website defacement
  - Denial of service attacks
  - Theft and distribution of data that results in negative publicity

# Classes of Intruders – State-Sponsored Organizations

Groups of  
hackers  
sponsored  
by

Also known as  
Advanced Persistent  
Threats (APTs)

due to the  
Widespread  
covert  
nature  
and  
persistence  
scope of  
these  
extended  
activities

by a  
wide  
range of  
countries  
from  
China to  
the USA

# Classes of Intruders – Others

- Hackers with motivations other than those previously listed
- Include classic hackers or crackers who are motivated by technical challenge or by peer-group esteem and reputation
- Many of those responsible for discovering new categories of buffer overflow vulnerabilities could be regarded as members of this class
- Given the wide availability of attack toolkits, there is a pool of “hobby hackers” using them to explore system and network security

# Intruder Skill Levels – Apprentice

- Hackers with minimal technical skill who primarily use existing attack toolkits
- They likely comprise the largest number of attackers, including many criminal and activist attackers
- Given their use of existing known tools, these attackers are the easiest to defend against
- Also known as “script-kiddies” due to their use of existing scripts (tools)

# Intruder Skill Levels – Journeyman

- Hackers with sufficient technical skills to modify and extend attack toolkits to use newly discovered, or purchased, vulnerabilities
- They may be able to locate new vulnerabilities to exploit that are similar to some already known
- Hackers with such skills are likely found in all intruder classes
- Adapt tools for use by others

# Intruder Skill Levels – Master

- Hackers with high-level technical skills capable of discovering brand new categories of vulnerabilities
- Write new powerful attack toolkits
- Some of the better known classical hackers are of this level
- Some are employed by state-sponsored organizations
- Defending against these attacks is of the highest difficulty



# Examples of Intrusion

- Remote root compromise
- Web server defacement
- Guessing/cracking passwords
- Copying databases containing credit card numbers
- Viewing sensitive data without authorization
- Running a packet sniffer
- Distributing pirated software
- Using an unsecured modem to access internal network
- Impersonating an executive to get information

# Intruder Behavior

**Target acquisition and  
information gathering**

**Initial access**

**Privilege escalation**

**Information gathering  
or system exploit**

**Maintaining access**

**Covering tracks**

### **(a) Target Acquisition and Information Gathering**

- Explore corporate website for information on corporate structure, personnel, key systems, as well as details of specific web server and OS used.
- Gather information on target network using DNS lookup tools such as dig, host, and others; and query WHOIS database.
- Map network for accessible services using tools such as NMAP.
- Send query email to customer service contact, review response for information on mail client, server, and OS used, and also details of person responding.
- Identify potentially vulnerable services, eg vulnerable web CMS.

### **(b) Initial Access**

- Brute force (guess) a user's web content management system (CMS) password.
- Exploit vulnerability in web CMS plugin to gain system access.
- Send spear-phishing email with link to web browser exploit to key people.

### **(c) Privilege Escalation**

- Scan system for applications with local exploit.
- Exploit any vulnerable application to gain elevated privileges.
- Install sniffers to capture administrator passwords.
- Use captured administrator password to access privileged information.

### **(d) Information Gathering or System Exploit**

- Scan files for desired information.
- Transfer large numbers of documents to external repository.
- Use guessed or captured passwords to access other servers on network.

### **(e) Maintaining Access**

- Install remote administration tool or rootkit with backdoor for later access.
- Use administrator password to later access network.
- Modify or disable anti-virus or IDS programs running on system.

### **(f) Covering Tracks**

- Use rootkit to hide files installed on system.
- Edit logfiles to remove entries generated during the intrusion.

## Table 8.1

# Examples of Intruder Behavior

und on pages 255-256 in the

# Definitions

- Security Intrusion:

Unauthorized act of bypassing the security mechanisms of a system

- Intrusion Detection:

A hardware or software function that gathers and analyzes information from various areas within a computer or a network to identify possible security intrusions

# Intrusion Detection System (IDS)

- Host-based IDS (HIDS)
    - Monitors the characteristics of a single host for suspicious activity
  - Network-based IDS (NIDS)
    - Monitors network traffic and analyzes network, transport, and application protocols to identify suspicious activity
  - Distributed or hybrid IDS
    - Combines information from a number of sensors, often both host and network based, in a central analyzer that is able to better identify and respond to intrusion activity
- **Comprises three logical components:**
    - **Sensors - collect data**
    - **Analyzers - determine if intrusion has occurred**
    - **User interface - view output or control system behavior**

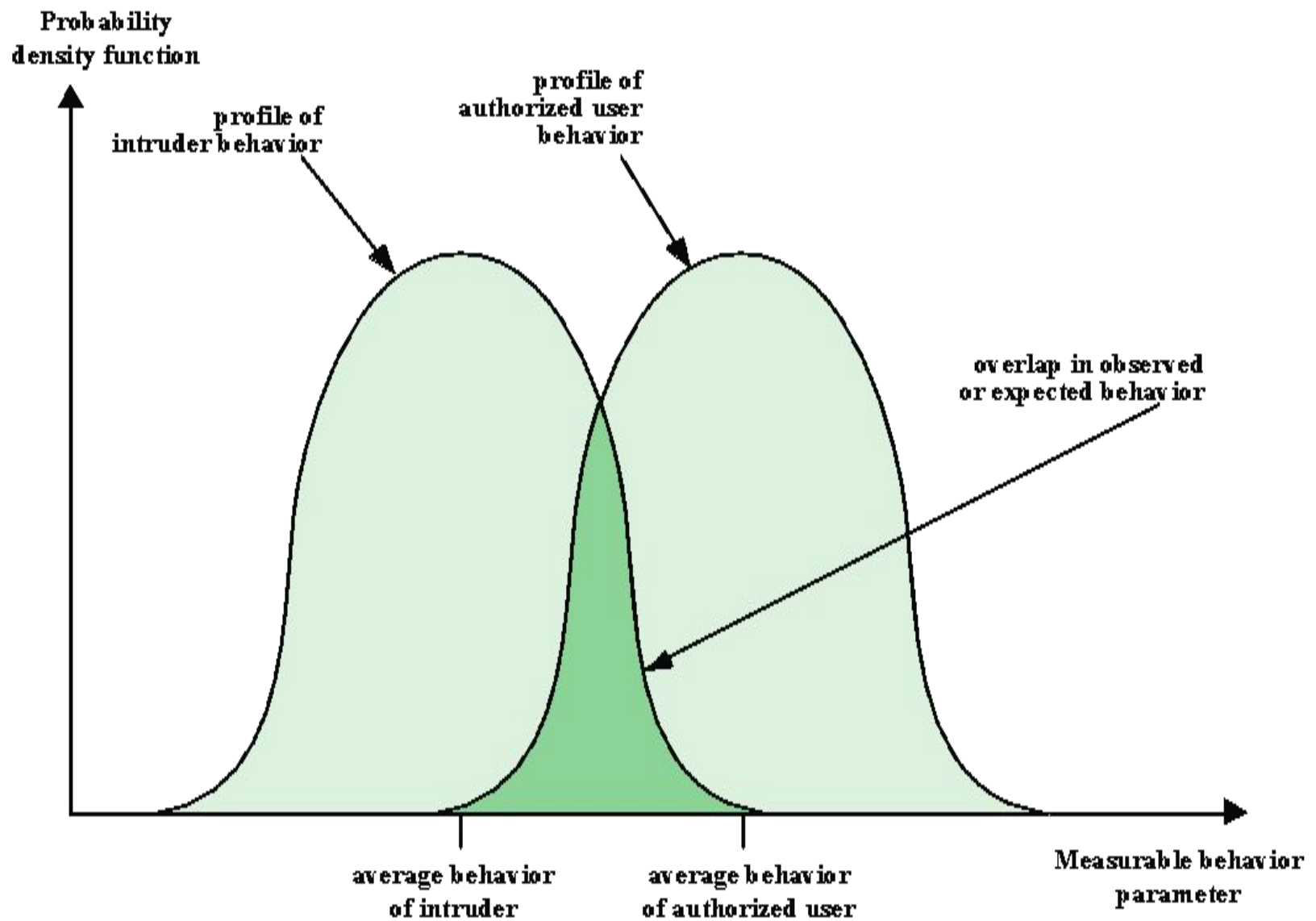


Figure 8.1 Profiles of Behavior of Intruders and Authorized Users

# IDS Requirements

**Run continually**

**Be fault tolerant**

**Resist subversion**

**Impose a  
minimal  
overhead on  
system**

**Configured  
according to  
system security  
policies**

**Adapt to  
changes in  
systems and  
users**

**Scale to monitor  
large numbers  
of systems**

**Provide graceful  
degradation of  
service**

**Allow dynamic  
reconfiguration**

# Analysis Approaches

## Anomaly detection

- Involves the collection of data relating to the behavior of legitimate users over a period of time
- Current observed behavior is analyzed to determine whether this behavior is that of a legitimate user or that of an intruder

## Signature/Heuristic detection

- Uses a set of known malicious data patterns or attack rules that are compared with current behavior
- Also known as misuse detection
- Can only identify known attacks for which it has patterns or rules



# Anomaly Detection

A variety of classification approaches are used:

## Statistical

- Analysis of the observed behavior using univariate, multivariate, or time-series models of observed metrics

## Knowledge based

- Approaches use an expert system that classifies observed behavior according to a set of rules that model legitimate behavior

## Machine-learning

- Approaches automatically determine a suitable classification model from the training data using data mining techniques

# Signature or Heuristic Detection

## Signature approaches

Match a large collection of known patterns of malicious data against data stored on a system or in transit over a network

The signatures need to be large enough to minimize the false alarm rate, while still detecting a sufficiently large fraction of malicious data

Widely used in anti-virus products, network traffic scanning proxies, and in NIDS

## Rule-based heuristic identification

Involves the use of rules for identifying known penetrations or penetrations that would exploit known weaknesses

Rules can also be defined that identify suspicious behavior, even when the behavior is within the bounds of established patterns of usage


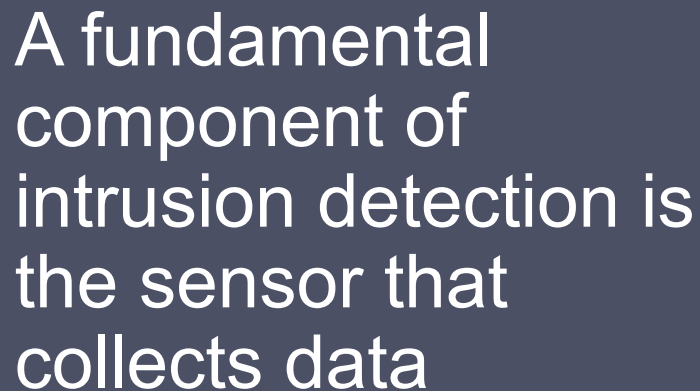
Typically rules used are specific

SNORT is an example of a rule-based NIDS

# Host-Based Intrusion Detection (HIDS)

- Adds a specialized layer of security software to vulnerable or sensitive systems
- Can use either anomaly or signature and heuristic approaches
- Monitors activity to detect suspicious behavior
  - Primary purpose is to detect intrusions, log suspicious events, and send alerts
  - Can detect both external and internal intrusions

# Data Sources and Sensors



A fundamental component of intrusion detection is the sensor that collects data

Common data sources include:

- System call traces
- Audit (log file) records
- File integrity checksums
- Registry access

### (a) Ubuntu Linux System Calls

accept, access, acct, adjtime, aiocancel, aioread, aiowait, aiowrite, alarm, async\_daemon, auditsys, bind, chdir, chmod, chown, chroot, close, connect, creat, dup, dup2, execv, execve, exit, exportfs, fchdir, fchmod, fchown, fchroot,fcntl, flock, fork, fpathconf, fstat, fstat, fstatfs, fsync, ftime, fruncate, getdents, getdirentries, getdomainname, getdopt, getdtablesize, getfh, getgid, getgroups, gethostid, gethostname, getitimer, getmsg, getpagesize, getpeername, getpgrp, getpid, getpriority, getrlimit, getrusage, getsockname, getsockopt, gettimeofday, getuid, gttty, ioctl, kill, killpg, link, listen, lseek, lstat, madvise, mctl, mincore, mkdir, mknod, mmap, mount, mount, mprotect, mpxchan, msgsys, msync, munmap, nfs\_mount, nfssvc, nice, open, pathconf, pause, pcfs\_mount, phys, pipe, poll, profil, ptrace, putmsg, quota, quotactl, read, readlink, readv, reboot, recv, recvfrom, recvmsg, rename, resuba, rfssys, rmdir, sbreak, sbrk, select, semsys, send, sendmsg, sendto, setdomainname, setdopt, setgid, setgroups, sethostid, sethostname, setitimer, setpgid, setpgrp, setpgrp, setpriority, setquota, setregid, setreuid, setrlimit, setsid, setsockopt, settimeofday, setuid, shmsys, shutdown, sigblock, sigpause, sigpending, sigsetmask, sigstack, sigsys, sigvec, socket, socketaddr, socketpair, sstk, stat, stat, statfs, stime, stty, swapon, symlink, sync, sysconf, time, times, truncate, umask, umount, uname, unlink, unmount, ustat, utime, utimes, vadvice, vfork, vhangup, vlimit, vpixsys, vread, vtimes, vtrace, vwrite, wait, wait3, wait4, write, writev

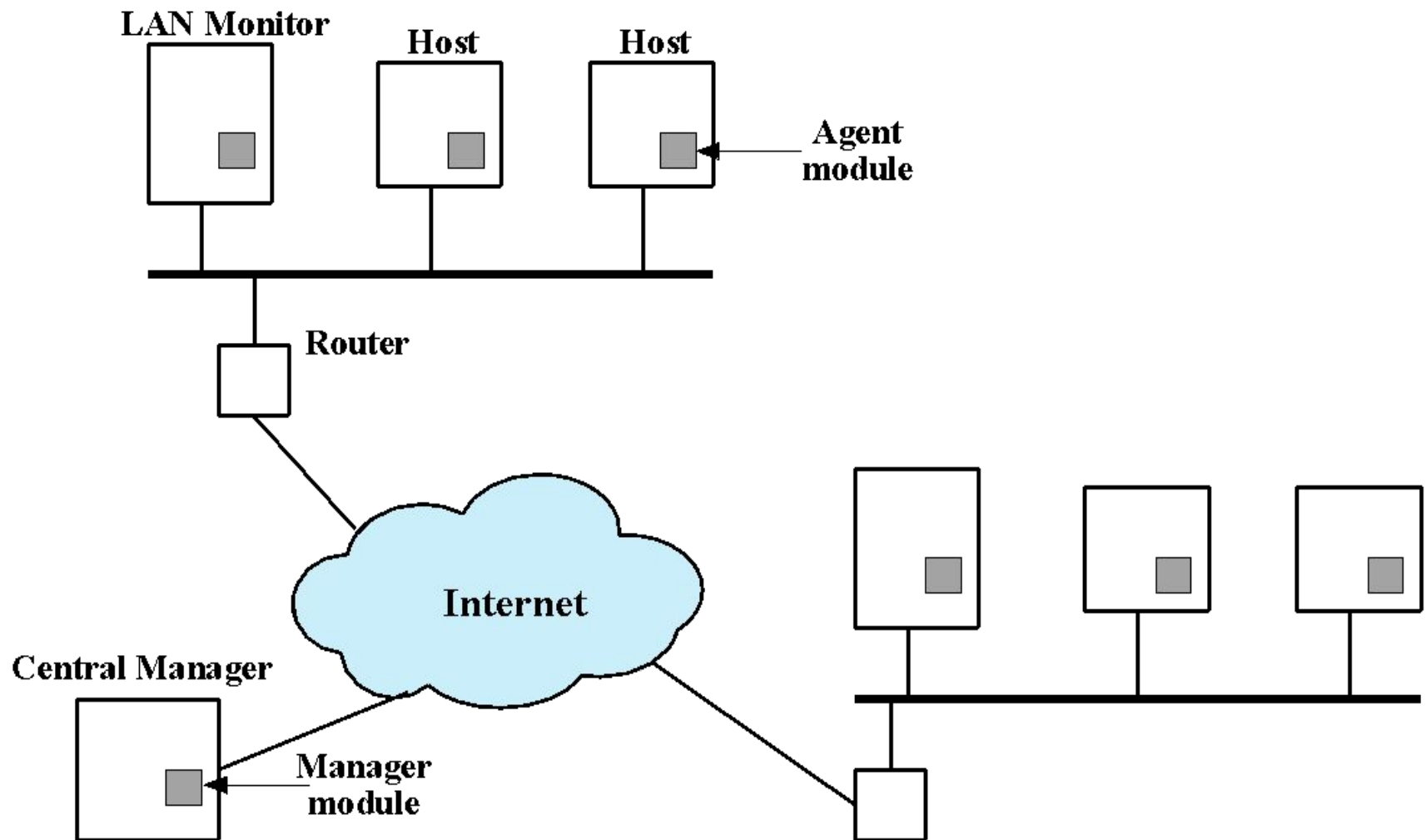
### (b) Key Windows DLLs and Executables

comctl32  
kernel32  
msvcpp  
msvert  
mswsock  
ntdll  
ntoskrnl  
user32  
ws2\_32

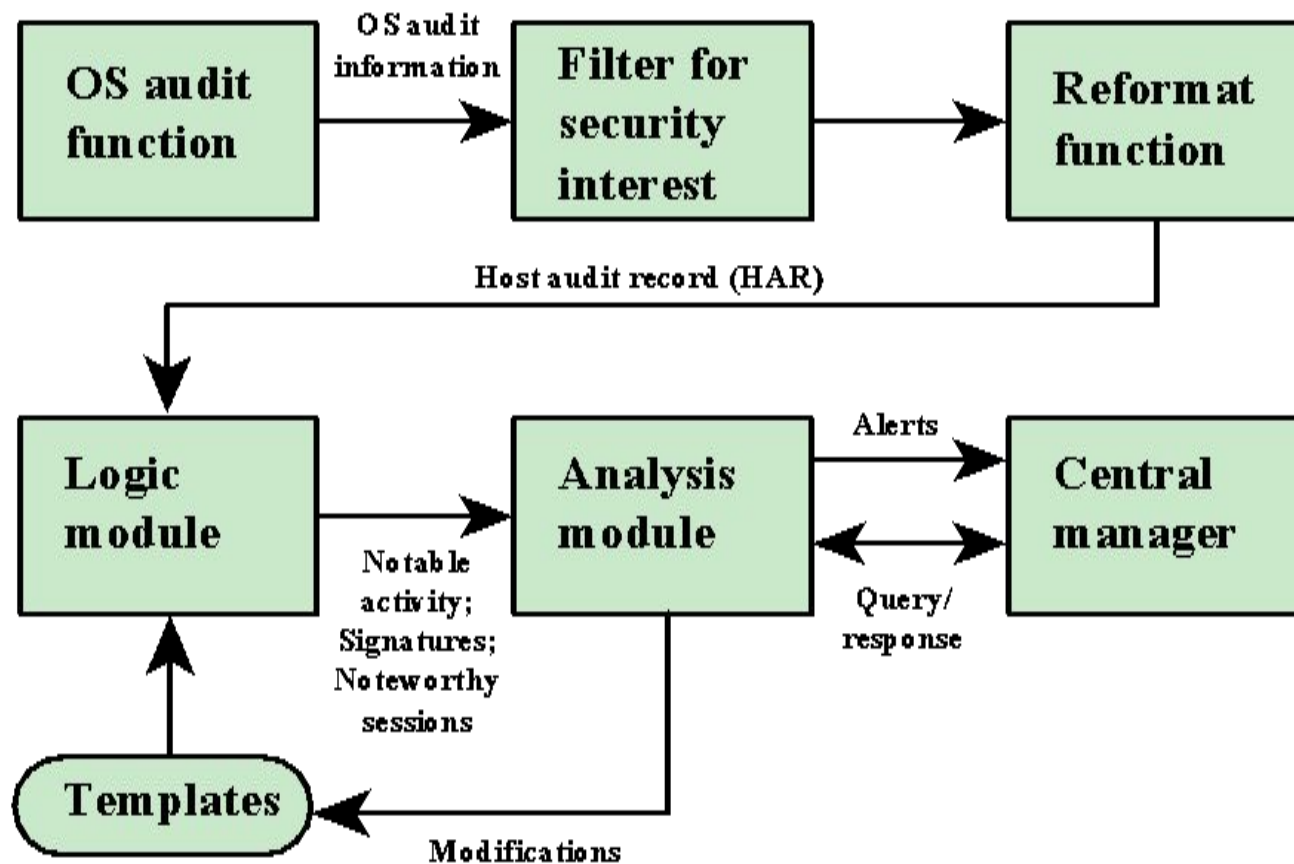
# Table 8.2

## Linux System Calls and Windows DLLs Monitored

(Table can be found on page 264 in the textbook)



**Figure 8.2 Architecture for Distributed Intrusion Detection**



**Figure 8.3 Agent Architecture**

# Network-Based IDS (NIDS)

Monitors traffic at selected points on a network

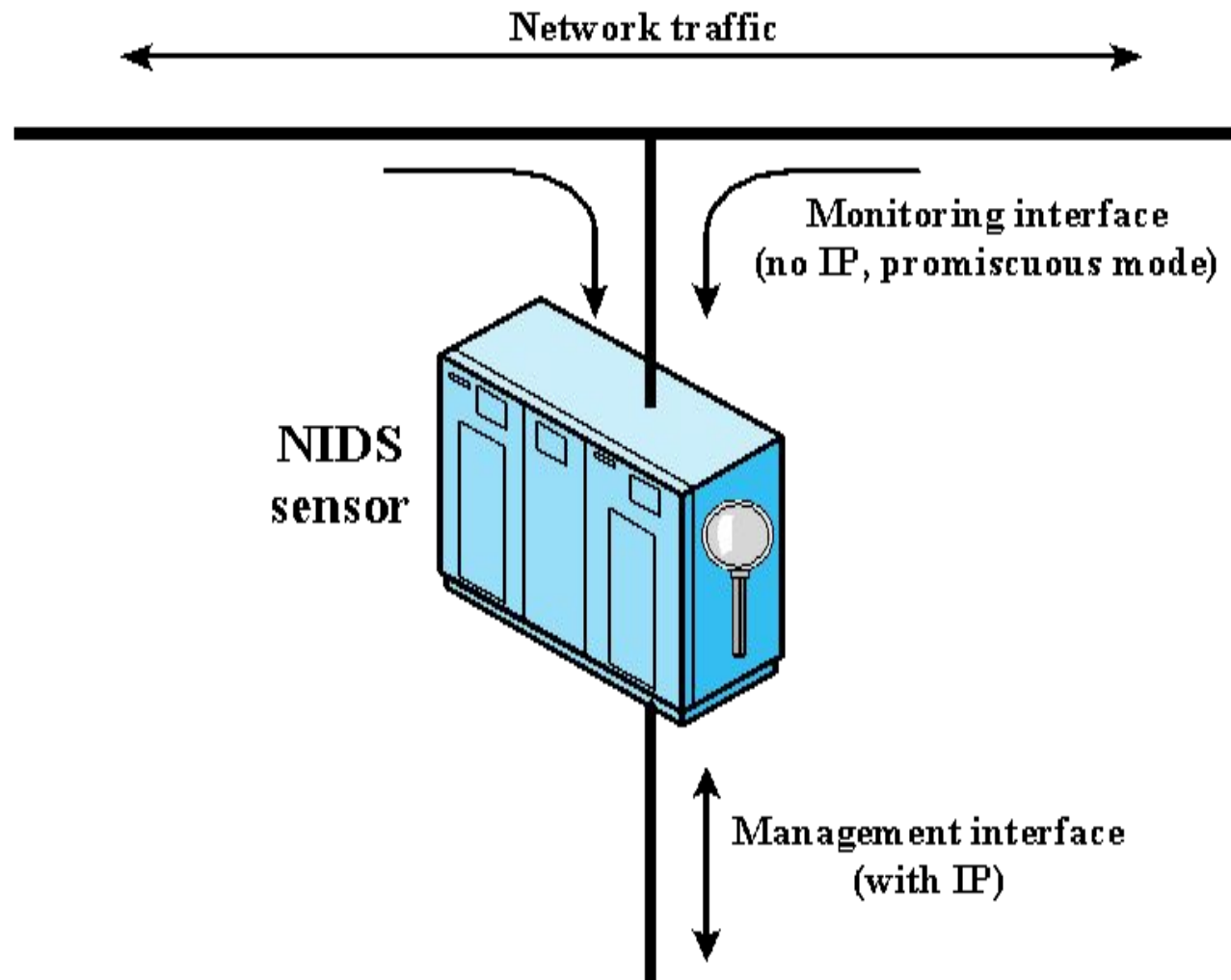
Examines traffic packet by packet in real or close to real time

May examine network, transport, and/or application-level protocol activity

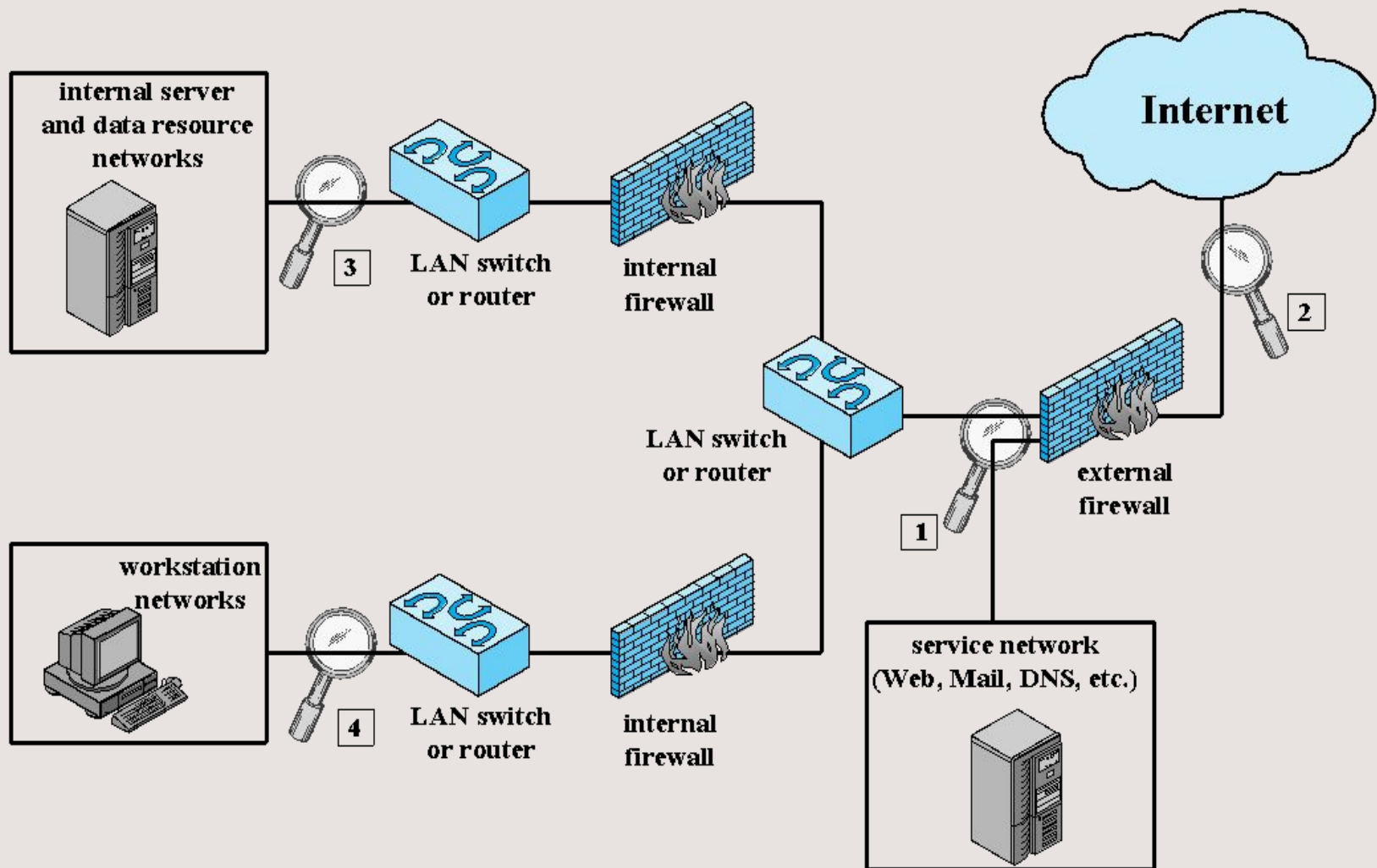
Comprised of a number of sensors, one or more servers for NIDS management functions, and one or more management consoles for the human interface

Analysis of traffic patterns may be done at the sensor, the management server or a combination of the two





**Figure 8.4 Passive NIDS Sensor**



**Figure 8.5 Example of NIDS Sensor Deployment**

# Intrusion Detection Techniques

## Attacks suitable for Signature detection

- Application layer reconnaissance and attacks
- Transport layer reconnaissance and attacks
- Network layer reconnaissance and attacks
- Unexpected application services
- Policy violations

## Attacks suitable for Anomaly detection

- Denial-of-service (DoS) attacks
- Scanning
- Worms