INTRUDERS

Tran Song Dat Phuc
Department of Computer Science and Engineering
SeoulTech 2014

Intruders

Is one of the two most publicized threats to security (the other is viruses)

Intruders

- Anderson [ANDE80] identified three classes of intruders:

Masquerader	Misfeasor	Clandestine user
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.	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.	An individual who seizes supervisory control of the system and uses this control to evade auditing and access controls or to suppress audit collection.

Intruders

- Intruder attacks range from the benign to the serious.
- At the benign, people just simply want to explore internets and see what is out there.
- At the serious, people attempt to read privileged data, perform unauthorized modifications to data, or disrupt the system.

Intruders

- [GRAN04] lists some examples of intrusion, consists of:
  - Performing a remote root compromise of an email 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
  - Using an unattended, logged-in workstation without permission …

Intruder Behavior Patterns

Intruder Behavior Patterns

- The techniques and behavior patterns of intruders are constantly shifting, to exploit newly discovered weaknesses and to evade detection and countermeasures.

- **HACKERS:**
  - Those who hack into computers do so for the thrill of it or for status.
  - Attackers often look for targets of opportunity and share the information with others within the hacking community.
  - The intruder took advantage of the fact that the corporate network was running unprotected services.
  - The key to the break-in was the PCAnywhere application.
  - The intruder can discover when a vice president walk into his office as well as see the files on his Windows workstation.

Intruder Behavior Patterns

- **HACKERS:**
  - Benign intruders might be tolerable, they just consume resources and may slow performance for legitimate users.
  - Serious (malign) intruders may lead to big damage, especially in official or government systems.

Gary McKinnon infiltrated into U.S. government computer networks in late 2001 and early 2002. He installed hacking software, deleted important files and stole information about UFOs.

Intruder Behavior Patterns

- **HACKERS:**
  - Benign intruders might be tolerable, they just consume resources and may slow performance for legitimate users.
  - Serious (malign) intruders may lead to big damage for network, especially in official or government systems.

The famous assassinating President Kennedy 1963.

Intruder Behavior Patterns

- Intrusion detection systems (**IDS**s) and intrusion prevention systems (**IPS**s) are designed to counter this type of hacker threat.
- Organizations can consider restricting remote logons to specific IP addresses and/or use virtual private network technology.
- Computer emergency response teams (CERTs) have established with the cooperative ventures collect information about system vulnerabilities and disseminate it to systems managers.
- The systems administrators will quickly insert all software patches to discover and fix those vulnerabilities.
- Ex: the versions of jailbreak (offer officially by a third party) on Iphone, Ipod, Ipad … devices using standard iOS help Apple Inc. find out vulnerabilities of security and fix them in next updating of its software.

Intruder Behavior Patterns

- **CRIMINALS:**
  - Organized groups of hackers have become a widespread and common threat to Internet-based systems.
  - Oftenly, attackers cover underground forums to trade tips and data and coordinate attacks.
  - A common target is a credit card file at an e-commerce server. Attackers attemp to gain root access.
  - The card numbers are used to purchase expensive items, and then posted in carder sites, where others can access and continue use it.
  - IDSs and IPSs can be used for these types of attackers, but maybe less effective because of the quick in-and-out nature of the attack.

Intruder Behavior Patterns

- **CRIMINALS:**

2013 $45-million ATM cyber looting

Intruder Behavior Patterns

- **CRIMINALS:**
  - For e-commerce sites, database encryption should be used for sensitive customer information, especially credit cards.
  - E-commerce organization should use dedicated server (not support multiple customers) and closely monitor the provider's security services.

- **INSIDER ATTACKS:**
  - Among the most difficult to detect and prevent.
  - Those who already have access and knowledge about the structure and content of corporate database.
  - Can be motivated by revenge or certain special reasons, such as feeling of entitlement ...

Intruder Behavior Patterns

- **INSIDER ATTACKS:**
- 2013, Edward Snowden, a computer specialist, former employee of CIA and NSA, disclosed thousands of classified documents to the media. The leaked documents have weakened national security.

Intruder Behavior Patterns

- **INSIDER ATTACKS:**
- IDSs and IPSs can be useful to counter this attack, combine with some approaches as follows:
- Enforce least privilege, only allowing access to the resources employees need to do their job.
- Set logs to see what users access and what commands they are entering.
- Protect sensitive resources with strong authentication.
- Upon termination, delete employee's computer and network access.
- Upon termination, make a mirror image of employee's hard drive before reissuing it. It is useful when your company information turns up at a competitor

…

Summary

Table 9.1 Some Examples of Intruder Patterns of Behavior

(a) Hacker

1. Select the target using IP lookup tools such as NSLookup, Dig, and others.
2. Map network for accessible services using tools such as NMAP.
3. Identify potentially vulnerable services (in this case, pcAnywhere).
4. Brute force (guess) pcAnywhere password.
5. Install remote administration tool called DameWare.
6. Wait for administrator to log on and capture his password.
7. Use that password to access remainder of network.

(b) Criminal Enterprise

1. Act quickly and precisely to make their activities harder to detect.
2. Exploit perimeter through vulnerable ports.
3. Use Trojan horses (hidden software) to leave back doors for reentry.
4. Use sniffers to capture passwords.
5. Do not stick around until noticed.
6. Make few or no mistakes.

(c) Internal Threat

1. Create network accounts for themselves and their friends.
2. Access accounts and applications they wouldn't normally use for their daily jobs.
3. E-mail former and prospective employers.
4. Conduct furtive instant-messaging chats.
5. Visit Web sites that cater to disgruntled employees, such as f'dcompany.com.
6. Perform large downloads and file copying.
7. Access the network during off hours.

Intrusion Techniques

Intrusion Techniques

- The objective of the intruder is to gain access to a system, or increase the range of privileges accessible on a system.
- The intruder attempts to acquire information that should have been protected. In some case, this information is user password.
- A system must maintain a file that associates a password with each authorized user.
- The password can be protected in one of two ways:
  - **One-way function**: the system stores only the value of a function based on the user's password. In practical, the password is used to generate a key for the one-way function and a fixed-length output is produced.
  - **Access control**: access to the password file is limited to one or a very few accounts.

Intrusion Techniques

- [ALVA90] reports some techniques for learning passwords:
  - Try default passwords used with standard system
  - Exhaustively try all short password (one to three characters)
  - Try words in system's online dictionary, or list of likely passwords
  - Collect information about users (names, hobbies, habit …)
  - Try user's phone numbers, Social Security numbers, room numbers
  - Use the trojan horse to bypass restrictions on access

…

Intrusion Detection

Detection is concerned with learning of an attack, either before or after its success.

Intrusion Detection

- 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.

- The area of research on intrusion detection focus on:

- The sooner the intrusion is detected, the less damage and the more quickly recovery can be achieved.
- An effective intrusion detection system acts to prevent intrusions.
- Intrusion detection enables the collection of information about intrusion techniques that can be used to strengthen the intrusion prevention facility.

Intrusion Detection

- [PORR92] identifies some approaches to intrusion detection:

- **Statistical anomaly detection:** Involves the collection of data relating to the behavior of legitimate users over a period of time. Then statistical tests are applied to determine whether that the 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.

Intrusion Detection

- **Rule-based detection:** Involves an attempt to define a set of rules that can be used to decide that a given behavior is that of an intruder.
  - **Anomaly detection:** Rules are developed to detect deviation from previous usage patterns.
  - **Penetration identification:** An expert system approach that searches for suspicious behavior.

Intrusion Detection

- **Statistical approaches** attempt to define normal, or expected, behavior, whereas **rule-based approaches** attempt to define proper behavior.
- Statistical detection is effective against masqueraders, who are unlikely to mimic the behavior patterns of the accounts they appropriate.
- Rule-based approaches is effective against misfeasors, able to recognize events and sequences that, reveal penetration.
- In practice, a system exhibit a combination of both approaches to be effective against a broad range of attacks.

Audit Records

Audit Records

- Some record of ongoing activity by users must be maintained as input to an intrusion detection system.

Native audit records	Detection-specific audit records
Virtually all operating systems include accounting software that collects information on user activity. **The advantage**: no additional collection software is needed. **The disadvantage**: the native audit records may not contain the needed information or may not contain it in a convenient form.	A collection facility can be implemented that generates audit records containing only information required by the intrusion detection system. **The advantage:** it could be made vendor independent and ported to a variety of systems. **The disadvantage:** the extra overhead involved in having, in effect, two accounting packages running on a machine.

Audit Records

- Dorothy Denning [DENN87] developed a good example of detection-specific audit records, covered:

- **Subject:** Initiators of actions. All activity arises through commands issued by subjects.
- **Action:** Operation performed by the subject on or with an object; for example, login, read, perform I/O, execute.
- **Object:** Receptors of actions, include files, programs, messages, records, terminals, printers, …
- **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 (e.g., number of records read or written, processor time, I/O units used, …).
- **Time-Stamp:** Unique time-and-date stamp identifying when the action took place.

Audit Records

- Most user operations are made up of a number of elementary actions.

COPY GAME.EXE TO <Library>GAME.EXE

Smith	execute	<Library>COPY.EXE	0	CPU = 00002	11058721678
Smith	read	<Smith>GAME.EXE	0	RECORDS = 0	11058721679
Smith	execute	<Library>COPY.EXE	write-viol	RECORDS = 0	11058721680

Statistical Anomaly Detection

Statistical Anomaly Detection

- **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.
- Threshold analysis is ineffective detector of sophisticated attacks. However, simple threshold detectors may be useful in conjunction with more sophisticated techniques.

- **Profile-based anomaly detection:** focuses on characterizing the past behavior of individual users or related groups of users and then detecting significant deviations.

Statistical Anomaly Detection

- Some metrics that are useful for **profile-based intrusion detection**:

- **Counter:** A nonnegative integer that may be incremented but not decremented. A count of certain event types is kept over a particular period of time (number of logins by a single user during an hour, number of password failures during a minute…)
- **Gauge:** A nonnegative integer that may be incremented or decremented. A gauge is used to measure the current value of some entity (number of logical connections assigned to a user application and the number of outgoing messages queued for a user process.)
- **Interval timer:** The length of time between two related events (the length of time between successive logins to an account.)
- **Resource utilization:** Quantity of resources consumed during a specified period (number of pages printed during a user session and total time consumed by a program execution.)

Rule-Based Intrusion Detection

Rule-Based Intrusion Detection

- Rule-based 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:** historical audit records are analyzed to identify usage patterns and to generate automatically rules that describe those patterns. Rules represent past behavior patterns of users, 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.
- **Rule-based penetration identification:** The key feature of such systems is 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.

The Base-Rate Fallacy

The Base-Rate Fallacy

- An intrusion detection system should detect a substantial percentage of intrusions while keeping the false alarm rate at an acceptable level.

- It is very difficult to meet the standard of high rate of detections with a low rate of false alarms. In general, 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.

Distributed Intrusion Detection

Distributed Intrusion Detection

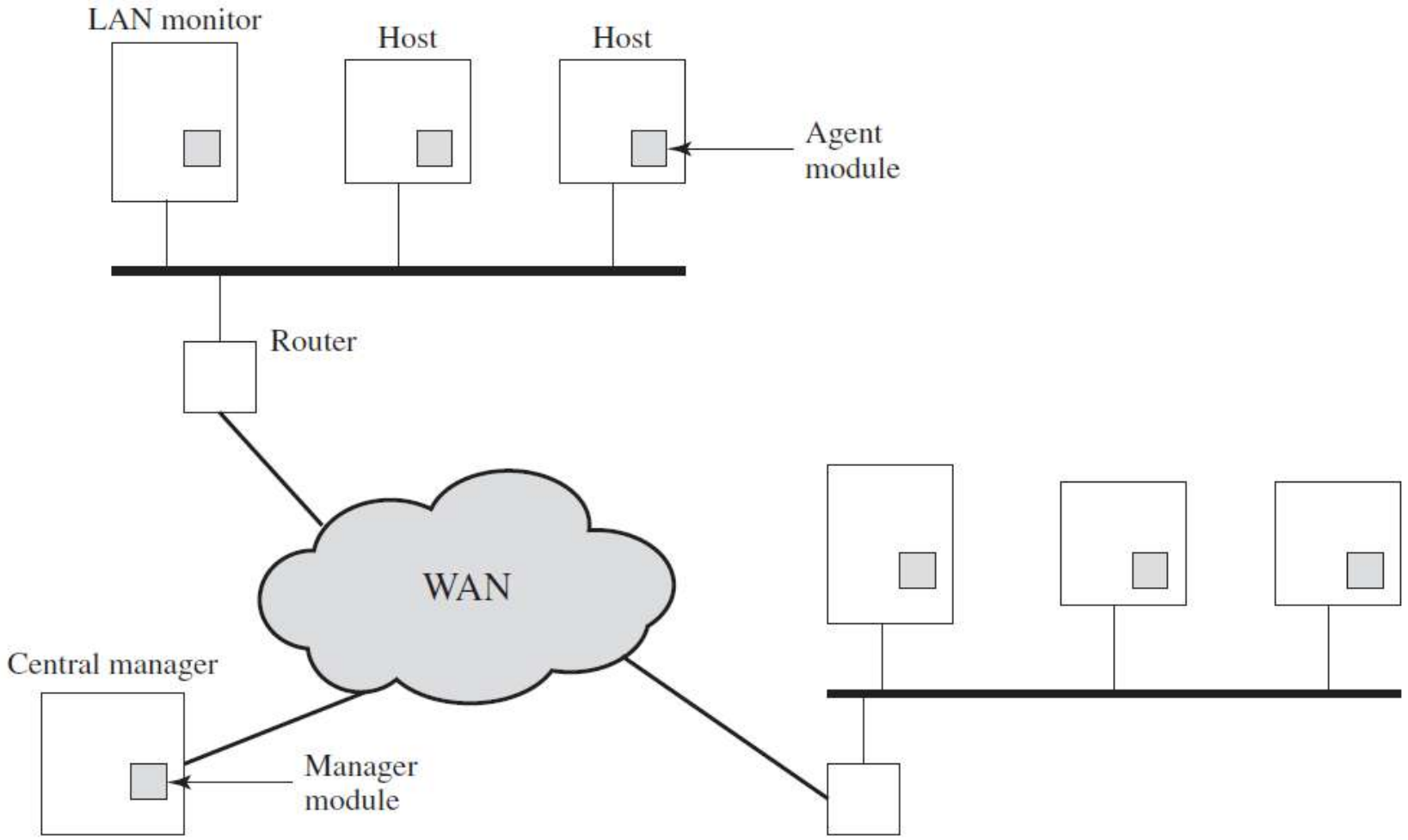


Figure 9.2 Architecture for Distributed Intrusion Detection

Distributed Intrusion Detection

- A good example of a distributed intrusion detection system is one developed at the University of California at Davis [HEBE92, SNAP91].

- **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.
- **LAN monitor agent module:** Operates in the same fashion as a host agent module except that it analyzes LAN traffic and reports the results to the central manager.
- **Central manager module:** Receives reports from LAN monitor and host agents and processes and correlates these reports to detect intrusion.

Distributed Intrusion Detection

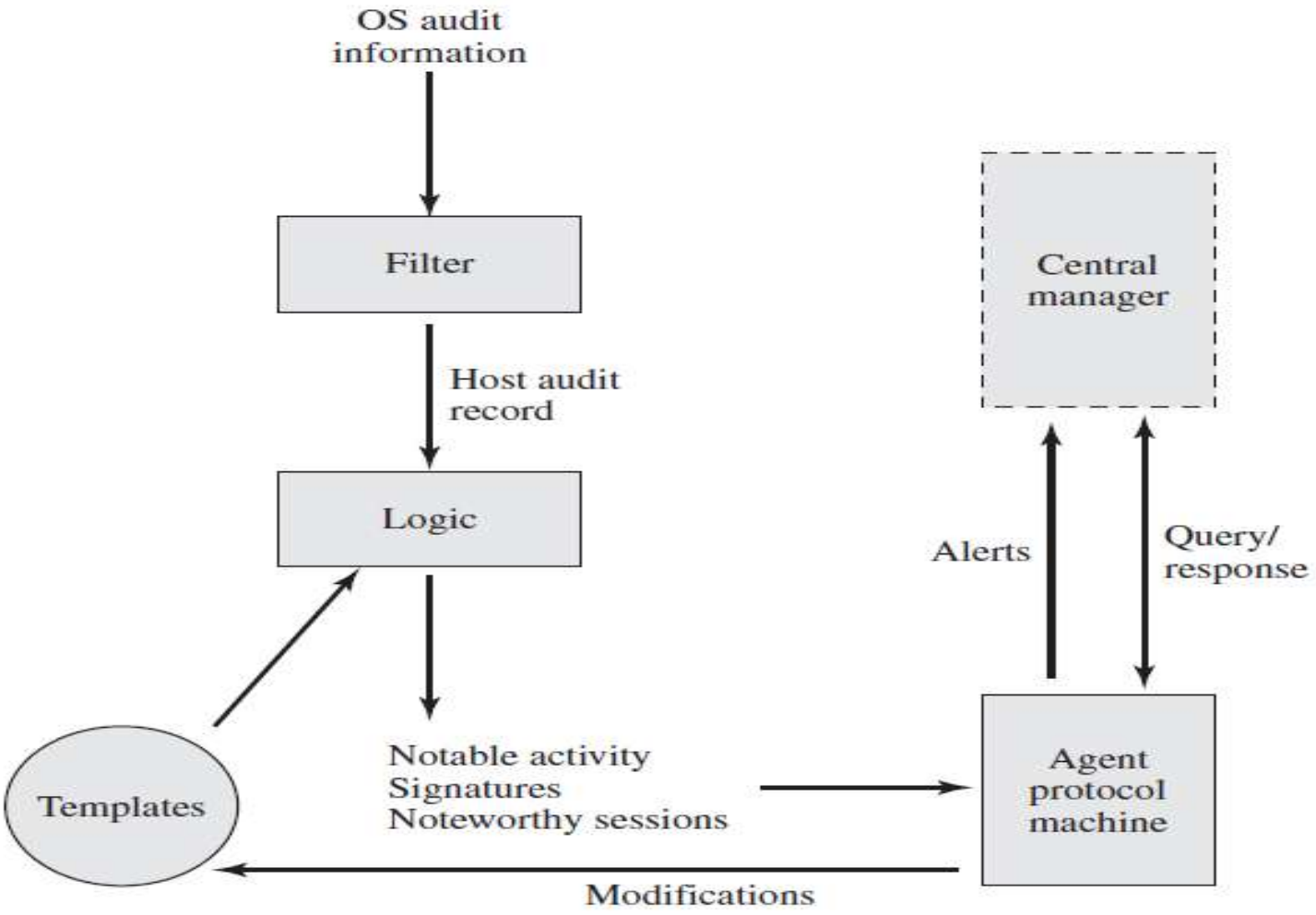


Figure 9.3 Agent Architecture

Distributed Intrusion Detection

- The agent captures each audit record produced by the native audit collection system.
- A filter is applied that retains only those records that are of security interest.
- These records are then reformatted into a standardized format referred to as the host audit record (HAR).
- Next, a template-driven logic module analyzes the records for suspicious activity. At the lowest level, the agent scans for notable events that are of interest independent of any past events. Examples include failed file accesses, accessing system files, and changing a file's access control.
- At the next higher level, the agent looks for sequences of events, such as known attack patterns (signatures).
- Finally, the agent looks for anomalous behavior 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.

Distributed Intrusion Detection

- When suspicious activity is detected, an alert is sent to the central manager.
- The central manager includes an expert system that can draw inferences from received data. The manager may also query individual systems for copies of HARs to correlate with those from other agents.
- The LAN monitor agent also supplies information to the central manager.The LAN monitor agent audits host-host connections, services used, and volume of traffic.
- It searches for significant events, such as sudden changes in network load, the use of security-related services, and network activities such as *rlogin*.

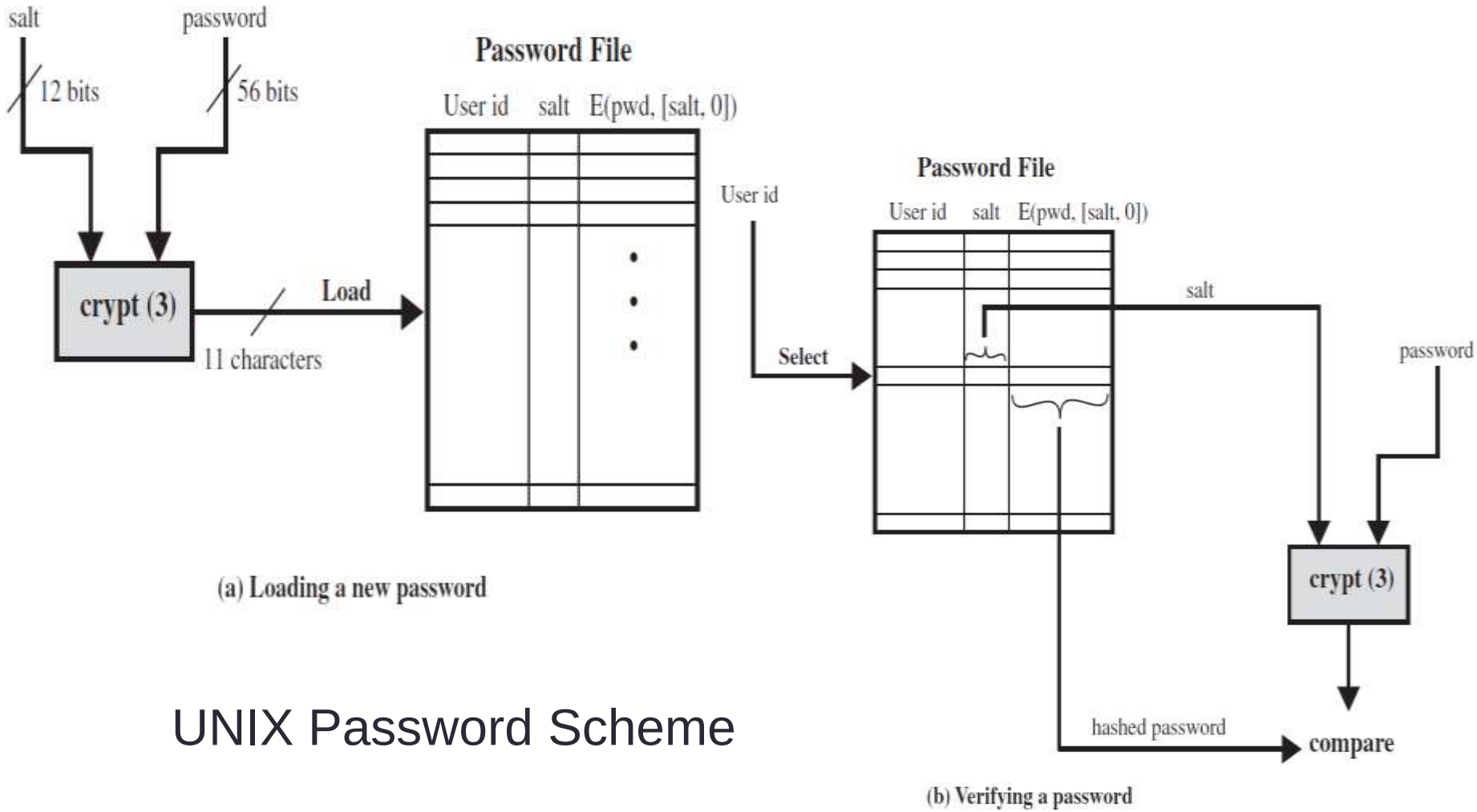
Honeypots

Honeypots

- Honeypots are decoy systems that are designed to lure a potential attacker away from critical systems.
  - 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.
- Because any attack against the honeypot is made to seem successful, administrators have time to mobilize and log and track the attacker without ever exposing productive systems.

Password Management

Password Protection



UNIX Password Scheme

Password Protection

- Each user selects a password of up to eight printable characters in length. This is converted into a 56-bit value (using 7-bit ASCII) that serves as the key input to an encryption routine.
- The encryption routine, known as crypt(3), is based on DES. The DES algorithm is modified using a 12-bit "**salt**" value. This value is related to the time at which the password is assigned to the user. The modified DES algorithm is exercised with a data input consisting of a 64-bit block of zeros.
- The output of the algorithm then serves as input for a second encryption. This process is repeated for a total of 25 encryptions.
- The resulting 64-bit output is then translated into an 11-character sequence.
- The hashed password is then stored, together with a plaintext copy of the salt, in the password file for the corresponding user ID.
- This method has been shown to be secure against a variety of cryptanalytic attacks [WAGN00].

Password Protection

- The **salt** serves three purposes:
  - It prevents duplicate passwords from being visible in the password file.
  - It effectively increases the length of the password, the number of possible passwords is increased by a factor of 4096, hence increases the difficulty of guessing a password.
  - It prevents the use of a hardware implementation of DES, which would ease the difficulty of a brute-force guessing attack.

Password Protection

- Password cracker was reported on the Internet in August
- 1993 [MADS93] using a Thinking Machines Corporation parallel computer, a performance of 1560 encryptions per second per vector unit was achieved.

- With four vector units per processing node, this works out to 800,000 encryptions per second on a 128-node machine and 6.4 million encryptions per second on a 1024-node machine.

Password Protection

- Instead of using a dumb brute-force technique of trying all possible combinations of characters to discover a password, password crackers rely on the fact that some people choose easily guessable passwords.
- Some users, when permitted to choose their own password, pick one that is short. An attacker could begin the attack by exhaustively testing all possible passwords of length 3 or fewer.
- On the other hands, many people pick a password that is guessable, such as their name, their street name, a common dictionary word, and so forth. The cracker simply has to test the password file against lists of likely passwords.

Password Protection

Table 9.4 Observed Password Lengths [SPAF92a]

Length	Number	Fraction of Total
1	55	.004
2	87	.006
3	212	.02
4	449	.03
5	1260	.09
6	3035	.22
7	2917	.21
8	5772	.42
Total	13787	1.0

Password Protection

Table 9.5 Passwords Cracked from a Sample Set of 13,797 Accounts [KLEI90]

Type of Password	Search Size	Number of Matches	Percentage of Passwords Matched	Cost/Benefit Ratio[a]
User/account name	130	368	2.7%	2.830
Character sequences	866	22	0.2%	0.025
Numbers	427	9	0.1%	0.021
Chinese	392	56	0.4%	0.143
Place names	628	82	0.6%	0.131
Common names	2239	548	4.0%	0.245
Female names	4280	161	1.2%	0.038
Male names	2866	140	1.0%	0.049
Uncommon names	4955	130	0.9%	0.026
Myths & legends	1246	66	0.5%	0.053
Shakespearean	473	11	0.1%	0.023
Sports terms	238	32	0.2%	0.134
Science fiction	691	59	0.4%	0.085
Movies and actors	99	12	0.1%	0.121
Cartoons	92	9	0.1%	0.098
Famous people	290	55	0.4%	0.190

Password Protection

Phrases and patterns	933	253	1.8%	0.271
Surnames	33	9	0.1%	0.273
Biology	58	1	0.0%	0.017
System dictionary	19683	1027	7.4%	0.052
Machine names	9018	132	1.0%	0.015
Mnemonics	14	2	0.0%	0.143
King James bible	7525	83	0.6%	0.011
Miscellaneous words	3212	54	0.4%	0.017
Yiddish words	56	0	0.0%	0.000
Asteroids	2407	19	0.1%	0.007
TOTAL	62727	3340	24.2%	0.053

Access Control

- One way to thwart a password attack is to deny the opponent access to the password file.

- If the encrypted password portion of the file is accessible only by a privileged user, then the opponent cannot read it without already knowing the password of a privileged user.

Password Selection Strategies

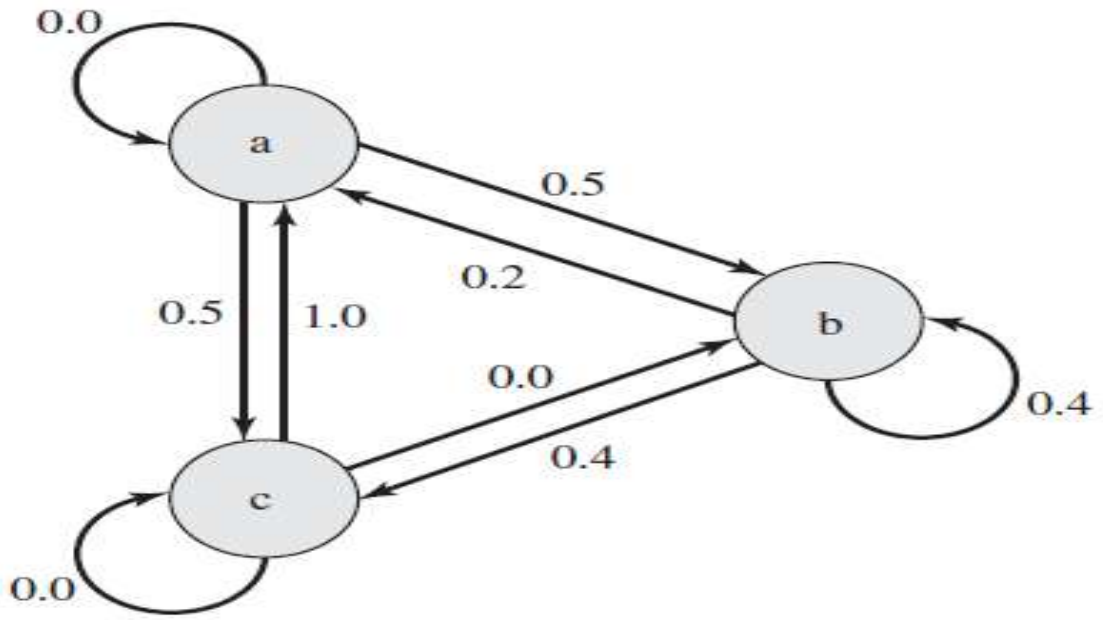
- If users are assigned passwords consisting of eight randomly selected printable characters, password cracking is effectively impossible.
- But it would be almost as impossible for most users to remember their passwords.
- Four basic techniques help to eliminate guessable passwords while allowing the user to select a password that is memorable:
  - User education: Users can be told the importance of using hard-to-guess passwords and can be provided with guidelines for selecting strong passwords.
  - Computer-generated passwords: FIPS PUB 181 defines a C source code of the algorithm. The algorithm generates words by forming pronounceable syllables and concatenating them to form a word. A random number generator produces a random stream of characters used to construct the syllables and words.

Password Selection Strategies

- reactive password checking: strategy is one in which the system periodically runs its own password cracker to find guessable passwords. The system cancels any passwords that are guessed and notifies the user.
- proactive password checker: a user is allowed to select his or her own password. However, at the time of selection, the system checks to see if the password is allowable and, if not, rejects it.
  - Some rules should be enforced:
    - All passwords must be at least eight characters long.
    - In the first eight characters, the passwords must include at least one each of uppercase, lowercase, numeric digits, and punctuation marks.

    …

Password Selection Strategies



M = {3, {a, b, c} , T, 1 } where

$$T = \begin{bmatrix} 0.0 & 0.5 & 0.5 \\ 0.2 & 0.4 & 0.4 \\ 1.0 & 0.0 & 0.0 \end{bmatrix}$$

e.g., string probably from this language: abbcacaba

e.g., string probably not from this language: aacccbaaa

Figure 9.5 An Example Markov Model

Password Selection Strategies

- Spafford [SPAF92a, SPAF92b] used of a Bloom filter concept in another way to develop an effective and efficient proactive password checker.
- A Bloom filter of order k consists of a set of k independent hash functions $H_1(x)$, $H_2(x)$, … , $H_k(x)$ where each function maps a password into a hash value in the range 0 to N - 1.

$$H_i(X_j) = y \qquad 1 \le i \le k; \qquad 1 \le j \le D; \qquad 0 \le y \le N - 1$$

where

X_j = jth word in password dictionary

D = number of words in password dictionary

Password Selection Strategies

- The following procedure is then applied to the dictionary:
  - A hash table of bits is defined, with all bits initially set to 0.
  - For each password, its hash values are calculated, and the corresponding bits in the hash table are set to 1.Thus, if $H_i(X_j) = 67$ for some (i, j), then the sixty-seventh bit of the hash table is set to 1; if the bit already has the value 1, it remains at 1.
- When a new password is presented to the checker, its hash values are calculated.
- If all the corresponding bits of the hash table are equal to 1, then the password is rejected. All passwords in the dictionary will be rejected.

Password Selection Strategies

$H_1(\text{undertaker}) = 25$ $H_1(\text{hulkhogan}) = 83$ $H_1(\text{xG\%\#jj98}) = 665$

$H_2(\text{undertaker}) = 998$ $H_2(\text{hulkhogan}) = 665$ $H_2(\text{xG\%\#jj98}) = 998$

If the password xG%#jj98 is presented to the system, it will be rejected even though it is not in the dictionary.

The hash scheme to minimize false positives. The probability of a false positive can be approximated by:

$$P \approx \left(1 - e^{kD/N}\right)^k = \left(1 - e^{k/R}\right)^k$$

Password Selection Strategies

$$R \approx \frac{-k}{\ln(1 - P^{1/k})}$$

where

k = number of hash functions

N = number of bits in hash table

D = number of words in dictionary

$R = N/D$, ratio of hash table size (bits) to dictionary size (words)

Password Selection Strategies

- Suppose we have a dictionary of 1 million words and we wish to have a 0.01 probability of rejecting a password not in the dictionary. If we choose six hash functions, the required ratio is R = 9.6.
- Therefore, we need a hash table of 9.6×10^6 bits or about 1.2 MBytes of storage. In contrast, storage of the entire dictionary would require on the order of 8 MBytes. Thus, we achieve a compression of almost a factor of 7.
- Password checking involves the straightforward calculation of six hash functions and is independent of the size of the dictionary.

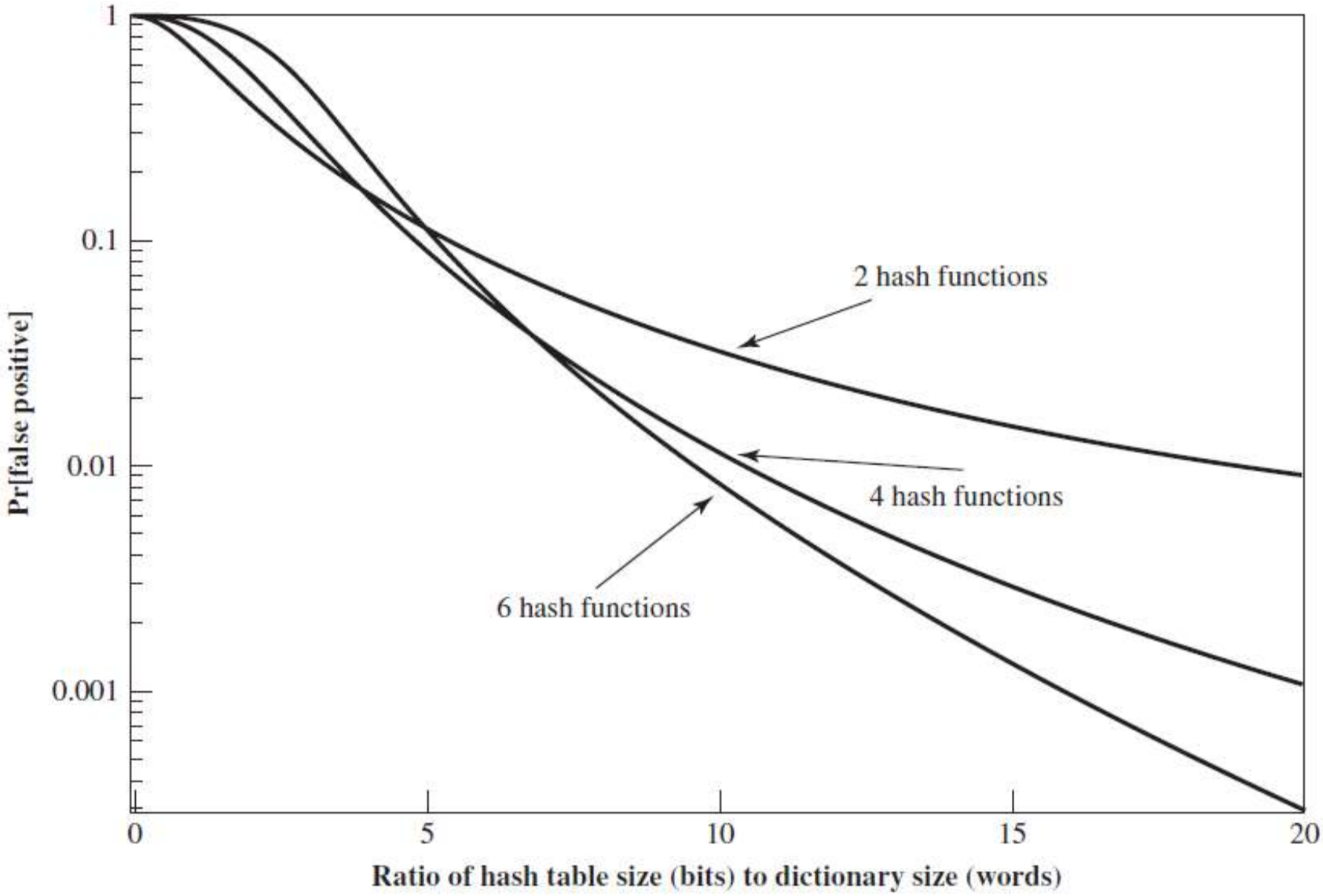


Figure 9.6 Performance of Bloom Filter

References

- William Stallings, “Network Security Essential – 4th Edition”
- http://en.wikipedia.org/wiki/IOS_jailbreaking
- http://en.wikipedia.org/wiki/Edward_Snowden
- http://www.nydailynews.com/new-york/cyber-thieves-busted-45-million-heist-article-1.1339051
- http://www.discovery.com/tv-shows/curiosity/topics/10-famous-hackers-hacks.htm

Q & A

THANKS FOR WATCHING