1. INTRUDERS:-

* An intruder is a person who attempts to gain unauthorized access to a system, to damage that system, or to disturb data on the system.
* Intruders attempts to violate security by interfering with system availability, data integrity, data confidentiality.
* Is one of the most publicized threats to security (other is viruses).
* Anderson identified three classes of intruders:-

(1) Masquerader
(2) Misfeasor
(3) Clandestine User.

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.

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.

Clandestine User:-

* An individual who seizes supervisory control of a system and uses this control to evade auditing and access controlls or to suppress audit collection.

* 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.

Examples of intrution:- Consists of,

- Performing a remote root compromise on 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:

The techniques and behaviour patterns of intruders are constantly shifting, to exploit newly discovered weakness and to evade detection and countermeasures.

1. Hackers
2. Criminals
3. Insider Attacks.

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 other 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 VC walk into his office as see the files on his windows workstation.
* Benign intruders might be tolerable, they just consume resources and may performance for legitimate users.
* Serious intruders may lead to big damage, especially in official or goverment systems

IDS (Intrution Detection Systems) and IPS (Intrution Prevention Systems) are designed to counter this type of hacker threat.

* Organization can consider restricting remote logons to specific Ip addresses and use virtual private network technology.
* Computer Emergency Response Teams have established with the cooperative ventures collect information about systems vulnarabilities and disseminate it to system managers.
* The systems administrators will quickly insert all software patches to discover and fix those vulnarabilities.

Ex:- Jailbreak on Iphone, Ipod, Ipad using standard ios help find out vulnarabilities and fix them

CRIMINALS:

* Organized group 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 attempt to gain root access.
* The card numbers are used to purchase expensive items, and the posted in carder sites, where others can access and continue use it.
* IDS's 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.
* For e-commerce sites, database encryption should be used for sensitive customer information, especially credit cards.
* They use dedicated server (not support multiple customers) and closely monitor the provider's security services.

INSIDER ATTACKS

* 2013, Edward Snowden, A computer specialist, former employee of CIA and NSA, disclosed thousand of classified documents to the media. This have weakened national security.
* IDSs and IPSs can be usefull to counter this attack, combine with some approaches.

* 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.

INTRUSION TECHNIQES:

* The objective of the intruder is to login 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 Cases, 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.
  (1) 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.
  (2) Access Control :- Access to the password file is limited to one or a very few accounts.

Some techniques for learning passwords:-

- Try default passwords used with standard system
- Exhaustively try all short password (1-3 char).
- Try words in system's online dictionary or list of likely passwords.
- Collect information about users (names, hobbies..)
- Try User's phone numbers, Social security numbers, room numbers.
- Use the Trojan horse to bypass restrictions on access.

2. INTRUSION DETECTION:

* Detection is concerned with learning of an attack, either before or after its success.
* Intrusion detection is based on the assumption that the behaviour of the intruder differs from that of legitimate user in ways that can be qualified.

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 institution / intrusion techniques that can be used to strengthen the intrusion prevention facility.

Some approaches to intrusion Detection:-

Statistical anomaly detection:

* Involves the collection of data relating to the behaviour of legitimate users over a period of time. The statistical tests are applied to determine whether that the behaviour is not legitimate user behaviour.

(*) Threshold detection:- This approach involves defining thresholds, independent of user, for the frequency of occurence of various events.

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

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.

  - Anamoly detection :- Rules are developed and detect deviation from previous usage patterns.
  - Penetration identification :- An expert system approach that searches for suspicious behaviour.

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

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

Detection specific audit records, covered:

Subject : Initiators of actions. All activity arises through commands issued by subjects.

Action : Operation performed by the subject on / with an object, for ex, login, read, performIo, execute.

Object : Receptors of actions, includes files, programs, messages, records, terminals, printers, . . .

Exception Condn : 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

TimeStamp : Unique time-and date stamp identifying when the action took place

Statistical anomaly Detection:

Threshold detection:

* Involves counting the number of occurances of a specific event type over an interval of time
* If the count surpass 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 conjuction with more sophisticated techniques.

Profile-based anomaly Detection:

* Focuses on charecterizing the past behaviour of individual users or related groups of users and then detecting significant deviations.

Some metrics are useful for profile-based intrusion Detection.

Counter:

* A non negative integer that may be incremented but not decremented. A count of certain event types is kept over a particular period of time.

Gauge:

* A non negative integer that may be incremented / decremented. A gauge is used to measure the current value of some entity.

Interval Timer:

* The length of time between two related events

Resource Utilization:

* Quantity of resources consumed during a specified period.

Rule-Based Intrusion Detection:

* Rule-based techniques detects 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.

Ruk-based anomaly detection:

* Historical audit records are analyzed to identify usage patterns and to generate automatically rules that describe those patterns and to generate automatically rules that describe those patterns.
* Rules represent past behaviour patterns of users, current behavior is then observed, and each transaction is matched against the set of rules to determine if it confirms to any historically observed pattern of behaviour.

Rule-based penetration detection:

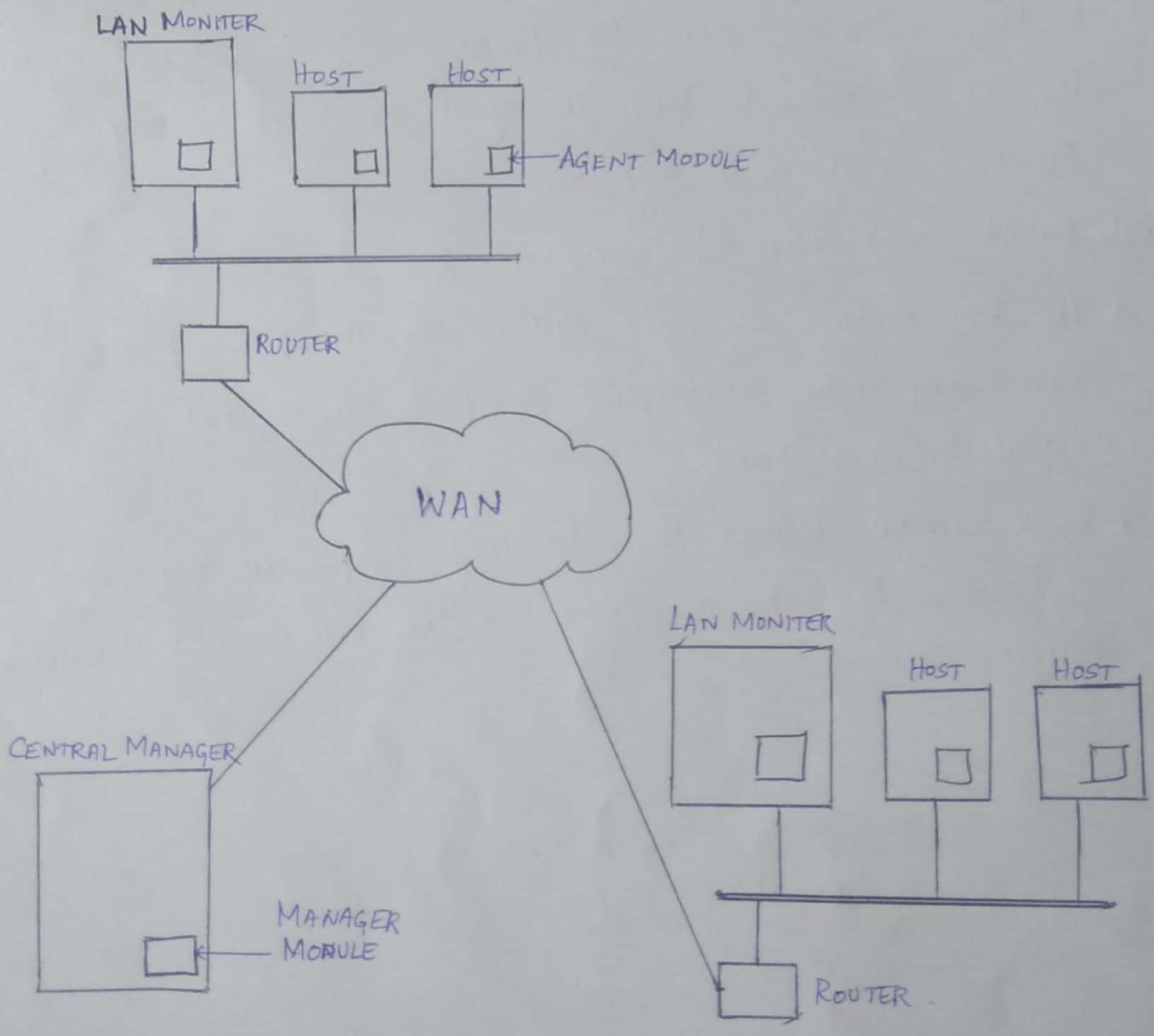
* The key feature of such systems is the use of rules for identifying known penetration or penetrations that would exploit known weakness.
* Rules can be defined that identify suspicious behaviour, even when the behaviour is within the bounds of established patterns of usage.

The Base-Rate Fallacy:

* An intrusion detection system should detect a substantial percentage of intrusions while keeping the false alarm rate at acceptable level.
* It is very difficult to meet the standard of high rate of detections with a low rate of false alarm. In general if the actual numbers of intrusion is low compared to the number of legitimate uses of a system, then the false alarm rate will be high.

Distributed Intrusion Detection:

Architecture:-



A good example of a distributed intrusion detection system is one developed at 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.

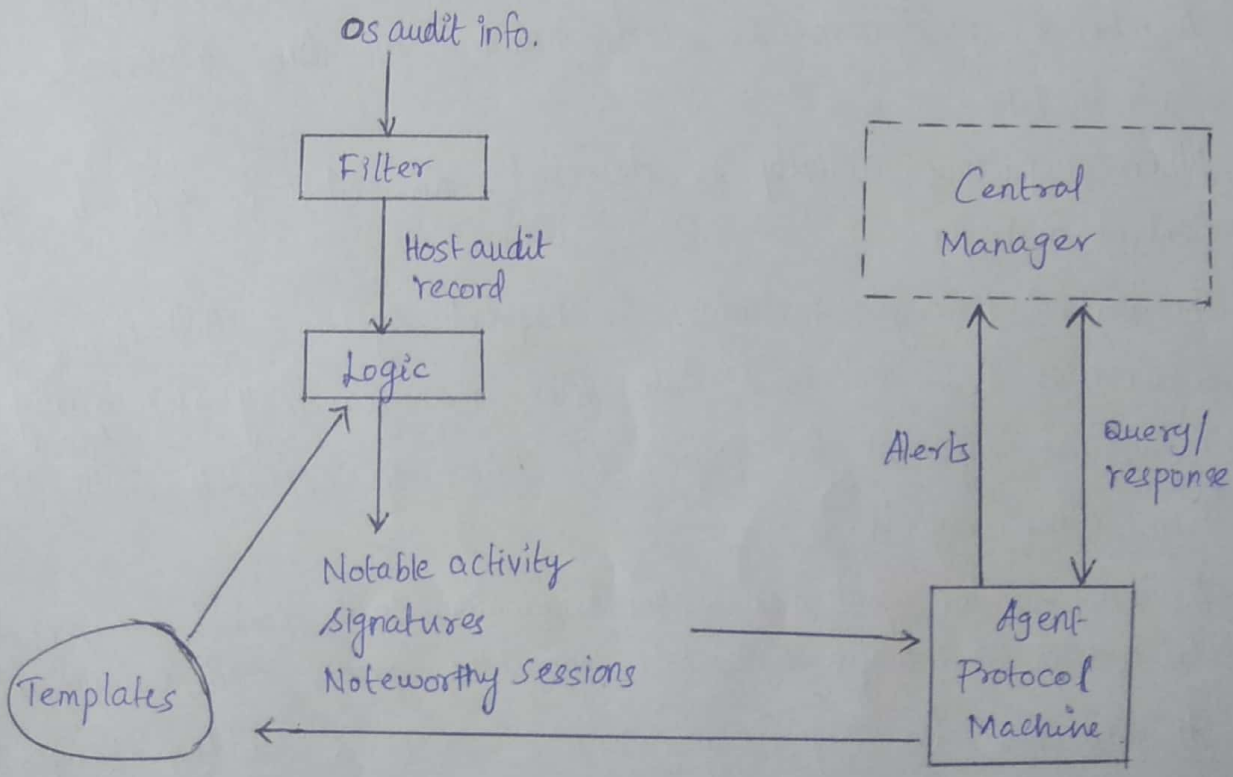
LAN Monitor agent module:

* operates in the same fashion as a host agent module except that it analyses 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.

Agent Architecture:-



* The agent captures each audit record produced by the native audit collection system.
* A filter is applied that retains only those records that are ~~not~~ of security interest.
* These records are then reformated into a standardized format referred to as the host audit record (HAR).
* Next a template-driven logic module analyses 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.
* 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.
* 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 moniter agent also supplies information to the central manager. The LAN moniter agent audits host-host connections, services used, volume of traffic
* It searches for significant events, such as sudden changes in network load, use of security related services, network activity such as rlogin.

Honeypots:-

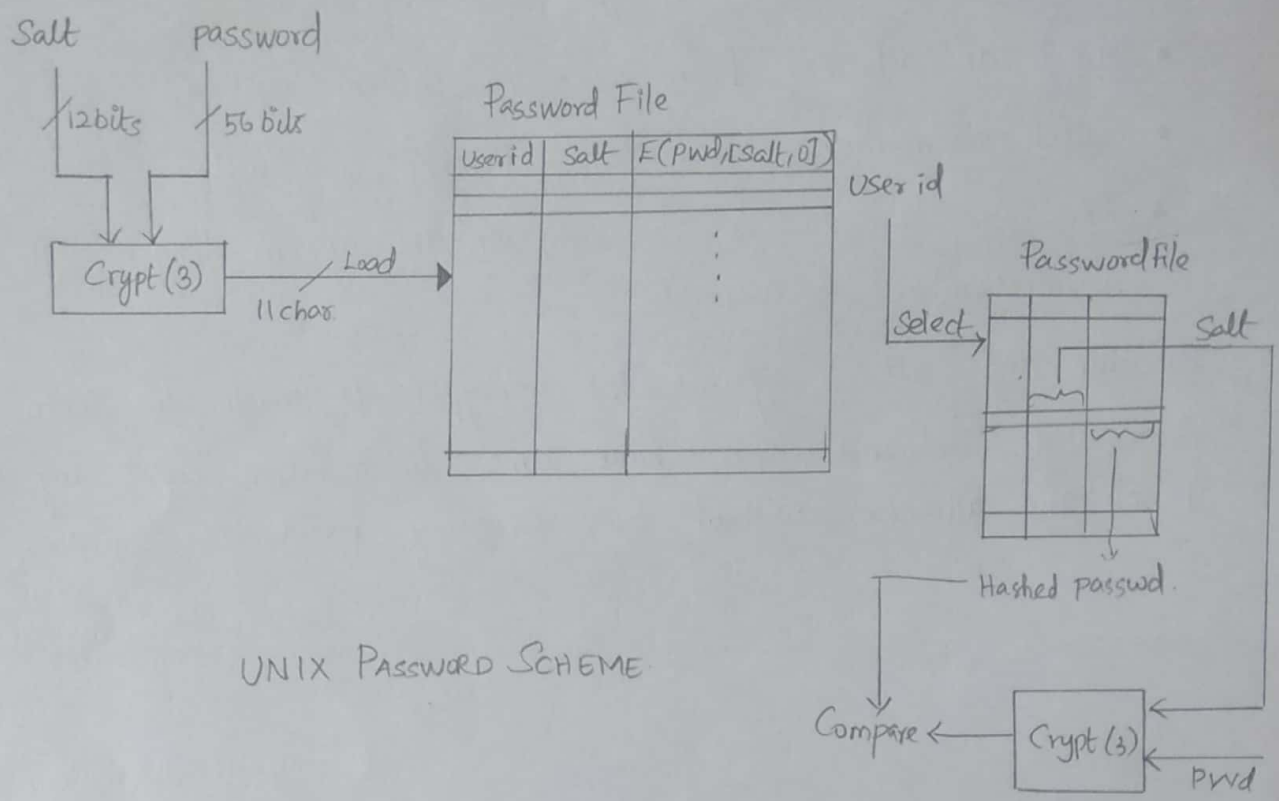
Honeypots are decoy system that are designed to lure a potential attacker away from critical system.

- Divert an attacker from accessing critical system
- 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.

3. PASSWORD MANAGEMENT:

Password Protection:-



UNIX PASSWORD SCHEME

- Each user selects a password of up to eight printable characters in length. This is converted into a 56 bit value 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 64bit 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 pwd is then stored, together with a plaintext copy of the salt, in the pwd file for the corresponding user ID.

- This method has been shown to be secure against a variety of cryptanalytic attacks.

The Salt serves 3 purposes:-

(1) It prevents duplicate passwords from being visible in the password file.

(2) It effectively increases the length of the password, the number of possible password is increased by a factor of 4096, hence increases the difficulty of guessing a password.

(3) It prevents the use of a hardware implementation of DES, which would ease the difficulty of a brute-force guessing attack.

- password Cracker was reported on the internet in August 1993 using a Thinking Machine Corporation parallel computer, a performance of 1560 encryptions per second per vector unit was acheived.
- With four vector unit per processing node, this works out to 800,000 encryptions per second on a 128-node machine and 6.4 million encryption per second on a 1024 node machine.
- Instead of using a dumb brute-force techniques 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 pwds of length 3 or fewer.
- Many people pick a pwd is guessable such as their name, there street name, a common dictionary word and so forth. The hacker/cracker simply has to test the password file against lists of likely passwords.

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.

(1) User Education:-

- User can be told the importance of using hard-to-guess passwords and can be provided with guidelines for selecting strong passwords.

(2) Computer-generated passwords:-

- FIPS PUB 181 defines a C source code of the algorithm. The algorithm generates words by forming pronounceables syllables and concatenating them to form a word. A random number generator produces a random stream of characters used to construct syllables and words.

(3) 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.

(4) Proactive password checker:-

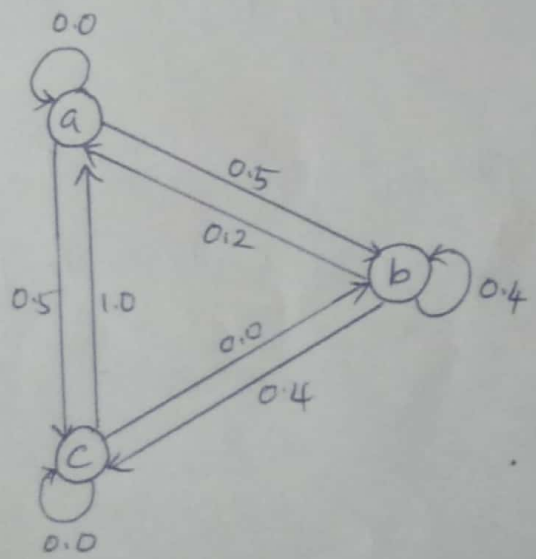
- A user is allowed to select his or her own password. However at the time of selection, the system checks to see if the password is allowable and if not, rejects it.
- Some rules should be enforced,
  - All passwords must be at least eight characters long.
  - In the first eight characters, the passwords must include at least one each of uppercase, lowercase, numeric digits, punctuation marks.

Two techniques for developing an effective and efficient proactive password checker are,

(1) Markov model

(2) Bloom filter (Spafford).

Markov model:



$M = \{3, \{a, b, c\}, T, \}$ where

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

- Markov model is a quadruple [m, A, T, k]

  m - Number of states

  A - State Space

  T - Matrix of transition probabilities

  k - Order of the model.

- The Transition matrix is calculated as,

$$T(i,j,k) = \frac{f(i,j,k)}{f(i,j,\infty)}$$

- First determine frequency matrix f, where f(i,j,k) is number of occurances of trigram.
- Then for each bigram ij, Calculate f(i,j,∞). is the total number of trigrams of the Message.

Bloom filter (Spafford):

- Spafford used a bloom filter Concept in another way to develop an effective and efficient proactive password checker.
- 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.

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

Where,

X_j - j^{th} Word in password dictionary.

D - Number of words in the password dictionary.

The following procedure is 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 corrosponding bits in the hash table are set to 1.
  - If $H_i(X_j) = 67$ for some $(i, j) \rightarrow$ 67 th bit of 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 corrosponding bits of the hash table are equal to 1 then the password is rejected. All passwords in the dictionary will be rejected.

* H_1(undertaker) = 25 H_1(hulkhogan) = 83 H_1(xG%#jj98) = 665
  H_2(undertaker) = 998 H_2(hulkhogan) = 665 H_2(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,

$$P \approx (1 - e^{KD/N})^k = (1 - e^{k/R})^k.$$

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

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)

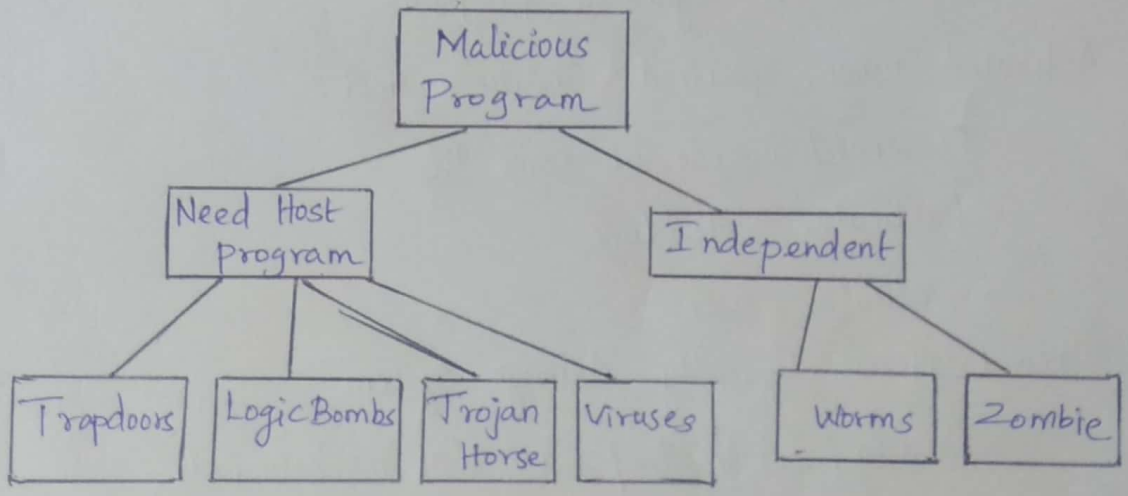
- Suppose We have a dictionary of 1 million words and we wish to have a 0.01 probability of rejecting a password not in the dictionary. If we choose six hash functions, the required ratio is $R = 9.6$.
- Therefore We need a hash table of 9.6×10^6 bits or about 1.2 MBytes to Storage. In constrast storage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

4. VIRUSES AND RELATED THREATS:

* Most sophisticated types of threats to computer systems are presented by programs that exploit vulnerabilities in computing systems.

Malicious Programs:-

* Malicious programs / Software is a Software that is intentionally included or inserted in a system for a harmful purpose.
* Malicious Software can be divided into two categories.
  (1) Software that need host program
  (2) Software that are independent.



(1) Need a host program referred to that cannot exist independently of some actual application program, utility, or system program. (Viruses, logic bombs, backdoors)

(2) Independent, a self-contained program that can be scheduled and run by the operating system. (Worms, boot programs)

Backdoor or Trapdoor :-

- Secret entry point into a program
- Allows those who know access bypassing usual security procedures.
- Have been commonly used by developers
- A threat when left in production programs allowing exploited by attackers.
- Very hard to block in Operating System.
- Require good software development and update.

Logic Bomb :-

- One of oldest types of malicious Software
- Code embedded in legitimate program
- Activated when specified conditions met
  - Presence / absense of some file
  - Particular date / time
  - Particular user.
- When triggered typically damage system
  - Modify / delete files / disks, halt machine, etc.

Trojan Horse :-

- Program with hidden side-effects which is usually apparently attractive. eg:- Game, software upgrade
- When run performs some additional tasks
  - Allows attacker to indirectly gains access they do no have directly
- Used to propagate a virus or worm or install a backdoor or simply to destroy data

Zombie:-

- Program which Secretly takes over another networked computer.
- Then uses it to indirectly launch attacks
- Often used to launch distributed denial of services (DoS) attacks.
- Exploits known flaws in network system.

Viruse:-

- A viruse is a piece of software that can infect other programs by modifying them, the modification includes a copy of the virus program which go on to infect other programs.
- A virus can do anything that other program do. The only difference is that is attaches itself to another program an executes secretly when the host program is run.
- Once the virus is executing, it can perform any function such as erasing files and programs.
- During its lifetime, A typical virus goes through the following four phases.

1. Dormant phase
2. Propagation phase
3. Triggering phase
4. Execution phase.

1. Dormant phase:-

* The virus is idle. The virus will eventually be activated by some event, such as date, the presence of another program or file, or the capacity of disk exeeding some limit. Not all viruses have this stage.

2. Propagation phase:-

* The virus places an identical copy of itself into other programs or into certain system areas on the disk.
* Each infected program will now contain a clone of the virus, which will itself enter a propagation phase.

3. Triggering phase:-

* The virus is activated to perform the function for which it was intended. As with dormant phase, the triggering phase can be caused by a variety of system events, including a count of the number of times that this copy of the virus has made copies of itself.

4. Execution phase:-

* The function is performed. The function may be harmless, such as message on the screen, or damaging such as the destruction of programs and data files.

Viruses are specific to operating system and hardware. They are taking the advantage of the details and weakness of particular systems.

Virus Structure:

* Virus can be prepended or postpended to an executable program or it can be embedded in some other fashion.
* The key to its operation is that the infected program when invoked, will first execute the virus code and then execute the original code of the program.
* An infected program begins with the virus code and works as follows.

- The first line of code is jump to main virus program.
- The second line is special marker that is used by the virus to determine whether or not potential victim program has already been infected with this virus.
- When the program is invoked control is immediately transferred to the main virus program.
- The virus program 1st seeks out uninfected executable files and infects, them.
- Next the virus may perform some action usually detrimental to the system.
- This action could be performed every time the program is invoked, or it could be a logic bomb that triggers only under certain conditions.
- Finally, the virus transfers control to the original program. If the infection phase of the program is reasonably rapid, a user is unlikely to notice any difference between the execution of an infected and uninfected program.

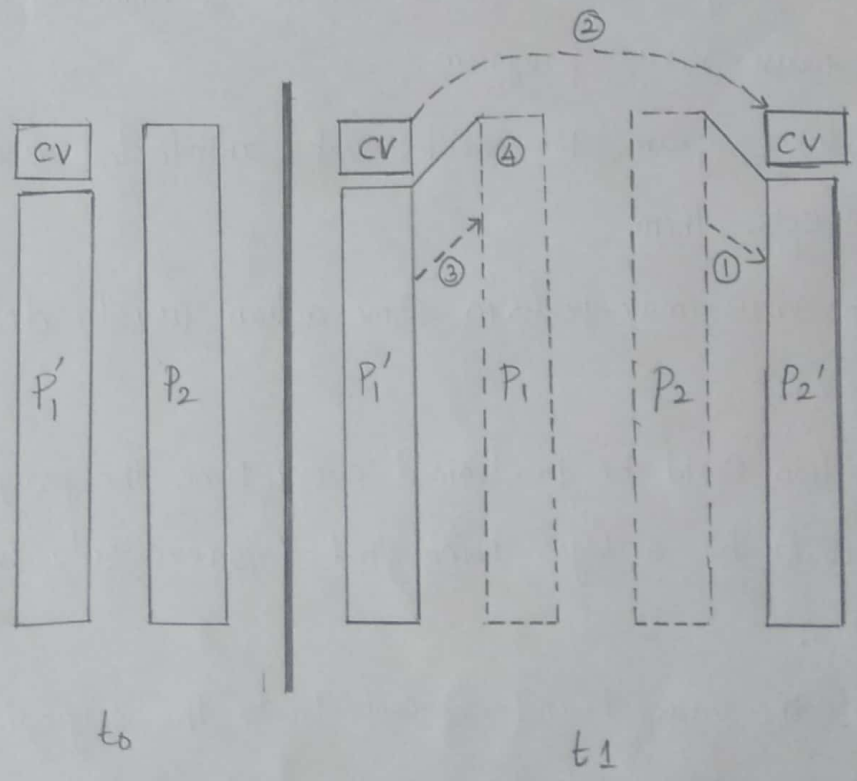
* We assume that program P_1 is infected with the virus Compression Virus (CV). When this program is invoked, Control passes to its virus, which performs the following steps,

(1) For each infected file P_2 that is found, the virus first compresses that file to produce P_2', which is shorter than the original program by the size of the Virus.

(2) A copy of the virus is prepended to the Compressed program.

(3) The compressed version of the original infected program P_1', is uncompressed.

(4) The uncompressed original program is executed.



Types of Viruses :-

Following categories are the most significant types of viruses,

Parasitic Virus:

- The traditional and still most common form of virus. A parasitic virus attaches itself to executable files and replicates, when the infected program is executed, by finding other executable files to infect.

Memory-resident Virus:

- Lodges in main memory as part of resident system program. From that point on, the virus infects every progra that executes.

Boot Sector Virus:

- Infects a master boot record or boot record and spreads when a system is booted from the disk containing the virus.

Stealth Virus :

- A form of virus explicitly designed to hide itself from detection by antivirus software.

Polymorphic Virus:

- A virus that mutates with every infection, making detection by the signature of the virus impossible.

Metamorphic Virus:

- Metamorphic virus mutates with every infection. The difference is that a metamorphic virus rewrites itself completely at each iteration, increasing the difficulty of detection. May change their behaviour and appearance.

Macro Virus:

* In the mid 1990's macro viruses became by far the most prevalent types of virus.
* Macro viruses are particularly threatening for a number of reasons,
  - A macro virus is platform independent. Virtually all of the macro viruses infect MS Word documents. Any hardware platform and OS that supports Word can be infected.
  - Macro virus infected documents not executable portion of code. most of the information introduced onto a computer system is in the form of a document rather than a program.
  - Macro viruses are spread easily. Very common method is by electronic mail (Email).
* Infects files with some macro code that is interpreted by an application and attached to some data file.

  Eg:- Word / Excel Macros

  Using auto command & Command macros
* A major Source of new viral infections.
* Blurs distinction between data and program files making task of detection much harder.
* Macro viruses are recognized by many anti-virus programs.
* Classic trade-off : "Ease of use" Vs "Security".

E-mail Viruses:-

* A more recent development in malicious software is the e-mail virus. The 1st rapidly spreading email viruses, as Melissa, made use of a microsoft Word macro embedded in an attachment.
* If the recepient opens the e-mail attachment, the word macro is activated then,

  (1) The email virus sends itself to everyone on the mailing list in the user's email package.

  (2) The virus does local damage.
* Spread using email with attachment containing a macro virus.
* Triggered when user opens attachment, or worse even when mail viewed by using scripting features in mail agent.
* Usually targeted at Microsoft outlook mail agent and word / Excel document.

Worms:-

* A worm is a program that can replicate itself and send Copies from Computer to Computer accross network Connections.
* Worm may be activated to replicate and propogate again. Network worm programs use network connections to spread from system to system.
* Once active within a system, a network worm can behave as a Computer virus / bacteria / it could implant trojan horse pgms or perform any no. of distributive / destructive actions.

* To replicate itself, a network worm uses some sort of network vehicle,

(1) Electronic Mail Facility :- A worm mails a copy of itself to other systems.

(2) Remote execution Capability :- A worm executes the copy of itself on another system.

(3) Remote Login Capability :- A worm logs onto a remote system as a user and then uses commands to copy itself from one system to the other.

* A network worm exhibits the same charecteristics as a Computer virus,
  - Dormant phase, propagation phase, Triggering phase, Execution phase
  - Propagation phase performs,
    1. Search for other systems to infect by examining host tables or similar repositories of remote system addresses
    2. Establish a connection with a remote system
    3. Copy itself to the remote system and cause the copy to be run.
* As with viruses, network worms are difficult to counter.

The Morris Worm:-

* The morris worm was designed to spread on UNIX systems and used a number of difficult techniques for propagation
* It attempt to log on to a remote host as a legitimate user. The worm first attempted to crack the local password file, then used the discovered passwords and corrosponding user IDs.

* The assumption was that many users would use the same password on different systems.

1. To obtain passwords the Worm ran a password cracking program that tried,
   - Each user's account name and simple permutations of it
   - A list of 432 built-in passwords that Morris thought to be likely candidates.
   - All the words in the local system directory.
2. It exploited a bug in the finger protocol, which reports the whereabouts of a remote user.
3. It exploited a trapdoor in the debug option of the remote process that receives and sends mail.

Worm Attacks:-

In late 2001, a more versatile Worm appeared known as Nimda. This Nimda Spreads by multiple mechanisms.

- From client to client via e-mail
- From client to client via open network shares
- From Web Server to client via browsing of compromised websites.
- From client to Web Server via active Scanning for and exploitation of various Microsoft IIS 4.0/5.0 directory traversal vulnerabilities.
- From client to Web Server via Scanning for backdoors left Behind by the Code Red II Worms.

5. VIRUS COUNTERMEASURES:

Antivirus Approaches:-

* The ideal solution to the threat of viruses is prevention. Do not allow the virus to get into the system in the first place.
* The next best approach is to be able to do the following.
  1. Detection
  2. Identification
  3. Removal.

Detection:- Once the infection has occured, determine that it has occared and locate the virus.

Identification:- Once detection has been acheived, identify the specific virus that has infected a program.

Removal :- once the specific virus has identified, remove all traces of the virus from the infected program and restore it to its original state.

* If detection succeeds but either identification and removal is not possible, then the alternate is to discard the infected program and reload a clean backup version.

There are 4 generations of antivirus software:-

First Generation :- Simple Scanners

Second Generation :- Heuristic Scanners

Third Generation :- Activity traps

Fourth Generation :- Full-featured protection.

First Generation Scanner:-

* Requires a virus signature to identify a virus. Such signature-specific scanners are limited to the detection of known viruses.
* Another type of 1st generation scanner maintains a record of the length of programs and looks for changes in length.

Second Generation Scanner:-

* Does not rely on a specific signature. Rather the scanner uses heuristic rules to search for probable virus infection.
* One class of such scanners looks for fragments of code that are often associated with viruses.
* Another 2nd gen. approach is integrity checking. A checksum can be appended to each program. If a virus infects the program without changing the checksum, then an integrity check will catch the change.

Third Generation programs:-

* These are memory resident programs that identify a virus by its actions rather than its structure in an infected program.
* Such programs have the advantage of not necessary to develop signatures & heuristic for wide array of viruses.

Fourth Generation Products:-

* These are packages consisting of variety of antivirus techniques used in conjunction. These include scanning and activity trap components.
* Such package includes access control capabilities.

Advanced Antivirus Approaches:-

* Advanced antiviruses are most sophisticated antivirus approaches and products that are as follows,

1. Generic Decryption
2. Digital Immune System.

Generic Decryption:- (GD)

* GD technology enables the antivirus program to easily detect even the most complex polymorphic viruses, while maintaing fast scanning speeds.
* Inorder to detect such a structure, executable files are ran through a GD Scanner, which contains the following,

1. CPU Emulator
2. Virus Signature Scanner
3. Emulation Control Module.

CPU Emulator:-

* A software based virtual computer. Instructions in an executable file are interpreted by the emulator rather than executed on the underlying processor.
* The emulator includes software versions of all registers and other processor hardware, so that the underlying processor is unaffected by programs interpreted to emulator.

Virus Signature Scanner:-

* A Module that scans the target code looking for known virus signatures.

Emulation Control Module:-

* It controls the execution of the target code.

Digital Immune System:-

* It is a Comprehensive approach to a virus protection developed by IBM.
* The motivation for this development has been the rising threat of Internet-based virus propagation.
* Two major trends in Internet technology have had an increasing impact on the rate of virus propagation.

1. Integrated mail systems
2. Mobile-program systems.

Integrated Mail Systems:-

* Systems such as Lotus notes and Microsoft outlook make it very simple to send anything to anyone and to work with objects that are received.

Mobile-Program Systems:-

* Capabilities such as Java and ActiveX allow programs to ~~re~~move on their own from one system to another system.

In response to the threat posed by these internet-base Capabilities, IBM has developed a prototype digital Immune System. This system expands the use of program emulation Provides general emulation and virus protection system.

Steps in Digital Immune System Operation:-

1. A monitoring program on each PC uses a variety of heuristics based on system behaviour, suspicious changes to programs, or family signature to infer that a virus may be present. The monitoring pgm forwards a copy of any program thought to be infected to an administrative machine within the organization.
2. The administrative machine encrypts the samples and sends it to a central virus analysis machine
3. The machine creates an environment in which the infected program can be safely run the analysis. The virus analysis machine then produces a prescription for identifying & removing the virus.
4. The resulting prescription is sent back to the administrative machine.
5. The administrative machine forwards the prescription to the infected client.
6. The prescription is also forwarded to other clients in the organization
7. Subscribers around the world receive regular antivirus updates that protect them from the new virus.

Behaviour - Blocking Software :-

* Unlike heuristics or fingerprint based scanners, behaviour blocking software integrates with the operating system of a host computer & monitors program behaviour in real time for malicious actions. Monitored behaviours can include the following,

  - Attempts to open, View, Delete, and/or modify files.
  - Attempts to format disk drives and other uncoverable disk operations.
  - Modifications to the logic of executable files or Macros.
  - Modification of critical system settings such as Strat-up settings.
  - Scripting of e-mail and instant messaging clients to send executable content.
  - Initiation of network communication.