SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

School of Computing
Department of Computer Science and Engineering

UNIT I - System Security and Practices - SCS7008

Building a secure organization- A Cryptography primer- detecting system Intrusion Preventing system Intrusion- Fault tolerance and Resilience in cloud computing environments- Security web applications, services and servers.

Building a Secure Organization

Using the Ten Steps for Building a Secure Organization. Consider a Real Scenario of Some Organization/company/super Market etc. and Discuss All Required Actions to Be Taken to Successfully Build a Secure Organization. Give Proper Examples and Discuss Every Step in Detail for the Chosen Organization.

1. Evaluate the Risks and Threats.
2. Beware of Common Misconceptions.
3. Provide Security Training for IT Staff Now and Forever.
4. Think "Outside the Box".
5. DOXing.
6. Train Employees: Develop a Culture of Security.
7. Identify and Utilize Built-in Security Features of the Operating System and Applications.
8. Monitor Systems.
9. Hire a Third Party to Audit Security.
10. Don't Forget the Basics

A Cryptography Primer

Cryptographic Attacks

Several attacks can be mounted against a cryptosystem. These are divided into two basic categories: passive attacks, where the adversary only monitors the communication channel, and active attacks, where the adversary not only monitors but also attempts to alter the communication. Thus, passive attacks threaten only the confidentiality of the communica tion, whereas active ones may also threaten integrity and authentication.

Passive attacks appear in several forms. In a ciphertext-only attack, the adversary attempts to obtain the plaintext by only observing the ciphertext. In this case a cryptosystem is considered totally insecure. In a known-plaintext attack, the adversary has a number of plaintexts and the corresponding ciphertexts and tries to obtain the plaintext of a certain target ciphertext. In a chosen-plaintext attack, the adversary is further allowed to choose the plaintexts for which he or she will be given the corresponding ciphertexts. In addition, in the adaptive chosen-plaintext attack the adversary is allowed to choose the next plaintext depending on the results of the previous plaintext requests. In a chosen ciphertext attack, theadversary selects ciphertexts and is given the corresponding plaintexts. From the information gained, the adversary tries to deduce the plaintext of a target ciphertext. Finally, in an adaptive chosen ciphertext attack, the choice of the next ciphertext may depend on the results of the previous ciphertext requests.

Security Models

The security of modern cryptography is based on *Kerckhoff 's assumption* under which the adversary (the opponent of the cryptosystem) has access to the encryption and decryption functions as well as the ciphertext messages but not to the secret key(s) used. If this assumption is not fulfilled and secrecy of the algorithms used is required, then the cryp tosystem is not considered secure and it belongs to the class known as "security through obscurity."

In order to evaluate the security of a cryptographic algorithm or protocol, several security evaluation models may be used. The highest security model is *unconditional security*. In this model the adversary is assumed to have unlimited computational power. The system is unconditionally secure if a ciphertext does not provide any information to the adversary regarding the plaintext. A necessary condition for a symmetric cryptosystem to be unconditionally secure is that the key must be equal in size with the plaintext, whileasymmetric cryptosystems cannot, by definition, be unconditionally secure. For these reasons, the unconditional security model is not practical.

A practical security model is *computational security.* In this model a cryptographic algorithm or protocol is considered computationally secure if the level of computational power required to defeat it by using the best-known attack is significantly higher than the expected computational resources that may be available to the adversary. Obviously, the security provided by this category weakens as the time passes since the computational powerof the adversary increases as technology improves and the cost of processing power is reduced. However, since this model is practical, most of the cryptosystems used today belong to this security model.

A special case of computational security is *provable security. A* cryptographic method is considered provably secure if the computational difficulty of defeating it can be reduced to the difficulty of solving a well-known and generally considered hard problem. Such problems are usually number-theoretic problems and are considered hard under certain assumptions. The most commonly used problems in cryptography are the integer factoriza tion problem and the discrete logarithm problem. This model of security is also verypopular.

Finally, another approach to measure security is *heuristic security*. The cryptographic algorithms and protocols that belong to this model are analyzed against known attacks. Only convincing claims of their resistance against these attacks can be provided taking into consideration resource requirements, but not a formal proof. Obviously, this is the weakest security model since several possible attacks may not have been considered. However, the security of several cryptographic protocols can only be considered in this model.

Having defined the basic terms regarding cryptographic algorithms, protocols, attacks, and security models, in the following section we define some basic functions that are widelyused in cryptography.

SYMMETRIC-KEY CRYPTOGRAPHY

Symmetric-key cryptography describes all the algorithms and protocols that use one key for both encryption and decryption per entity. First, we describe symmetric cryptosystems (symmetric ciphers), which are used to protect the privacy of a message. We explain the differences between the two variations of symmetric ciphers (stream and block), and we provide basic representatives of each variation along with a description of their character- istics. Then, we describe other uses of symmetric-key cryptography and in particular how symmetric cryptosystems can be combined with other cryptographic primitives such as hash functions to provide additional security objectives such as message integrity and origin authentication.

Symmetric Cryptosystems

As discussed earlier, a symmetric-key cryptosystem consists of a set of encryption and decryption transformations (functions) *E* and *D* which use the same secret information

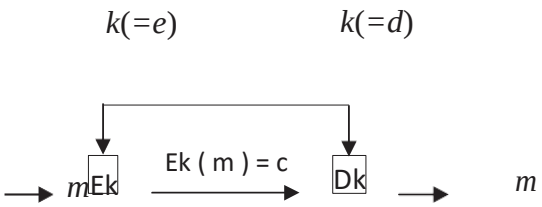


Fig.1 Symmetric cryptosystems.

(key) for both encryption (the enciphering key) and decryption (the deciphering key). Thus, if *e* denotes the enciphering key and *d* the deciphering key, in a symmetric cryptosystem *e* □ *d k*, where, for simplicity, *k* is called the secret (or private) key of the cryptosystem.
The encryption function *E* takes as input the plaintext message *m* and the secret key *k* and outputs the ciphertext *c*. The decryption function *D* takes as input the ciphertext *c* and the secret key *k* and outputs the original plaintext *m*. Figure A.1 describes the above process. Symmetric cryptosystems are divided into two categories: *stream ciphers* and *block ciphers*.

Stream Ciphers

Stream ciphers perform encryption and decryption of the plaintext one bit at a time. Stream ciphers are generally faster than block ciphers in hardware implementations as they require less complex hardware circuits. Thus, they are more appropriate in cases where buffering of the plaintext is limited or in cases where received characters must be encrypted and decrypted as they are received, such as in telecommunication applications.
With a stream cipher, the secret key k is used to generate a sequence of bits, known as the keystream. Encryption is performed by combining the keystream bits with the plaintext bits, usually with an XOR operation. If the generation of the keystream is independent of the plaintext and the ciphertext, then the stream cipher is called synchronous. Where the generation of the keystream depends on the plaintext and its encryption, the stream cipher is called self-synchronizing. Most stream cipher designs are synchronous.
A very interesting and well-studied instance of stream ciphers is the one-time pad, also known as the Vernam cipher. The one-time pad is of high theoretical interest since it has unconditional security. A one-time pad uses as a key a string of bits that is generated completely at random. The keystream has the same length as the plaintext message and it is XORed with the plaintext to produce the ciphertext. Since the entire keystream is random and used only once, the adversary has no better choice than random guessing, even if the adversary has unbounded computational power. The analysis of the one-time pad is considered one of the cornerstones of modern cryptography [. However, although the one- time pad offers unconditional security, there are practical dif- ficulties in its use, such as the key length and the key exchange. Although perfectly secure, it is impractical for generaluse. Practical stream ciphers attempt to simulate the one-time pad. They are either computationally or provably secure but not perfectly secure. Examples of widely used stream ciphers are A5 and RC4. Furthermore, certain modes of operation of a block cipher, referred to in the following paragraph, can be used as a stream cipher.

A5 is the cipher used in the Global System for Mobile Communications (GSM). Its actual key size is 40 bits, which makes the A5 cipher weak in brute-force attacks. RC4 is designed for keys of variable length, up to 2048 bits, with typical key sizes between 40

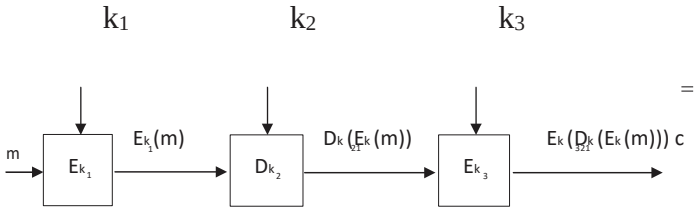


Fig.2 3-DES encryption.

and 256 bits. The key is used to initialize a 256-byte state table. The state table is used for subsequent generation of pseudorandom bytes and then to generate a pseudorandom stream, which is XORed with the plaintext to give the ciphertext. RC4 is used in many commercial applications and it is also part of the cellular specification. Although it is considered a strong cipher, weaknesses have been identified, mainly in its key scheduling algorithm. This indicates that special care should be taken in the implementations of the RC4 algorithm since widely used modes of operation, such as the mode of operation used in the Wired Equivalent Protocol (WEP) defined in the Institute of Electrical and Electron- ics Engineers (IEEE) standard 802.11b, were found insecure.

Block Ciphers

Block ciphers perform encryption and decryption on a group of characters (bits) of the plaintext. Examples of block ciphers are the data encryption standard (DES), triple-DES (3- DES), and advanced encryption standard (AES). DES is one of the first and most popular block ciphers. It is based on the data encryption algorithm, a 16-round Feistel cipher. It encrypts data by splitting it in blocks 64 bits long and by transforming the blocks with a 56- bit secret key.[2] Decryption is performed by using the same key and by "revers- ing" the transformations on the ciphertext blocks. DES was for many years the NIST- certified block cipher for use in transferred or stored data. A measure of the security of a cipher is its key length. The longer the key is, the harder it is for the adversary to brute force the algorithm, that is, try all possible keys. Although cryptanalytic efforts have not found practical, low- cost attacks on DES, there have been proposed attacks that are more effective than brute forcing, such as differential cryptanalysis and linear cryptanalysis. These new attacks minimize the effort to break DES from 2^{55} for a brute-force attack to 2^{47} for differential cryptanalysis and to 2^{43} for linear cryptanalysis. Thus, DES was replaced by 3-DES as the NIST-certified block cipher and more recently by AES.

3-DES is actually a DES encryption–decryption–encryption (EDE) sequence of the plaintextusing two or three different keys, as shown in Figure A.2. Let k_1, k_2, and k_3 be DES keys. Then, if m is the plaintext message, the corresponding 3-DES encryption c of m by using thekeys k_1, k_2, and k_3 is defined as $c = E_{k3}\{D_{k2}[E_{k1}(m)]\}$, where E_k and D_k denote a DES encryption and decryption with a key k, respectively. The keys k_1, k_2, and k_3 may be independent keys or the first and the last key may be the same ($k_1 = k_3$). Where all three keys are the same, then 3-DES reduces to the simple DES. FIPS 46-3 includes a definition of 3-DES. Although 3-DES is considered considerably more secure than simple DES, using two or three independent keys does not increase the security proportionally, as one might expect. In fact, it is not clear what is the level of additional security due to the repeated encryptions.

The full length of a DES key is 64 bits, including a parity of 8 bits. Thus the actual length of the

key is the remaining 56 bits.AES replaces DES and 3-DES as the NIST-approved block cipher. A basic requirement for AES candidate algorithms was the use of considerably stronger keys than DES and 3-DES, at least 128-bit keys. The algorithm that was selected as AES is the Rijndael algorithm. It uses variable key size and variable block size of 128, 192, or 256 bits. However, AES only allows for variable key size and defines block size at 128 bits. According to the key size, AES works with several numbers of rounds, 10, 12, and 14, for keys of size 128, 192, and 256 bits, respectively. The blocks are represented as arrays of bytes. In each round the bytes are transformed, the rows are rotated, and the columns are multiplied to a constant matrix. Each round is concluded with a XORing of the resulting arrayto the key. Other well-respected block ciphers that satisfy the security requirements of the advanced encryption standard are CAST-256, RC6, Twofish, Serpent, and MARS.

Block ciphers are used in what is called modes of operation. The operation mode must be at least as secure and as efficient as the underlying cipher. Each mode of operation has additional properties from the properties of the basic cipher. DES has four modes of opera- tion—electronic code book (EBC), cipher block chaining (CBC), cipher feedback (CF), and output feedback (OF)—which are described in FIPS 81. ECB mode encrypts each 64-bit
block of plaintext sequentially with the encryption key. With CBC mode, each block is first XORed with the previous ciphertext block and then it is encrypted with the key. Thus, the encryption of each block depends on previous blocks. CF and OFB modes allow use of DESas a stream cipher. A version of the standard generalized these modes to be applicable to a block cipher of any block size.
Apart from encryption, symmetric-key cryptosystems can be used to construct other cryptographic primitives, such as PRBGs and message authentication codes.

Characteristics of Symmetric-Key Cryptosystems

Symmetric-key cryptosystems, stream or block ones, have several features regarding their use and deployment. Bellow we provide a brief list of their positive and negative features.

Advantages of Symmetric-Key Cryptosystems

- **Efficiency.** Symmetric-key cryptosystems, stream or block ones, are in general very efficient algorithms. Hardware implementations of symmetric ciphers may encrypt hundreds of megabytes per second and software implementations may encrypt several megabytes per second.
- **Small Key Size.** Symmetric-key algorithms use keys of considerably smaller length in comparison with asymmetric ones for the same level of security. Typical key sizes for symmetric ciphers are 64, 80, 128, 192, and 256 bits, although at least 128 bits are required for adequate security. For asymmetric ciphers, typical key sizes are 512, 1024, 2048, and 4096 bits, although at least 1024 bits are required for adequate security.

Disadvantages of Symmetric-Key Cryptosystems

Number of Keys Required. While symmetric ciphers use keys of small size, the number of keys that are required increases dramatically with the number of com- municating entities. Consider the case where n entities must communicate with eachnother securely. Then, each entity must have n □1 keys, one key for every other entity. Thus, n(n □1) keys are required since each pair needs one key, this is reduced to a total number of n(n 1)/2 keys. This leads to a need for 10

different keys for 5 communicating entities, which grows to 1.225 keys for 50 entities and to 124.750 different keys for 500 entities.

Key Management. Due to the increase in the number of keys required, it is difficult to manage the keys in symmetric-key cryptosystems.

Key Exchange. The secret key that each communicating pair will share for encryption must be somehow securely exchanged between each pair of communicating entities.

Secrecy of the Key. Since each key is shared between two entities, the secrecy of the key must be protected at both sides. Furthermore, if a key is compromised, it is not clear which side has been compromised.

Detecting System Intrusion Preventing System Intrusion

Intrusion Detection System: An intrusion detection system (IDS) is a device, typically another separate computer, that monitors activity to identify malicious or suspicious events. An IDS is a sensor, like a smoke detector, that raises an alarm if specific things occur. A model of an IDS is shown in below figure. The components in the figure are the four basic elements of an intrusion detection system, based on the Common Intrusion Detection Framework of [STA96]. An IDS receives raw inputs from sensors. It saves those inputs, analyzes them, and takes some controlling action.

Types of IDSs

The two general types of intrusion detection systems are signature based and heuristic. Signature- based intrusion detection systems perform simple patternmatching and report situations that match a pattern corresponding to a known attack type. Heuristic intrusion detection systems, also known as anomaly based, build a model of acceptable behavior and flag exceptions to that model; for the future, the administrator can mark a flagged behavior as acceptable so that the heuristic IDS will now treat that previously unclassified behavior as acceptable. Intrusion detection devices can be network based or host based. A network-based IDS is a stand-alone device attached to the network to monitor traffic throughout that network; a host-based IDS runs on a single workstation or client or host, to protect that one host.

Signature-Based Intrusion Detection:

A simple signature for a known attack type might describe a series of TCP SYN packets sent to many different ports in succession and at times close to one another, as would be the case for a port scan. An intrusion detection system would probably find nothing unusual in the first SYN, say, to port 80, and then another (from the same source address) to port 25. But as more and more ports receive SYN packets, especially ports that are not open, this pattern reflects a possible port scan. Similarly, some implementations of the protocol stack fail if they receive an ICMP packet with a data length of 65535 bytes, so such a packet would be a pattern for which to watch.

Heuristic Intrusion Detection:

Because signatures are limited to specific, known attack patterns, another form of intrusion detection becomes useful. Instead of looking for matches, heuristic intrusion detection looks for behavior that isout of the ordinary. The original work in this area focused on the individual, trying to find characteristics of that person that might be helpful in understanding normal and abnormal behavior. For example, one user might always start the day by reading e-mail, write many documents using a word processor, and occasionally back up files. These actions would be normal. This user does not seem to use many administrator utilities.

If that person tried to access sensitive system management utilities, this new behavior might be a clue that someone else was acting under the user's identity. Inference engines work in two ways. Some, called state-based intrusion detection systems, see the system going through changes of overall state or configuration. They try to detect when the systemhas veered into unsafe modes. Others try to map current activity onto a model of unacceptable activity
and raise an alarm when the activity resembles the model. These are called model-based intrusion detection systems. This approach has been extended to networks in [MUK94]. Later work sought to build a dynamic model of behavior, to accommodate variation and evolution in a person's actions over time. The technique compares real activity with a known representation of normality. Alternatively, intrusion detection can work from a model of known bad activity. For example, except for a few utilities (login, change password, create user), any other attempt to access a password file is suspect. This form of intrusion detection is known as misuse intrusion detection. In this work, the real activity is compared against a known suspicious area.

Stealth Mode:

An IDS is a network device (or, in the case of a host-based IDS, a program running on a network device). Any network device is potentially vulnerable to network attacks. How useful would an IDS be if it itself were deluged with a denial-of-service attack? If an attacker succeeded in logging in to a system within the protected network, wouldn't trying to disable the IDS be the next step? To counter those problems, most IDSs run in stealth mode, whereby an IDS has two network interfaces: one for the network (or network segment) being monitored and the other to generate alerts and perhaps other administrative needs. The IDS uses the monitored interface as input only; it never sends packets out through that interface. Often, the interface is configured so that the device has no published address through the monitored interface; that is, a router cannot route anything to that address directly, because the router does not know such a device exists. It is the perfect passive wiretap. If the IDS needs to generate an alert, it uses only the alarm interface on a completely separate control network.

Goals for Intrusion Detection Systems:

1. **Responding to alarms**

Whatever the type, an intrusion detection system raises an alarm when it finds a match. The alarm canrange from something modest, such as writing a note in an audit log, to something significant, such as paging the system security administrator. Particular implementations allow the user to determine what action the system should take on what events.

In general, responses fall into three major categories (any or all of which can be used in a single response):

Monitor, collect data, perhaps increase amount of data collectedProtect,

act to reduce exposure

Call a human

2. **False Results**

Intrusion detection systems are not perfect, and mistakes are their biggest problem. Although an IDS might detect an intruder correctly most of the time, it may stumble in two different ways: by raisingan alarm for

something that is not really an attack (called a false positive, or type I error in the statistical community) or not raising an alarm for a real attack (a false negative, or type II error). Too many false positives means the administrator will be less confident of the IDS's warnings, perhaps leading to a real alarm's being ignored. But false negatives mean that real attacks are passing the IDS without action. We say that the degree of false positives and false negatives represents the sensitivity of the system. Most IDS implementations allow the administrator to tune the system's sensitivity, to strike an acceptable balance between false positives and negatives.

IDS Strength and Limitations

On the upside, IDSs detect an ever-growing number of serious problems. And as we learn more about problems, we can add their signatures to the IDS model. Thus, over time, IDSs continue to improve. At the same time, they are becoming cheaper and easier to administer. On the downside, avoiding an IDS is a first priority for successful attackers. An IDS that is not well defended is useless. Fortunately,stealth mode IDSs are difficult even to find on an internal network, let alone to compromise. IDSs look for known weaknesses, whether through patterns of known attacks or models of normal behavior. Similar IDSs may have identical vulnerabilities, and their selection criteria may miss similarattacks. Knowing how to evade a particular model of IDS is an important piece of intelligence passed within the attacker community. Of course, once manufacturers become aware of a shortcoming in their products, they try to fix it. Fortunately, commercial IDSs are pretty good at identifying attacks. Another IDS limitation is its sensitivity, which is difficult to measure and adjust. IDSs will never be perfect, so finding the proper balance is critical. In general, IDSs are excellent additions to a network'ssecurity. Firewalls block traffic to particular ports or addresses; they also constrain certain protocolsto limit their impact. But by definition, firewalls have to allow some traffic to enter a protected area. Watching what that traffic actually does inside the protected area is an IDS's job, which it does quite well.

MALICIOUS SOFTWARE

The most sophisticated types of threats to computer systems are presented by programs that exploit vulnerabilities in computing systems. Such threats are referred to as malicious software, or malware. Malicious software can be divided into two categories: those that needa host program, and those that are independent. The former, referred to as parasitic, are essentially fragments of programs that cannot exist independently of some actual application program, utility, or system program. Viruses, logic bombs, and backdoors are examples. Independent malware is a self-contained program that can be scheduled and run by the operating system. Worms and bot programs are examples. key categories of malicious software:

Backdoor

A backdoor, also known as a trapdoor, is a secret entry point into a program that allows someone who is aware of the backdoor to gain access without going through the usual security access procedures. Programmers have used backdoors legitimately for many years to debug and test programs; such a backdoor is called a maintenance hook. This usually is done when the programmer is developing an application that has an authentication procedure, or a long setup, requiring the user to enter many different values to run the application. To debug the program, the developer may wish to gain special privileges or to avoid all the necessary setup and authentication. The programmer may also want to ensure that there is a method of activating the

program should something be wrong with the authentication procedure that is being built into the application. The backdoor is code that recognizes some special sequence of input or is triggered by being run from a certain user ID or by an unlikely sequence of events. Logic Bomb One of the oldest types of program threat, predating viruses and worms, is the logic bomb.The logic bomb is code embedded in some legitimate program that is set to "explode" when certain conditions are met. Examples of conditions that can be used as triggers for a logic bomb are the presence or absence of certain files, a particular day of the week or date, or a particular user running the application. Once triggered, a bomb may alteror delete data or entire files, cause a machine halt, or do some other damage.

Trojan Horses

A Trojan horse is a useful, or apparently useful, program or command procedure containing hidden code that, when invoked, performs some unwanted or harmful function. Trojan horse programs can be used to accomplish functions indirectly that an unauthorized user could not accomplish directly. For example, to gain access to the files of another user on a shared system, a user could create a Trojan horse program that, when executed, changes the invoking user's file permissions so that the files are readable by any user. The author could then induce users to run the program by placing it in a common directory and naming it such that it appears to be a useful utility program or application. An example is a program that ostensibly produces a listing of the user's files in a desirable format. After another user has run the program, the author of the program can then access the information in the user's files. An example of a Trojan horse program that would be difficult to detect is a compiler that has been modified to insert additional code into certain programs as they are compiled, such as a system login program. The code creates a backdoor in the login program that permits the author to log on to the system using a special password.This Trojan horse can never be discovered by reading the source code of the login program. Trojan horses fit into one ofthree models:

• Continuing to perform the function of the original program and additionally performing a separate malicious activity
• Continuing to perform the function of the original program but modifying the function to perform malicious activity (e.g., a Trojan horse version of a login program that collects passwords) or to disguise other malicious activity (e.g., a Trojan horse version of a process listing program that does not display certain processes that are malicious)

• Performing a malicious function that completely replaces the function of the originalprogram

Mobile Code

Mobile code refers to programs (e.g., script, macro, or other portable instruction) that can be shipped unchanged to a heterogeneous collection of platforms and execute with identical semantics [JANS01]. The term also applies to situations involving a large homogeneous collection of platforms (e.g., Microsoft Windows).

Mobile code is transmitted from a remote system to a local system and then executed on the local system without the user's explicit instruction. Mobile code often acts as a mechanism for a virus,

worm, or Trojan horse to be transmitted to the user's workstation. In other cases, mobile code takes advantage of vulnerabilities to perform its own exploits, such as unauthorized data access or root compromise. Popular vehicles for mobile code include Java applets, ActiveX, JavaScript, and VBScript.The most common ways of using mobile code for malicious operations on local system are cross-site scripting, interactive and dynamic Web sites, e-mail attachments, and downloads from untrusted sites or of untrusted software.

Multiple-Threat Malware

Viruses and other malware may operate in multiple ways. The terminology is far from uniform; this subsection gives a brief introduction to several related concepts that could be considered multiple-threat malware. A multipartite virus infects in multiple ways. Typically, the multipartite virus is capable of infecting multiple types of files, so that virus eradication must deal with all of the possible sites of infection. A blended attack uses multiple methods of infection or transmission, to maximize the speed of contagion and the severity of the attack. Some writers characterize a blended attack as a package that includes multiple typesof malware. An example of a blended attack is the Nimda attack, erroneously referred to as simply a worm.

Viruses

A computer virus is a piece of software that can "infect" other programs by modifying them; the modification includes injecting the original program with a routine to make copies of the virus program, which can then go on to infect other programs. A computer virus carries in its instructional code the recipe for making perfect copies of itself. The typical virus becomes embedded in a program on a computer. Then, whenever the infected computer comes into contact with an uninfected piece of software, a fresh copy of the virus passes into the new program. Thus, the infection can be spread from computer to computer by unsuspecting users who either swap disks or send programs to one another over a network. In a network environment, the ability to access applications and system services on other computers provides a perfect culture for the spread of a virus. A virus can do anything that other programs do. The difference is that a virus attaches itself to another program and executes secretly when the host program is run. Once a virus is executing, it can perform any function, such as erasing files and programs that is allowed by the privileges of the current user. A computer virus has three parts:

• Infection mechanism: The means by which a virus spreads, enabling it to replicate. The mechanism is also referred to as the infection vector.

• Trigger: The event or condition that determines when the payload is activated or delivered.

• Payload: What the virus does, besides spreading.

The FIREWALLS

Firewalls were officially invented in the early 1990s, but the concept really reflects the reference monitor from two decades earlier. What is a Firewall? A firewall is a device that filters all traffic between a protected or "inside" network and a less trustworthy or "outside" network. Usually a firewall runs on a dedicated device; because it is a single point through which traffic is channeled,

performance is important, which means non-firewall functions should not be done on the same machine. Because a firewall is executable code, an attacker could compromise that code and execute from the firewall's device. Thus, the fewer pieces of code on the device, the fewer tools the attacker would have by compromising the firewall. Firewall code usually runs on a proprietary or carefully minimized operating system. The purpose of a firewall is to keep "bad" things outside a protected environment. To accomplish that, firewalls implement a security policy that is specifically designed to address what bad things might happen. For example, the policy might be to prevent any access from outside (while still allowing traffic to pass from the inside to the outside). Alternatively, the policy might permit accesses only from certain places, from certain users, or for certain activities. Part of the challenge of protecting a network with a firewall is determining which security policy meets the needs of the installation.

Design of Firewalls:
A reference monitor must be Always invoked Tamperproof Small and simple enough for rigorous analysis A firewall is a special form of reference monitor. By carefully positioning a firewall within a network, we can ensure that all network accesses that we want to control must pass through it. This restriction meets the "always invoked" condition. A firewall is typically well isolated, making it highly immune to modification. Usually a firewall isimplemented on a separate computer, with direct connections only to the outside and inside networks. This isolation is expected to meet the "tamperproof" requirement. And firewall designers strongly recommend keeping the functionality of the firewall simple. Types of Firewalls: Firewalls have a wide range of capabilities.

Types of firewalls include

Packet filtering gateways or screening routers

Stateful inspection firewalls

Application proxie

Guards

Personal firewalls Packet Filtering Gateway:

A packet filtering gateway or screening router is the simplest, and in some situations, themost effective type of firewall. A packet filtering gateway controls access to packets on the basis of packet address (source or destination) or specific transport protocol type (such as HTTP web traffic). As described earlier in this chapter, putting ACLs on routers may severely impede their performance. But a separate firewall behind (on the local side) of the router can screen traffic before it gets to the protected network. Figure 7-34 shows a packet filter that blocks access from (or to) addresses in one network; the filter allows HTTP traffic but blocks traffic using the Telnet protocol.

Stateful Inspection Firewall:

Filtering firewalls work on packets one at a time, accepting or rejecting each packet and moving on to the next. They have no concept of "state" or "context" from one packet to the next. A stateful

inspection firewall maintains state information from one packet to another in the input stream. One classic approach used by attackers is to break an attack into multiple packets by forcing some packets to have very short lengths so that a firewall cannot detect the signature of an attack split across two or more packets. (Remember that with the TCP protocols, packets can arrive in any order, and the protocol suite is responsible for reassembling the packet stream in proper order before passing it along to the application.) A
stateful inspection firewall would track the sequence of packets and conditions from one packet to another to thwart such an attack.

Application Proxy:

Packet filters look only at the headers of packets, not at the data inside the packets. Therefore, a packet filter would pass anything to port 25, assuming its screening rules allow inbound connections to that port. But applications are complex and sometimes contain errors. Worse, applications (such as the e-mail delivery agent) often act on behalf of all users, so they require privileges of all users (for example, to store incoming mail messages so that inside users can read them). A flawed application, running with all users' privileges, can cause much damage. An application proxy gateway, also called a bastion host, is a firewall that simulates the (proper) effects of an application so that the application receives only requests to act properly. A proxy gateway is a two-headed device: It looks to the inside as if it is the outside (destination) connection, while to the outside it responds just as the insider would. An application proxy runs pseudo-applications. For instance, when electronic mail is transferred to a location, a sending process at one site and a receiving process at the destination communicate by a protocol that establishes the legitimacy of a mail transfer and then actually transfers the mail message. The protocol between sender and destination is carefully defined. A proxy gateway essentially intrudes in the middle of this protocol exchange, seeming like a destination in communication with the sender that is outside the firewall, and seeming like the sender in communication with the real destination on the inside. The proxy in the middle has the opportunity to screen the mail transfer, ensuring that only acceptable e-mail protocol commands are sent to the destination.

Guard:

A guard is a sophisticated firewall. Like a proxy firewall, it receives protocol data units, interprets them, and passes through the same or different protocol data units that achieve either the same result or a modified result. The guard decides what services to perform on the user's behalf in accordance with its available knowledge, such as whatever it can reliably know of the (outside) user's identity, previous interactions, and so forth. The degree of control a guard can provide is limited only by what is computable. But guards and proxy firewalls are similar enough that the distinction between them is sometimes fuzzy. That is, wecan add functionality to a proxy firewall until it starts to look a lot like a guard.

Personal Firewalls:

A personal firewall is an application program that runs on a workstation to block unwanted traffic, usually from the network. A personal firewall can complement the work of a conventional firewall by screening the kind of data a single host will accept, or it can

compensate for the lack of a regular firewall, as in a private DSL or cable modem connection. The personal firewall is configured to enforce some policy. For example, the user may decide that certain sites, such as computers on the company network, are highly trustworthy, but most other sites are not. The user defines a policy permitting download of code, unrestricted data sharing, and management access from the corporate segment, but not from other sites. Personal firewalls can also generate logs of accesses, which can be useful to examine in case something harmful does slip through the firewall. A personal firewall runs on the very computer it is trying to protect. Thus, a clever attacker is likely to attempt an undetected attack that would disable or reconfigure the firewall for the future. Still, especially for cable modem, DSL, and other "always on" connections, the static workstation is a visible and vulnerable target for an everpresent attack community. A personal firewall can provide reasonable protection to clients that are not behind a network firewall.

Firewall Configurations

• In addition to the use of a simple configuration consisting of a single system, more complex configurations are possible and indeed more common.

• Figure illustrates three common firewall configurations.

• Figure (a) shows the “screened host firewall, single-homed bastion configuration”, where the firewall consists of two systems:

• a packet-filtering router - allows Internet packets to/from bastion only

• a bastion host - performs authentication and proxy functions

• This configuration has greater security, as it implements both packet-level & application- level filtering, forces an intruder to generally penetrate two separate systems to compromise internal security, & also affords flexibility in providing direct Internet access to specific internal servers (eg web) if desired.

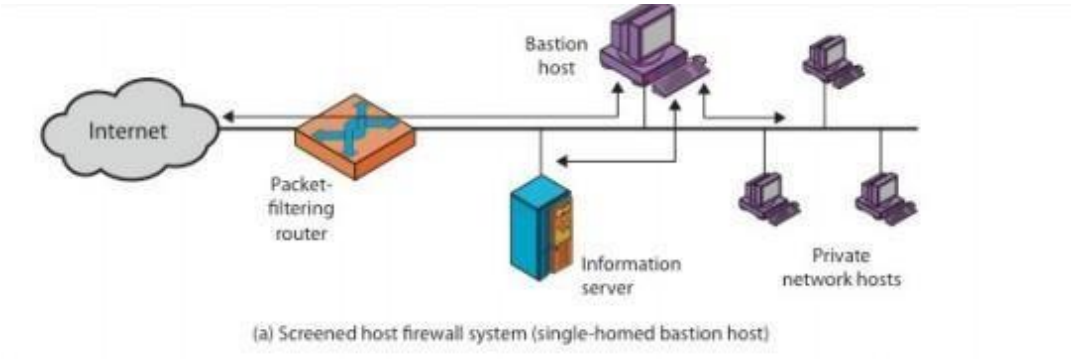


Fig. 4 Screen Host Firewall System (Single-Homed bastion host)

Figure (b) illustrates the "screened host firewall, dual-homed bastion configuration" which physically separates the external and internal networks, ensuring two systems must be compromised to breach security. The advantages of dual layers of security are also present here. Again, an information server or other hosts can be allowed direct communication with the router if this is in accord with the security policy, but are now separated from the internal network.

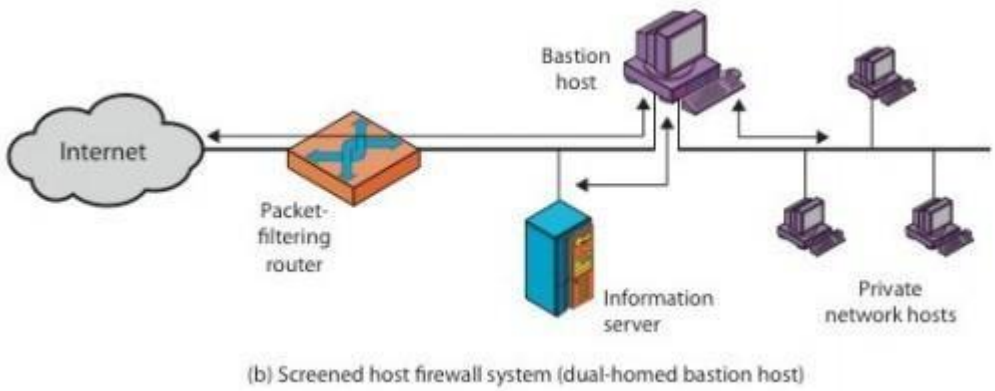


Fig. 5 Screen Host Firewall System (Dual-Homed bastion host)

Figure(c) shows the "screened subnet firewall configuration", being the most secure shown. It has two packet-filtering routers, one between the bastion host and the Internet and the other between the bastion host and the internal network, creating an isolated subnetwork. This may consist of simply the bastion host but may also include one or more information servers and modems for dial-in capability. Typically, both the Internet and the internal network have access to hosts on the screened subnet, but traffic across the screened subnet is blocked. This configuration offers several advantages: • There are now three levels of defense to thwart intruders • The outside router advertises only the existence of the screened subnet to the Internet; therefore the internal network is invisible to the Internet • Similarly, the inside router advertises only the existence of the screened subnet to the internal network; hence systems on the inside network cannot construct direct routes to the Internet

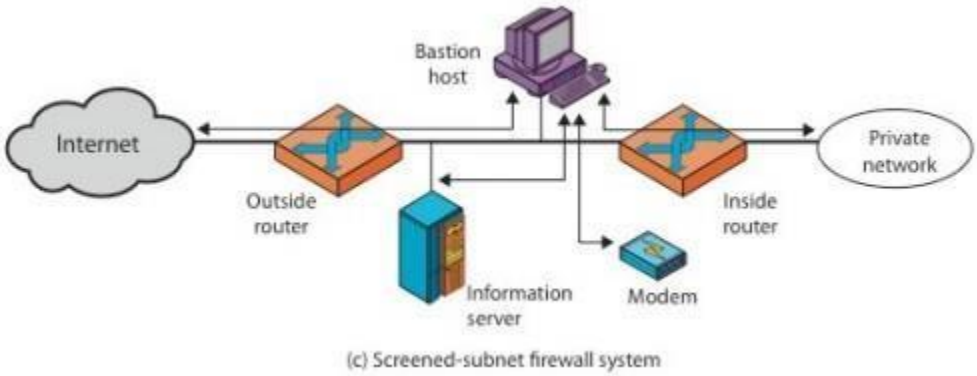


Fig. 6 Screened Subnet Firewall System

1.5 Security Web Applications, Services and Servers

WEB SECURITY

SSL Architecture

SSL is designed to make use of TCP to provide a reliable end-to-end secure service. SSL is not a single protocol but rather two layers of protocols, as illustrated in Figure. The SSL Record Protocol provides basic security services to various higher layer protocols. In particular, the Hypertext Transfer Protocol (HTTP), which provides the transfer service for Web client/server interaction, can operate on top of SSL. Three higher-layer protocols are defined as part of SSL: the Handshake Protocol, The Change Cipher Spec Protocol, and the Alert Protocol. These SSL-specific protocols are used in the management of SSL exchange. Two important SSL concepts are the SSL session and the SSL connection, which are defined in the specification as follows. • Connection: A connection is a transport (in the OSI layering model definition) that provides a suitable type of service. For SSL, such connections are peer-to-peer relationships. The connections are transient. Every connection is associated with one session. • Session: An SSL session is an association between a client and a server. Sessions are created by the Handshake Protocol. Sessions define a set of cryptographic

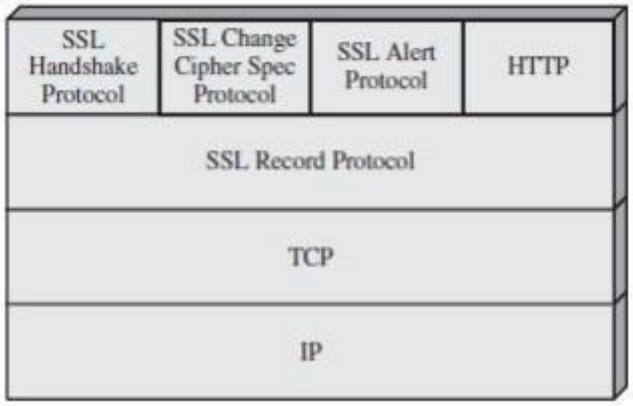


Figure SSL Protocol Stack

Fig. 7 SSL Protocol Stack

security parameters which can be shared among multiple connections. Sessions are used to avoid the expensive negotiation of new security parameters for each connection. Between any pair of parties (applications such as HTTP on client and server), there may be multiple secure connections. In theory, there may also be multiple simultaneous sessions between parties, but this feature is not used in practice. There are a number of states associated with each session. Once a session is established, there is a current operating state for both read and write (i.e., receive and send). In addition, during the Handshake Protocol, pending read and write states are created. Upon successful conclusion of the Handshake Protocol, the pending states become the current states.

A session state is defined by the following parameters.

• **Session identifier:** An arbitrary byte sequence chosen by the server to identify an active or resumable session state.

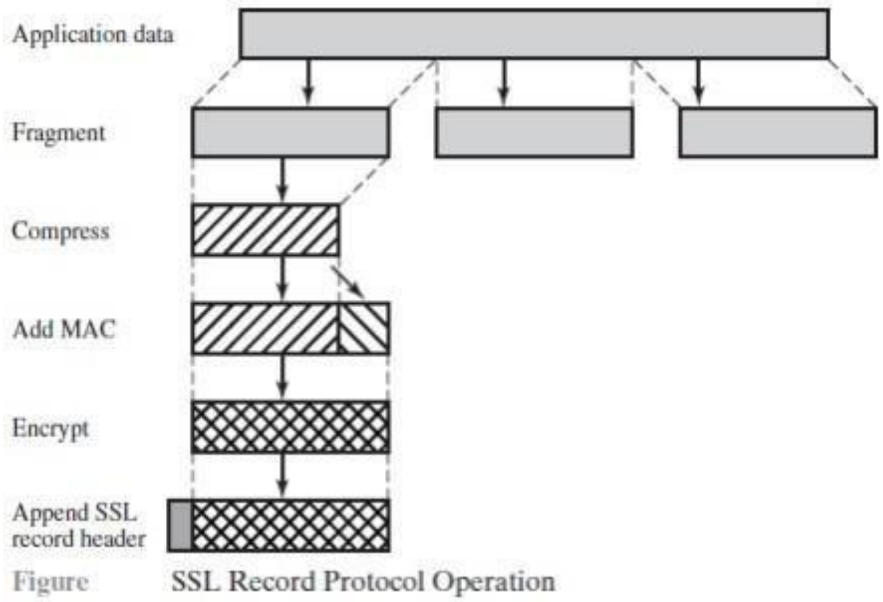


Fig. 8 SSL Record Protocol Operation

Peer certificate: An X509.v3 certificate of the peer. This element of the state may be null.

• **Compression method:** The algorithm used to compress data prior to encryption.

• **Cipher spec:** Specifies the bulk data encryption algorithm (such as null,AES, etc.) and a hash algorithm (such as MD5 or SHA-1) used for MAC calculation. It also defines cryptographic attributes such as the hash_size.

• **Master secret:** 48-byte secret shared between the client and server.

• **Is resumable:** A flag indicating whether the session can be used to initiate new connections. A connection state is defined by the following parameters.

• **Server and client random:** Byte sequences that are chosen by the server and client for each connection.

• **Server write MAC secret**: The secret key used in MAC operations on data sent by the server.

• **Client write MAC secret:** The secret key used in MAC operations on data sent by the client. • Server write key: The secret encryption key for data encrypted by the server and decrypted by the client.

• **Client write key:** The symmetric encryption key for data encrypted by the client and decrypted by the server.

• **Initialization vectors:** When a block cipher in CBC mode is used, an initialization vector (IV) is maintained for each key.This field is first initialized by the SSL Handshake Protocol. Thereafter, the final ciphertext block from each record is preserved for use as the IV with the

following record.

• **Sequence numbers:** Each party maintains separate sequence numbers for transmitted and received messages for each connection. When a party sends or receives a change cipher spec message, the appropriate sequence number is set to zero. Sequence numbers may not exceed 264 – 1.

SSL Record Protocol

The SSL Record Protocol provides two services for SSL connections:

• Confidentiality: The Handshake Protocol defines a shared secret key that is used for conventional encryption of SSL payloads.

• Message Integrity: The Handshake Protocol also defines a shared secret key that is used to form a message authentication code (MAC).

Figure indicates the overall operation of the SSL Record Protocol. The Record Protocol takes an application message to be transmitted, fragments the data into manageable blocks, optionally compresses the data, applies a MAC, encrypts, adds a header, and transmits the resulting unit in a TCP segment. Received data are decrypted, verified, decompressed, and reassembled before being delivered to higher-level users. The first step is fragmentation. Each upper-layer message is fragmented into blocks of 214 bytes (16384 bytes) or less. Next, compression is optionally applied. Compression must be lossless and may not increase the content length by more than 1024 bytes.1In SSLv3 (as well as the current version of TLS), no compression algorithm is specified, so the default compression algorithm is null. The next step in processing is to compute a message authentication code over the compressed data. For this purpose, a shared secret key is used. The calculation is defined as hash(MAC_write_secret || pad_2||

hash(MAC_write_secret || pad_1||seq_num ||

SSLCompressed.type || SSLCompressed.length ||

SSLCompressed.fragment))

where || = concatenation

MAC_write_secret = shared secret key

hash = cryptographic hash algorithm; either

MD5 or SHA-1

pad_1 = the byte 0x36 (0011 0110) repeated

48 times (384 bits) for MD5 and 40

times (320 bits) for

SHA-1 pad_2 = the byte 0x5C (0101 1100) repeated 48

times for MD5 and 40 times for SHA-1

seq_num = the sequence number for this message

SSLCompressed.type = the higher-level protocol used to process this fragment
SSLCompressed.length = the length of the compressed fragment
SSLCompressed.fragment = the compressed fragment (if compression is not used, this is the plaintext fragment)

Question Bank:

1. Discuss the steps must be followed to build any secure organization.
2. Criticize the mechanisms used for securing web applications.
3. Examine fault tolerance in cloud environment to maintain a secured environment.
4. Elaborate any three types of substitutional ciphers with example
5. Explain the cloud computing architecture and basic fault tolerance concepts
6. Encrypt the given plaintext: COMMUNICATIONS SECURITY using Playfair Cipher. The key is GALOIS
7. What is the difference between authentication, integrity, confidentiality and nonrepudiation?
8. Explain the Importance of monitoring systems are in achieving security policies?
9. Demonstrate the XOR cipher with logical operands.
10. Discuss various cryptographic devices used in older days.
11. Compare substitutional ciphers and block ciphers.
12. Summarize the security policies need to be followed by every network.
13. What are the issues in information security and network security? How they can be solved?
14. Discuss Spam Filtering controls imposed by various email vendors.
15. Cipher: EXXEGOEXSRGI.
    Find the plaintext from the above cipher text by using the Vigenère substitutional cipher method and the key is MONARCHY.

School of Computing
Department of Computer Science and Engineering

UNIT – II - System Security and Practices SCS7008

Internet Security - Botnet Problem- Intranet security- Local Area Network Security - Wireless Network Security - Wireless Sensor Network Security- Cellular Network Security, OpticalNetwork Security- Optical wireless Security.

Internet Security

The term malware is a contraction of malicious software. Put simply, malware is any piece of software that was written with the intent of damaging devices, stealing data, and generally causing a mess. Viruses, Trojans, spyware, and ransomware are among the different kinds of malware.

Malware is often created by teams of hackers: usually, they're just looking to make money, either by spreading the malware themselves or selling it to the highest bidder on the Dark Web. However, there can be other reasons for creating malware too — it can be used as a tool for protest, a way to test security, or even as weapons of war between governments.

What does malware do?

All kinds of things. It's a very broad category, and what malware does or how malware works changes from file to file. The following is a list of common types of malware, but it's hardly exhaustive:

- Virus: Like their biological namesakes, viruses attach themselves to clean files and infect other clean files. They can spread uncontrollably, damaging a system's core functionality and deleting or corrupting files. They usually appear as an executable file (.exe).
- Trojans: This kind of malware disguises itself as legitimate software, or is hidden in legitimate software that has been tampered with. It tends to act discreetly and create backdoors in your security to let other malware in.
- Spyware: No surprise here — spyware is malware designed to spy on you. It hides in the background and takes notes on what you do online, including your passwords, credit cardnumbers, surfing habits, and more.
- Worms: Worms infect entire networks of devices, either local or across the internet, by using network interfaces. It uses each consecutively infected machine to infect others.
- Ransomware: This kind of malware typically locks down your computer and your files, and threatens to erase everything unless you pay a ransom.
- Adware: Though not always malicious in nature, aggressive advertising software can undermine your security just to serve you ads — which can give other malware an easy way in. Plus, let's face it: pop-ups are really annoying.
- Botnets: Botnets are networks of infected computers that are made to work together under the control of an attacker.

How to protect against malware

When it comes to malware, prevention is better than a cure. Fortunately, there are some commonsense, easy behaviors that minimize your chances of running into any nasty software.

- Don't trust strangers online! "Social engineering", which can include strange emails, abrupt alerts, fake profiles, and curiosity-tickling offers, are the #1 method of delivering malware. If you don't know exactly what it is, don't click on it.
- Double-check your downloads! From pirating sites to official storefronts, malware is often lurking just around the corner. So before downloading, always double-check thatthe provider is trustworthy by carefully reading reviews and comments.

- Get an ad-blocker! Malvertising – where hackers use infected banners or pop-up ads to infect your device – is on the rise. You can't know which ads are bad: so it's safer to just block them all with a reliable ad-blocker.

- Careful where you browse! Malware can be found anywhere, but it's most common in websites with poor backend security, like small, local websites. If you stick to large, reputable sites, you severely reduce your risk of encountering malware.

Unfortunately, even if you follow the above advice to the letter, you might still get infected with malware: hackers have found ways to sneak their viruses into every corner of the web. For real security, you need to combine healthy online habits with powerful and reliable anti-malware software, like AVG AntiVirus FREE, which detects and stops malware before it infects your PC,Mac, or mobile device.

How to detect malware

Certain strains of malware are easier to detect than others. Some, like ransomware and adware, make their presence known immediately, either by encrypting your files or by streaming endless ads at you. Others, like Trojans and spyware, go out of their way to hide from you as long as possible, meaning they could be on your system a long time before you realize that they're present. And then there are others, like viruses and worms, that might operate in secret for atime, before the symptoms of their infection start to appear, such as freezing, deleted or replaced files, sudden shutdowns, or a hyperactive processor.

The only surefire way to detect all malware before it infects your PC, Mac, or mobile is toinstall anti-malware software, which will come packaged with detection tools and scans that can catch malware currently on your device, as well as block malware trying to infect it.

How to remove malware

Each form of malware has its own way of infecting and damaging computers and data, and so each one requires a different malware removal method. To get started, check out our tipsfor getting rid of viruses and malware.
That said, the best way to stay protected or remove an infection is to use anti-malware software, more commonly called an antivirus. The best malware removal tools are included in the most advanced antivirus, and even free ones like AVG AntiVirus FREE have all you need to stay safe from the most common threats.

Malware on Android

PCs aren't the only devices that get malware: any device that can connect to the internet

is atrisk, and that includes your Android phone. While you might not hear about them as much, Android attacks are on the rise, with phishing websites, fake apps, and unofficial app stores being the main distributors of dangerous software.

Android malware, much like PC malware, can do any number of things and cause all kinds of damage. There are viruses, ransomware, botnets, and trojans, spyware, and more: just whatyou'd expect in this era of smart devices!

Fortunately, if a device can be attacked, it can also be secured, and Android phones are no exception. Download a free Android antivirus like AVG AntiVirus for Android to make sure your phone is protected against anything and everything that might threaten it online.

Malware on Mac

Macs have a reputation for being virus-proof, and while that was never true, it was true that, for a long time, the amount of malware that could infect a Mac was laughably small. But while the number of threats for Mac are still small when compared to the enormous malware library that attacks PCs, it's no longer small enough to ignore. There's a very real threat your Mac could become infected if you're not careful and if it's not secured with a powerful, trustworthy antivirus.
That's why we recommend you download AVG AntiVirus for Mac to ensure your favorite Apple laptop or desktop isn't compromised by any malware threats that may be lurking on the web.

WHAT IS A BOTNET?
What is a botnet?
A botnet is a network of compromised computers under the control of a malicious actor. Each individual device in a botnet is referred to as a bot. A bot is formed when a computer gets infected with malware that enables third-party control. Bots are also known as "zombie computers" due to their ability to operate under remote direction without their owners' knowledge. The attackers that control botnets are referred to as "bot herders" or "bot masters."

Attackers use botnets for a variety of purposes, many of them criminal. The most common applications for botnets include email spam campaigns, denial-of-service attacks, spreading adware/spyware, and data theft (particularly of financial information, online identities and user logins). A botnet attack starts with bot recruitment. Bot herders often recruit bots by spreading botnet viruses, worms, or other malware; it is also possible to use web browser hacking to infect computers with bot malware. Once a computer has been infected with a botnet virus it will connect back to the bot herder's command and control (C&C) server. From here the attacker is capable of communicating with and controlling the bot. When the botnet grows to its desiredsize, the herder can exploit the botnet to carry out attacks (stealing information, overloading servers, click fraud, sending spam, etc).

Example: Zeus Botnets

Zeus is a Trojan horse for Windows that was created to steal bank information using botnets. First discovered in 2007, Zeus spread through email, downloads, and online messaging to users across the globe. Zeus botnets used millions of zombie computers to execute keystroke logging and form grabbing attacks that targeted bank data, account logins,

and private user data. The information gathered by Zeus botnets has been used in thousands of cases of online identity theft,credit card theft, and more.

In October 2010, the FBI disclosed that it had detected an international cyber crime ring that had used Zeus botnets to steal over $70 million dollars from bank accounts in the United States. This spurred an FBI crackdown on the Zeus Trojan and Zeus botnets that led to the arrest of over 100 cyber-criminals.

In March 2012, Microsoft announced that they had taken over and shut down most of the control-and-command servers that were being used by Zeus botnets. According to Microsoft, all but three C&C domains had been taken down in the effort (formally referred to as Operation b71). While Microsoft wasn't able to eliminate every C&C server, their efforts are expected to slow or stop many of the cyber-criminals that were using Zeus botnets.[1]

Botnet Detection and Prevention

Botnet detection can be difficult, as bots are designed to operate without users' knowledge. However, there are some common signs that a computer may be infected with a botnet virus (listed below). While these symptoms are often indicative of bot infections, some can also be symptoms of malware infections or network issues and should not be taken as a sure sign that a computer is infected with a bot.

- IRC traffic (botnets and bot masters use IRC for communications)
- Connection attempts with known C&C servers
- Multiple machines on a network making identical DNS requests
- High outgoing SMTP traffic (as a result of sending spam)
- Unexpected popups (as a result of clickfraud activity)
- Slow computing/high CPU usage
- Spikes in traffic, especially Port 6667 (used for IRC), Port 25 (used in email spamming), and Port 1080 (used by proxy servers)
- Outbound messages (email, social media, instant messages, etc) that weren't sent by the user
- Problems with Internet access

There are several measures that users can take to prevent botnet virus infection. Since bot infections usually spread via malware, many of these measures actually focus on preventing malware infections. Recommended practices for botnet prevention include:

- Network baselining: Network performance and activity should be monitored so thatirregular network behavior is apparent.
- Software patches: All software should be kept up-to-date with security patches.

- Vigilance: Users should be trained to refrain from activity that puts them at risk of bot infections or other malware. This includes opening emails or messages, downloading attachments, or clicking links from untrusted or unfamiliar sources.

- Anti-Botnet tools: Anti-botnet tools provide botnet detection to augment preventative efforts by finding and blocking bot viruses before infection occurs. Most programs also offer features such as scanning for bot infections and botnet removal as well. Firewalls and antivirus software typically include basic tools for botnet detection, prevention, and removal. Tools like Network Intrusion Detection Systems (NIDS), rootkit detection packages, network sniffers, and specialized anti-bot programs can be used to provide more sophisticated botnet detection/prevention/removal.

How to Protect Internal Websites from Security Threats

Since internal websites usually keep sensitive employee and client information, intranet security is a high priority, especially for regulated industries like banking or healthcare that may get hefty fines because of security vulnerabilities.
We've been developing secure intranets based on SharePoint for 12 years and we'd like to sharewith you the most common threats and best practices to handle them in this article.

Intranet security risks and ways to handle them

Intranet security risks are divided into two groups: internal vulnerabilities and external

threats.To internal vulnerabilities belong:

- Weak passwords. 80% of hacking-related breaches are tied to weak, compromised orreused passwords.

Your IT team should introduce the policy of creating strong passwords and the necessity to resetthem regularly, for example, after every 60 days of use. Also, admins should enable automatic
logoff after a certain period of user inactivity and prevent logins from saving on computers and mobile devices.

- Non-restricted access. If any user is able to view any information on the intranet, including sensitive data, it can often lead to information leaks.

Your IT specialists need to configure role-based permissions that determine who can view, edit, or share certain files. Also, they can enable two-factor authentication. What's more, an audit trail can be used to track all content-related user activities like uploading and modification.

- Unprotected data. While not encrypted, intranet data may be susceptible to security breaches.

Your admin should encrypt intranet data at rest and in transit. For example, in SharePoint intranets, BitLocker offers two-level encryption of data at rest: it encrypts all data on a disk and provides a unique key for each file. And data in transit is protected due to SSL/TLS connection.

- Unsecured remote access. Employees can enter cloud intranets remotely via their mobile devices. These devices usually don't have reliable antiviruses or firewalls capable of protecting corporate information within public 3G, 4G or Wi-Fi networks.

Your IT professionals should arrange employee training on the importance of securing their mobile devices, especially if they use them to reach the corporate intranet. What's more, the IT team should have remote access to work-related data on these devices. It will help them to monitor users' activities, collect log info and implement remote wipe of sensitive data in case thedevices are lost or stolen. As a result, a device is restored to default configuration with all intranet-related data, apps and settings removed.

To external threats belong:

- Malware. Viruses, ransomware, and spyware can attack an intranet and seriously affectits performance, for example, cause slow operating and technical errors.

The IT team should regularly conduct intranet event monitoring, which helps to detect such issues as an unusual activity or an uncommonly large data inflow. Since the malware threat is constantly changing, it's crucial to timely update anti-malware tools.

- Social engineering (phishing). Phishing attacks using tools integrated with an intranet (e.g. email, chat) can lure employees to disclose sensitive information like customers'
  contacts or account numbers, which can damage the reputation of the company concerned.

Your IT team needs to install anti-phishing software. Also, the team should encourage users tobe hyper-aware of emails asking for personal and financial information, inspect URLs and links for odd characters or misspelled words, and more.

- Distributed Denial of Service (DDoS) attacks. These attacks are aimed at overwhelmingan intranet with data requests, which makes it inoperable.

To protect against DDoS attacks, your IT professionals should utilize such tools as firewalls, andload balancers to control the volume of traffic reaching your intranet.

Protect your intranet

Using a variety of technical tools and preventive measures like security analytics taken by your IT team is half of the battle for a secure internal website. The other half rests with intranet users who should undergo security and compliance training and follow corporate security policies.

What is a phishing attack?

“Phishing” refers to an attempt to steal sensitive information, typically in the form of usernames, passwords, credit card numbers, bank account information or other important data in order to utilize or sell the stolen information. By masquerading as a reputable source with an enticing request, an attacker lures in the victim in order to trick them, similarly to how a fisherman uses bait to catch a fish.

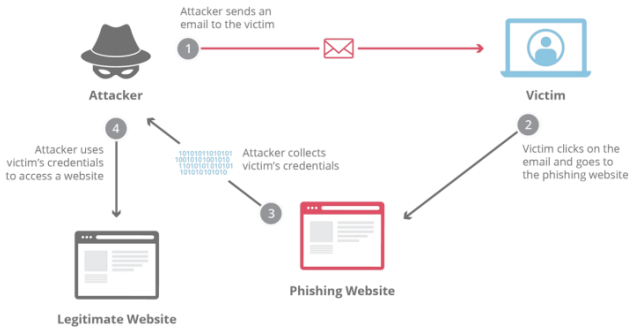


Fig 8. Phishing Attack

How is phishing carried out?

The most common examples of phishing are used to support other malicious actions, suchas man-in-the-middle and cross-site scripting attacks. These attacks typically occur via email or instant message, and can be broken down into a few general categories. It’s useful to become familiar with a few of these different vectors of phishing attacks in order to spot them in the wild.

Advanced-fee scam

This common email phishing attack is popularized by the “Nigerian prince” email, where an alleged Nigerian prince in a desperate situation offers to give the victim a large sum of moneyfor a small fee upfront. Unsurprisingly, when the fee is paid, no large sum of money ever arrives. The interesting history is that this type of scam has been occurring for over a hundred years in different forms; it was originally known in the late 1800s as the Spanish Prisoner scam, in which a con artist contacted a victim to pray on their greed and sympathy. The con artist is allegedly trying to smuggle out a wealthy Spanish prisoner, who will reward the victim handsomely in exchange for the money to bribe some prison guards.

This attack (is all its forms) is mitigated by not responding to requests from unknown parties in which money has to be given to receive something in return. If it sounds too good to be true, it
probably is. A simple Google search on the theme of the request or some of the text itself will often bring up the details of the scam.

Account deactivation scam

By playing off the urgency created in a victim who believes an important account is going to be deactivated, attackers are able to trick some people into handing over important information such as login credentials. Here's an example: the attacker sends an email that appears to come from animportant institution like a bank, and they claim the victim's bank account will be deactivated if they do not take action quickly. The attacker will then request the login and password to the victim's bank account in order to prevent the deactivation. In a clever version of the attack, once the information is entered, the victim will be directed to the legitimate bank website so that nothing looks out of place.

This type of attack can be countered by going directly to the website of the service in question and seeing if the legitimate provider notifies the user of the same urgent account status. It's also good to check the URL bar and make sure that the website is secure. Any website requesting a login and password that is not secure should be seriously questioned, and nearly without exception should not be used.

Website forgery scam

This type of scam is commonly paired with other scams such as the account deactivation scam.In this attack, the attacker creates a website that is virtually identical to the legitimate website ofa business the victim uses, such as a bank. When the user visits the page through whatevermeans, be it an email phishing attempt, a hyperlink inside a forum, or via a search engine, the victim reaches a website which they believe to be the legitimate site instead of a fraudulent copy. All information entered by the victim is collected for sale or other malicious use.

In the early days of the Internet, these types of duplicate pages were fairly easy to spot dueto their shoddy craftsmanship. Today the fraudulent sites may look like a picture-perfect representation of the original. By checking the URL in the web browser, it is usually pretty easy to spot a fraud. If the URL looks different than the typical one, this should be considered highly suspect. If the pages listed as insecure and HTTPS is not on, this is a red flag and virtually guarantees the site is either broken or a phishing attack.

Securing Wireless Networks

In today's connected world, almost everyone has at least one internet-connected device. With the number of these devices on the rise, it is important to implement a security strategy to minimize their potential for exploitation. Internet-connected devices
may be used by nefarious entities to collect personal information, steal identities, compromise financial data, and silently listen to—or watch—users. Taking a few precautions in the configuration and use of your devices can help prevent this type of activity.

What are the risks to your wireless network?

Whether it's a home or business network, the risks to an unsecured wireless network are the same. Some of the risks include:

Piggybacking

If you fail to secure your wireless network, anyone with a wireless-enabled computer in range of your access point can use your connection. The typical indoor broadcast range of an access point is 150–300 feet. Outdoors, this range may extend as far as 1,000 feet. So, if your neighborhood is closely settled, or if you live in an apartment or condominium, failure to

secure your wireless network could open your internet connection to many unintended users. These users may be ableto conduct illegal activity, monitor and capture your web traffic, or steal personal files.

Wardriving

Wardriving is a specific kind of piggybacking. The broadcast range of a wireless access point can make internet connections available outside your home, even as far away as your street. Savvy computer users know this, and some have made a hobby out of driving through cities and neighborhoods with a wireless-equipped computer—sometimes with a powerful antenna— searching for unsecured wireless networks. This practice is known as "wardriving."

Evil Twin Attacks

In an evil twin attack, an adversary gathers information about a public network access point, thensets up their system to impersonate it. The adversary uses a broadcast signal stronger than theone generated by the legitimate access point; then, unsuspecting users connect using the stronger signal. Because the victim is connecting to the internet through the attacker's system, it's easy for the attacker to use specialized tools to read any data the victim sends over the internet. This data may include credit card numbers, username and password combinations, and other personal information. Always confirm the name and password of a public Wi-Fi hotspot prior to use. This will ensure you are connecting to a trusted access point.

Wireless Sniffing

Many public access points are not secured and the traffic they carry is not encrypted. This can put your sensitive communications or transactions at risk. Because your connection is being transmitted "in the clear," malicious actors could use sniffing tools to obtain sensitive information such as passwords or credit card numbers. Ensure that all the access points you connect to use at least WPA2 encryption.

Unauthorized Computer Access

An unsecured public wireless network combined with unsecured file sharing could allow a malicious user to access any directories and files you have unintentionally made available for sharing. Ensure that when you connect your devices to public networks, you deny sharing files and folders. Only allow sharing on recognized home networks and only while it is necessary to share items. When not needed, ensure that file sharing is disabled. This will help prevent an unknown attacker from accessing your device's files.

Shoulder Surfing

In public areas malicious actors can simply glance over your shoulder as you type. By simply watching you, they can steal sensitive or personal information. Screen protectors that prevent shoulder-surfers from seeing your device screen can be purchased for little money. For smaller devices, such as phones, be cognizant of your surroundings while viewing sensitive information or entering passwords.

Theft of Mobile Devices

Not all attackers rely on gaining access to your data via wireless means. By physically stealing your device, attackers could have unrestricted access to all of its data, as well as any connected cloud accounts. Taking measures to protect your devices from loss or theft is important, but should the worst happen, a little preparation may protect the data inside. Most mobile devices, including laptop computers, now have the ability to fully encrypt their stored

data—making devices useless to attackers who cannot provide the proper password or personal identification number (PIN). In addition to encrypting device content, it is also advisable to configure your device's applications to request login information before allowing access to any cloud-based information. Last, individually encrypt or password-protect files that contain personal or sensitive information. This will afford yet another layer of protection in the event an attacker is able to gain access to your device.

What can you do to minimize the risks to your wireless network?

- Change default passwords. Most network devices, including wireless access points, are pre-configured with default administrator passwords to simplify setup. These default passwords are easily available to obtain online, and so provide only marginal protection. Changing default passwords makes it harder for attackers to access a device. Use and periodic changing of complex passwords is your first line of defense in protecting your device.
- Restrict access. Only allow authorized users to access your network. Each piece of hardware connected to a network has a media access control (MAC) address. You can
  restrict access to your network by filtering these MAC addresses. Consult your user documentation for specific information about enabling these features. You can also utilize the "guest" account, which is a widely used feature on many wireless routers. This feature allows you to grant wireless access to guests on a separate wireless channel with a separate password, while maintaining the privacy of your primary credentials.

- Encrypt the data on your network. Encrypting your wireless data prevents anyone who might be able to access your network from viewing it. There are several encryption protocols available to provide this protection. Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), WPA2, and WPA3 encrypt information being transmitted between wireless routers and wireless devices. WPA3 is currently the strongest encryption. WEP, WPA, and WPA2 are still available; however, it is advisable to use equipment that specifically supports WPA3, as using the other protocols could leave yournetwork open to exploitation.

- Protect your Service Set Identifier (SSID). To prevent outsiders from easily accessing your network, avoid publicizing your SSID. All Wi-Fi routers allow users to protect their device's SSID, which makes it more difficult for attackers to find a network. At the very least, change your SSID to something unique. Leaving it as the manufacturer's default could allow a potential attacker to identify the type of router and possibly exploit any known vulnerabilities.

- Install a firewall. Consider installing a firewall directly on your wireless devices (a host- based firewall), as well as on your home network (a router- or modem-based firewall). Attackers who can directly tap into your wireless network may be able to circumvent your network firewall—a host-based firewall will add a layer of protection to the data on your computer.

- Maintain antivirus software. Install antivirus software and keep your virus definitions up to date. Many antivirus programs also have additional features that detect or protect against spyware and adware.

- Use file sharing with caution. File sharing between devices should be disabled when not needed. You should always choose to only allow file sharing over home or work networks, never on public networks. You may want to consider creating a dedicated directory for file sharing and restrict access to all other directories. In addition, you

should password protect anything you share. Never open an entire hard drive for file sharing.

- Keep your access point software patched and up to date. The manufacturer of your wireless access point will periodically release updates to and patches for a device's
software and firmware. Be sure to check the manufacturer's website regularly for anyupdates or patches for your device.

- Check your internet provider's or router manufacturer's wireless security options. Your internet service provider and router manufacturer may provide information or resources to assist in securing your wireless network. Check the customer support area of their websites for specific suggestions or instructions.

- Connect using a Virtual Private Network (VPN). Many companies and organizations have a VPN. VPNs allow employees to connect securely to their network when away from the office. VPNs encrypt connections at the sending and receiving ends and keep out traffic that is not properly encrypted. If a VPN is available to you, make sure you log onto it any time you need to use a public wireless access point.

Question Bank:

1. Categorize the malwares based on its functionalities.
2. Compare bootstrapping in wireless ad hoc networks and wireless sensor networks
3. Assess the signal flow in call delivery services of cellular networks.
4. Determine the security threats imposed by phishing attacks and analyse how to overcome those threats?
5. How to analyse the data packets that represents potential or imminent threats to a local area network?
6. Summarize security mechanisms need to be applied to protect internal Websites of any firm.
7. Appraise the mechanism which prevents an online service is made unavailable by overwhelming it with traffic from multiple resources.
8. Elaborate the impact of network of compromised computers under the control of a malicious actor and discuss various mechanisms, which can handle this issue.
9. Discuss the protocols, which secures wireless networks.
10. Build the architecture of cellular networks and explain the steps involved in core networks design.
11. Construct a model of sample cascading attacks in the call delivery service.
12. Conclude various risks persists in wireless network environment
13. Explain 1. Replay attack 2. Denial of service attack
14. Discuss the categories of signature analysis on local area network
15. Elaborate categories of firewalls, which used to protect networks from intruders.

School of Computing
Department of Computer Science and Engineering

Unit III - System Security and Practices - SCS7008

Information security essentials for IT Managers- Security Management System – Policy Driven System Management- IT Security - Online Identity and User Management System - Intrusion and Detection and Prevention System.

Information security essentials for IT Managers

Information security involves the protection of organizational assets from the disruption of business operations, modification of sensitive data, or disclosure of proprietary information. The protection of this data is usually described as maintaining the confidentiality, integrity, and availability (CIA) of the organization's assets, operations, and information.

Information security management as a field is ever increasing in demand and responsibility because most organizations spend increasingly larger percentages of their IT budgets in attempting to manage risk and mitigate intrusions, not to mention the trend in many enterprises of moving all IT operations to an Internet-connected infrastructure, known as enterprise cloud computing. For information security managers, it is crucial to maintain a clear perspective of all the areas of business that require protection. Through collaboration with all business units, security managers must work security into the processes of all aspects of the organization, from employee training to research and development. Security is not an IT problem; it is a business problem.

Scope of Information Security Management

Information security is a business problem in the sense that the entire organization must frame and solve security problems based on its own strategic drivers, not solely on technical controls aimed to mitigate one type of attack. As identified throughout this chapter, security goes beyond technical controls and encompasses people, technology, policy, and operations in a way that few other business objectives do. The evolution of a risk-based paradigm, as opposed to a technical solution paradigm for security, has made it clear that a secure organization does not result from securing technical infrastructure alone. Furthermore, securing the organization's technical infrastructure cannot provide the appropriate protection for these assets, nor will it protect many other information assets that are in no way dependent on technology for their existence or protection. Thus, the organization would be lulled into a false sense of security if it relied on protecting its technical infrastructure alone.

CISSP 10 Domains of Information Security

In the information security industry there have been several initiatives to attempt to define security management and how and when to apply it. The leader in certifying information security professionals is the Internet Security Consortium, with its CISSP certification. In defining required skills for information security managers, the ISC has arrived at an agreement on 10 domains of information security that is known as the Common Body of Knowledge (CBK). Every security manager must understand and be well versed in all areas of the CBK.

In addition to individual certification there must be guidelines to turn these skills into actionable items that can be measured and verified according to some international standard or framework. The most widely used standard for maintaining and improving information security is ISO/IEC 17799:2005. ISO 17799 (see Figure 3.1) establishes guidelines and principles for initiating, implementing, maintaining, and improving information security management in an organization.
A new and popular framework to use in conjunction with the CISSP CBK and the ISO 17799 guidelines is ISMM. ISMM is a framework (see Figure 3.2) that describes a five-level evolutionary path of increasingly organized and systematically more mature security layers. It is proposed for the maturity assessment of information security management and the evaluation of the level of security awareness and practice at any organization, whether public or private. Furthermore, it helps us better understand where, and to what extent, the three main processes of security (prevention, detection, and recovery) are implemented and integrated.

ISMM helps us better understand the application of information security controls outlined in ISO 17799. Figure 1.3 shows a content matrix that defines the scope of applicability between various security controls mentioned in ISO 17799's 10 domains and the corresponding scope of applicability

on the ISMM Framework.

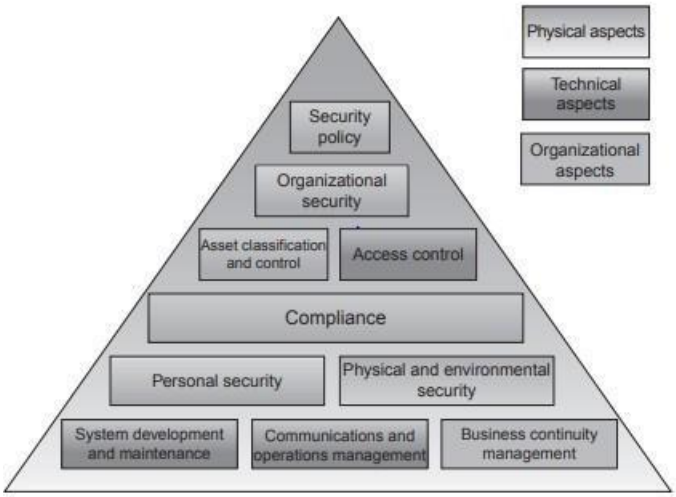


Figure 3.1: ISO 17799:2005 security model

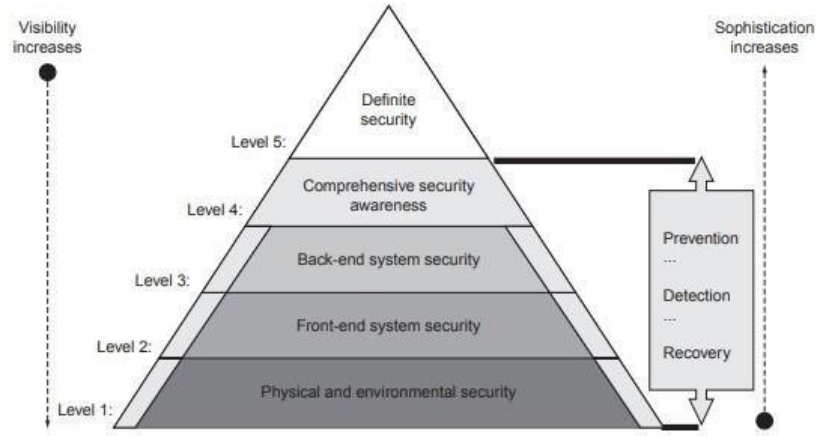


Figure 3.2: ISMM framework

CISSP 10 Domains: Common Body of Knowledge

- Access control. Methods used to enable administrators and managers to define what objects a subject can access through authentication and authorization, providing each subject a list of capabilities it can perform on each object. Important areas include access control security models, identification and authentication technologies, access control administration, and single sign-on technologies.

- Telecommunications and network security. Examination of internal, external, public, and private network communication systems, including devices, protocols, and remote access.

- Information security and risk management. Including physical, technical, and administrative controls surrounding organizational assets to determine the level of protection and budget warranted by highest to lowest risk. The goal is to reduce potential threats and money loss.

- Application security. Application security involves the controls placed within the application programs and operating systems to support the security policy of the organization and measure its effectiveness. Topics include threats, applications development, availability issues, security design and vulnerabilities, and application/data access control.

- Cryptography. The use of various methods and techniques such as symmetric and asymmetric encryption to achieve desired levels of confidentiality and integrity. Important areas include encryption protocols and applications and Public Key Infrastructures.

- Security architecture and design. This area covers the concepts, principles, and standards used to design and implement secure applications, operating systems, and all platforms based on international evaluation criteria such as Trusted Computer Security Evaluation Criteria (TCSEC) and Common Criteria.

- Operations security. Controls over personnel, hardware systems, and auditing and monitoring techniques such as maintenance of AV, training, auditing, and resource protection; preventive, detective, corrective, and recovery controls; and security and fault-tolerance technologies.

- Business continuity and disaster recovery planning. The main purpose of this area is to preserve business operations when faced with disruptions or disasters. Important aspects are to identify resource values; perform a business impact analysis; and produce business unit priorities, contingency plans, and crisis management.

- Legal, regulatory, compliance, and investigations. Computer crime, government laws and regulations, and geographic locations will determine the types of actions that constitute wrongdoing, what is suitable evidence, and what types of licensing and privacy laws your organization must abide by.

- Physical (environmental) security. Concerns itself with threats, risks, and countermeasures to protect facilities, hardware, data, media, and personnel. Main topics include restricted areas, authorization models, intrusion detection, fire detection, and security guards.

Security Management System

Information security management (ISM) describes controls that an organization needs to implement to ensure that it is sensibly protecting the confidentiality, availability, and integrity of assets from threats and vulnerabilities. By extension, ISM includes information risk management, a process which involves the assessment of the risks an organization must deal with in the management and protection of assets, as well as the dissemination of the risks to all appropriate stakeholders. This requires proper asset identification and valuation steps, including evaluating the value of confidentiality, integrity, availability, and replacement of assets. As part of information security management, an organization may implement an information security management system and other best practices found in the ISO/IEC 27001, ISO/IEC 27002, and ISO/IEC 27035 standards on information security.

Risk Management and Mitigation

Managing information security in essence means managing and mitigating the various threats and vulnerabilities to assets, while at the same time balancing the management effort expended on potential threats and vulnerabilities by gauging the probability of them actually occurring. A meteorite crashing into a server room is certainly a threat, for example, but an information security officer will likely put little effort into preparing for such a threat.

After appropriate asset identification and valuation has occurred, risk management and mitigation of those assets involves the analysis of the following issues

- Threats: Unwanted events that could cause the deliberate or accidental loss, damage, or misuse of information assets

- Vulnerabilities: How susceptible information assets and associated controls are to exploitation by one or more threats
- Impact and likelihood: The magnitude of potential damage to information assets from threats and vulnerabilities and how serious of a risk they pose to the assets; cost–benefit analysis may also be part of the impact assessment or separate from it
- Mitigation: The proposed method(s) for minimizing the impact and likelihood of potential threats and vulnerabilities

Once a threat and/or vulnerability has been identified and assessed as having sufficient impact/likelihood to information assets, a mitigation plan can be enacted. The mitigation method chosen largely depends on which of the seven Information Technology (IT) domains the threat and/or vulnerability resides in. The threat of user apathy toward security policies (the user domain) will require a much different mitigation plan than one used to limit the threat of unauthorized probing and scanning of a network (the LAN-to-WAN domain).

Information Security Management System

The information security management system (ISMS) represents the collation of all the interrelated/interacting information security elements of an organization so as to ensure policies, procedures, and objectives can be created, implemented, communicated, and evaluated to better guarantee an organization's overall information security. This system is typically influenced by organization's needs, objectives, security requirements, size, and processes. An ISMS includes and lends to effective risk management and mitigation strategies. Additionally, an organization's adoption of an ISMS largely indicates that it is systematically identifying, assessing, and managing information security risks and "will be capable of successfully addressing information confidentiality, integrity, and availability requirements." However, the human factors associated with ISMS development, implementation, and practice (the user domain) must also be considered to best ensure the ISMS' ultimate success.

Implementation and Education Strategy Components

Implementing effective information security management (including risk management and mitigation) requires a management strategy that takes note of the following:

- Upper-level management must strongly support information security initiatives, allowing information security officers the opportunity "to obtain the resources necessary to have a fully functional and effective education program" and, by extension, information security management system.

- Information security strategy and training must be integrated into and communicated through departmental strategies to ensure all personnel are positively affected by the organization's information security plan.

- A privacy training and awareness "risk assessment" can help an organization identify critical gaps in stakeholder knowledge and attitude towards security.

- Proper evaluation methods for "measuring the overall effectiveness of the training and awareness program" ensure policies, procedures, and training materials remain relevant.

- Policies and procedures that are appropriately developed, implemented, communicated, and enforced "mitigate risk and ensure not only risk reduction, but also ongoing compliance with applicable laws, regulations, standards, and policies."

- Milestones and timelines for all aspects of information security management help ensure future success.

Without sufficient budgetary considerations for all the above—in addition to the money allotted to standard regulatory, IT, privacy, and security issues—an information security management plan/system cannot fully succeed.

Relevant standards

Standards that are available to assist organizations with implementing the appropriate programs and controls to mitigate threats and vulnerabilities include the ISO/IEC 27000 family of standards, the ITIL framework, the COBIT framework, and O-ISM3 2.0. The ISO/IEC 27000 family represent some of the most well-known standards governing information security management and the ISMS and are based on global expert opinion. They lay out the requirements for best "establishing, implementing, deploying, monitoring, reviewing, maintaining, updating, and improving information security management systems. ITIL acts as a collection of concepts, policies, and best practices for the effective management of information technology infrastructure, service, and security, differing from ISO/IEC 27001 in only a few ways. COBIT, developed by ISACA, is a framework for helping information security personnel develop and implement strategies for information management and governance while minimizing negative impacts and controlling information security and risk management, and O-ISM3 2.0 is The Open Group's technology-neutral information security model for enterprise.

IT Security

As hackers get smarter, the need to protect your digital assets and network devices is even greater. While providing IT security can be expensive, a significant breach costs an organization far more. Large breaches can jeopardize the health of a small business. During or after an incident, IT security teams can follow an incident response plan as a risk management tool to gain control of the situation.

Difference between IT security and information security (InfoSec)

Although IT security and information security sound similar, they do refer to different types of security. Information security refers to the processes and tools designed to protect sensitive business information from invasion, whereas IT security refers to securing digital data, through computer network security.

Threats to IT security

Threats to IT security can come in different forms. A common threat is malware, or malicious software, which may come in different variations to infect network devices, including:

- Ransomware
- Spyware
- Viruses

These threats make it even more important to have reliable security practices in place. Learn more about malware to stay protected. IT security prevents malicious threats and potential security breaches that can have a huge impact on your organization. When you enter your internal company network, IT security helps ensure only authorized users can access and make changes to sensitive information that resides there. IT security works to ensure the

confidentiality of your organization's data.

Types of IT security

A. Network security

Network security is used to prevent unauthorized or malicious users from getting inside your network. This ensures that usability, reliability, and integrity are uncompromised. This type of security is necessary to prevent a hacker from accessing data inside the network. It also prevents them from negatively affecting your users' ability to access or use the network.

Network security has become increasingly challenging as businesses increase the number of endpoints and migrate services to public cloud.

B. Internet security

Internet security involves the protection of information that is sent and received in browsers, as well as network security involving web-based applications. These protections are designed to monitor incoming internet traffic for malware as well as unwanted traffic. This protection may come in the form of firewalls, antimalware, and antispyware.

C. Endpoint security

Endpoint security provides protection at the device level. Devices that may be secured by endpoint security include cell phones, tablets, laptops, and desktop computers. Endpoint security will prevent your devices from accessing malicious networks that may be a threat to your organization. Advance malware protection and device management software are examples of endpoint security.

D. Cloud security

Applications, data, and identities are moving to the cloud, meaning users are connecting directly to the Internet and are not protected by the traditional security stack. Cloud security can help secure the usage of software-as-a-service (SaaS) applications and the public cloud. A cloud-access security broker (CASB), secure Internet gateway (SIG), and cloud-based unified threat management (UTM) can be used for cloud security.

E. Application security

With application security, applications are specifically coded at the time of their creation to be as secure as possible, to help ensure they are not vulnerable to attacks. This added layer of security involves evaluating the code of an app and identifying the vulnerabilities that may exist within the software.

Online Identity and User Management System

IAM definition

Identity and access management (IAM) in enterprise IT is about defining and managing the roles and access privileges of individual network users and the circumstances in which users are granted (or denied) those privileges. Those users might be customers (customer

identity management) or employees (employee identity management. The core objective of IAM systems is one digital identity per individual. Once that digital identity has been established, it must be maintained, modified and monitored throughout each user's "access lifecycle."

Thus, the overarching goal of identity management is to "grant access to the right enterprise assets to the right users in the right context, from a user's system onboarding to permission authorizations to the offboarding of that user as needed in a timely fashion," according to Yassir Abousselham, senior vice president and chief security officer for Okta, an enterprise identity and access management provider.

IAM systems provide administrators with the tools and technologies to change a user's role, track user activities, create reports on those activities, and enforce policies on an ongoing basis. These systems are designed to provide a means of administering user access across an entire enterprise and to ensure compliance with corporate policies and government regulations.

IAM tools

Identity and management technologies include (but aren't limited to) password-management tools, provisioning software, security-policy enforcement applications, reporting and monitoring apps and identity repositories. Identity management systems are available for on-premises systems, such as Microsoft SharePoint, as well as for cloud-based systems, such as Microsoft Office 365.

API security enables IAM for use with B2B commerce, integration with the cloud, and microservices-based IAM architectures. Forrester sees API security solutions being used for single sign-on (SSO) between mobile applications or user-managed access. This would allow security teams to manage IoT device authorization and personally identifiable data.

Customer identity and access management (CIAM) allow "comprehensive management and authentication of users; self-service and profile management; and integration with CRM, ERP, and other customer management systems and databases," according to the report.

Identity analytics (IA) will allow security teams to detect and stop risky identity behaviors using rules, machine learning, and other statistical algorithms.

Identity as a service (IDaaS) includes "software-as-a-service (SaaS) solutions that offer SSO from a portal to web applications and native mobile applications as well as some level of user account provisioning and access request management," according to the report

Identity management and governance (IMG) provides automated and repeatable ways to govern the identity life cycle. This is important when it comes to compliance with identity and privacy regulations.

Risk-based authentication (RBA) solutions "take in the context of a user session and authentication and form a risk score. The firm can then prompt high-risk users for 2FA and allow low-risk users to authenticate with single factor (e.g., username plus password) credentials," according to the report. IAM systems must be flexible and robust enough to accommodate the complexities of today's computing environment. One reason: An enterprise's computing environment used to be largely on-premises, and identity

management systems authenticated and tracked users as they worked on-premises, says Jackson Shaw, vice president of product management for identity and access management provider One Identity. “There used to be a security fence around the premises,” Shaw noted. “Today, that fence isn’t there anymore.”

As a consequence, identity management systems today should enable administrators to easily manage access privileges for a variety of users, including domestic on-site employees and international off-site contractors; hybrid compute environments that encompass on-premise computing, software as a service (SaaS) applications and shadow IT and BYOD users; and computing architectures that include UNIX, Windows, Macintosh, iOS, Android and even internet of things (IoT) devices.

Ultimately, the identity and access management system should enable centralized management of users “in a consistent and scalable way across the enterprise,” says Abousselham. In recent years, identity-as-a-service (IDaaS) has evolved as a third-party managed service offered over the cloud on a subscription basis, providing identity management to a customers’ on-premises and cloud-based systems.

Identity and access management is a critical part of any enterprise security plan, as it is inextricably linked to the security and productivity of organizations in today’s digitally enabled economy. Compromised user credentials often serve as an entry point into an organization’s network and its information assets. Enterprises use identity management to safeguard their information assets against the rising threats of ransomware, criminal hacking, phishing and other malware attacks. In many organizations, users sometimes have more access privileges than necessary. A robust IAM system can add an important layer of protection by ensuring a consistent application of user access rules and policies across an organization.

Identity and access management systems can enhance business productivity. The systems’ central management capabilities can reduce the complexity and cost of safeguarding user credentials and access. At the same time, identity management systems enable workers to be more productive (while staying secure) in a variety of environments, whether they’re working from home, the office, or on the road.

IAM Means for Compliance Management

Many governments require enterprises to care about identity management. Regulations such as Sarbanes-Oxley, Gramm-Leach-Bliley, and HIPAA hold organizations accountable for controlling access to customer and employee information. Identity management systems can help organizations comply with those regulations.

The General Data Protection Regulation (GDPR) is a more recent regulation that requires strong security and user access controls. GDPR mandates that organizations safeguard the personal data and privacy of European Union citizens. Effective May 2018, the GDPR affects every company that does business in EU countries and/or has European citizens as customers.

On March 1, 2017, the state of New York’s Department of Financial Services (NYDFS) new cybersecurity regulations went into effect. The regulations prescribe many requirements for the security operations of financial services companies that operate in New York, including the need to monitor the activities of authorized users and maintain audit logs—something identity management systems typically do.

By automating many aspects of providing secure user access to enterprise networks and data, identity management systems relieve IT of mundane but important tasks and help them stay in compliance with government regulations. These are critical benefits, given that today, every IT position is a security position; there's a persistent, global cyber security workforce shortage; and penalties for not being compliant with relevant regulations can cost an organization millions or even billions of dollars.

Benefits of IAM Systems

Implementing identity and access management and associated best practices can give you a significant competitive advantage in several ways. Nowadays, most businesses need to give users outside the organization access to internal systems. Opening your network to customers, partners, suppliers, contractors and, of course, employees can increase efficiency and lower operating costs.

Identity management systems can allow a company to extend access to its information systems across a variety of on-premises applications, mobile apps, and SaaS tools without compromising security. By providing greater access to outsiders, you can drive collaboration throughout your organization, enhancing productivity, employee satisfaction, research and development, and, ultimately, revenue. Identity management can decrease the number of help-desk calls to IT support teams regarding password resets. Identity management systems allow administrators to automate these and other time-consuming, costly tasks.

An identity management system can be a cornerstone of a secure network, because managing user identity is an essential piece of the access-control picture. An identity management system all but requires companies to define their access policies, specifically outlining who has access to which data resources and under which conditions they have access. Consequently, well-managed identities mean greater control of user access, which translates into a reduced risk of internal and external breaches. This is important because, along with the rising threats of external threats, internal attacks are all too frequent. Approximately 60 percent of all data breaches are caused by an organization's own employees, according to IBM's 2016 Cyber Security Intelligence Index. Of those, 75 percent were malicious in intent; 25 percent were accidental.

As mentioned previously, IAM system can bolster regulatory compliance by providing the tools to implement comprehensive security, audit and access policies. Many systems now provide features designed to ensure that an organization is in compliance.

IAM Working Principal

In years past, a typical identity management system comprised four basic elements: a directory of the personal data the system uses to define individual users (think of it as an identity repository); a set of tools for adding, modifying and deleting that data (related to access lifecycle management); a system that regulates user access (enforcement of security policies and access privileges); and an auditing and reporting system (to verify what's happening on your system).

Regulating user access has traditionally involved a number of authentication methods for verifying the identity of a user, including passwords, digital certificates, tokens and smart cards. Hardware tokens and credit-card-sized smart cards served as one component in two-factor authentication, which combines something you know (your password) with something you have (the token or the card) to verify your identity. A smart card carries an embedded integrated circuit chip that can be either a secure microcontroller or equivalent intelligence

with internal memory or a memory chip alone. Software tokens, which canexist on any device with storage capability, from a USB drive to a cell phone, emerged in 2005.

In today's complex compute environments, along with heightened security threats, a strong user name and password doesn't cut it anymore. Today, identity management systems often incorporate elements of biometrics, machine learning and artificial intelligence, and risk-based authentication. At the user level, recent user authentication methods are helping to better protect identities. For example, the popularity of Touch ID-enabled iPhones has familiarized many people with using their fingerprints as an authentication method. Newer Windows 10 computers offer fingerprint sensors or iris scanning for biometric user authentication. The next iPhone, due out later this year, is rumored to include iris scanning or facial recognition to authenticate users instead of fingerprint scanning.

Multi-factor Authentication

Some organizations are moving from two-factor to three-factor authentication, says Abousselham, combining something you know (your password), something you have (a smartphone), and something you are (facial recognition, iris scanning or fingerprint sensors). "When you go from two-factor to three, you have more assurance that you're dealing with the correct user," he says. At the administration level, today's identity management systems offer more advanced user auditing and reporting, thanks to technologies such as context-aware network access control and risk-based authentication (RBA).

Context-aware network access control is policy-based. It predetermines an event as well as its outcome based on various attributes, says Joe Diamond, Okta's director of products. For example, if an IP address isn't whitelisted, it may be blocked. Or if there isn't a certificate that indicates a device is managed, then context-aware network access control might step-up the authentication process. By comparison, RBA is more dynamic and is often enabled by some level of AI. With RBA, "you're starting to open up risk scoring and machine learning to an authentication event," Diamond says.

Risk-based authentication dynamically applies various levels of strictness to authentication processes according to the current risk profile. The higher the risk, the more restrictive the authentication process becomes for a user. A change in a user's geographic location or IP address may trigger additional authentication requirements before that user can access the company's information resources.

Federated Identity Management

Federated identity management lets you share digital IDs with trusted partners. It's an authentication-sharing mechanism that allows users to employ the same user name, password or other ID to gain access to more than one network

Single sign-on (SSO) is an important part of federated ID management. A single sign-on standard lets people who verify their identity on one network, website or app carry over that authenticated status when moving to another. The model works only among cooperating organizations—known as trusted partners—that essentially vouch for each other's users.

IAM platforms based on open standards

Authorization messages between trusted partners are often sent using Security Assertion

Markup Language (SAML). This open specification defines an XML framework for exchanging security assertions among security authorities. SAML achieves interoperability across different vendor platforms that provide authentication and authorization services.

SAML isn't the only open-standard identity protocol, however. Others include OpenID, WS-Trust (short for Web Services Trust) and WS-Federation (which have corporate backing from Microsoft and IBM), and OAuth (pronounced "Oh-Auth"), which lets a user's account information be used by third-party services such as Facebook without exposing the password.

The challenges of implementing IAM

Dimensional Research released a report, Assessment of Identity and Access Management in 2018, in October 2018 based on a survey of more than 1,000 IT security professionals. Sponsored by IAM solution provider One Identity, the report asked those professionals about their biggest IAM challenges.Not surprisingly, 59 percent said that data protection was their biggest concern about their organization using IAM. Only 15 percent said they were completely confident their organization would not be hacked due to their access control system.

IAM systems hold the keys to some of a company's most valuable assets and critical systems, so the consequences of an IAM system failing are great. Specific concerns include disgruntled employees sharing sensitive data (27 percent), the CIO is interviewed on TV because of a data breach due to bad IAM, and finding their username/password lists posted to the dark web.

"The concept of putting all your eggs in one basket is scary," says One Identity's Shaw, "but if you don't unify the fundamentals of IAM you will never reduce risk. So the correct path is to arrive at a single approach (not necessarily a single solution) that provides all the scope, security and oversight you need (and were probably struggling to get with older projects) across everything, all user types, and all access scenarios."

Security professionals are also concerned about integrating IAM with legacy systems (50 percent), moving to the cloud (44 percent), and employees using unapproved technology (43 percent).

Much of that concern stems not from the current IAM technology itself, but with their organization's ability to implement it well, believes Shaw. "People have always been doing IAM (i.e., authentication, authorization and administration). It's just that now they are beginning to realize that doing those things poorly puts them at heightened risk and leaves the door open to bad actors doing bad things," he says.

"The biggest challenge is that old practices that were put in place to secure legacy systems simply don't work with newer technologies and practices," Shaw adds, "so often people have to reinvent the wheel and create duplicate workloads and redundant tasks. If the legacy practice was done poorly, trying to reinvent it on a newer paradigm will go poorly as well."

Shaw sees confidence and trust in IAM growing as companies gain experience administering the solutions, but that depends on how well that administration is executed. "Organizations are more-and-more learning that they can actually unify their

administration approach, streamline operations, remove much of the workload from IT and place it in the hands of the line-of-business, and place themselves in an audit-ready stance rather than a reactive stance," he says.

A successful implementation of identity and access management requires forethought and collaboration across departments. Companies that establish a cohesive identity management strategy—clear objectives, stakeholder buy-in, defined business processes—before they begin the project are likely to be most successful. Identity management works best "when you have human resources, IT, security and other departments involved," says Shaw.

Often, identity information may come from multiple repositories, such as Microsoft Active Directory (AD) or human resources applications. An identity management system must be able to synchronize the user identity information across all these systems, providing a single source of truth.

Given the shortage of IT people today, identity and access management systems must enable an organization to manage a variety of users in different situations and computing environments—automatically and in real-time. Manually adjusting access privileges and controls for hundreds or thousands of users isn't feasible.

For example, de-provisioning access privileges for departing employees can fall through the cracks, especially when done manually, which is too often the case. Reporting an employee's departure from the company and then automatically de-provisioning access across all the apps, services and hardware he or she used requires an automated, comprehensive identity management solution.

Authentication must also be easy for users to perform, it must be easy for IT to deploy, and above all it must be secure, Abousselham says. This accounts for why mobile devices are "becoming the center of user authentication," he added, "because smartphones can provide a user's current geolocation, IP address and other information that can be leveraged for authentication purposes."

One risk worth keeping in mind: Centralized operations present tempting targets to hackers and crackers. By putting a dashboard over all of a company's identity management activities, these systems reduce complexity for more than the administrators. Once compromised, they could allow an intruder to create IDs with extensive privileges and access to many resources.

IAM Terms

Buzzwords come and go, but a few key terms in the identity management space are worth knowing:

- **Access management:** Access management refers to the processes and technologies used to control and monitor network access. Access management features, such as authentication, authorization, trust and security auditing, are part and parcel of the top ID management systems for both on-premises and cloud-based systems.

- **Active Directory (AD):** Microsoft developed AD as a user-identity directory service for Windows domain networks. Though proprietary, AD is included in the Windows Server operating system and is thus widely deployed.

• **Biometric authentication:** A security process for authenticating users that relies upon the user's unique characteristics. Biometric authentication technologies include fingerprint sensors, iris and retina scanning, and facial recognition.

• **Context-aware network access control:** Context-aware network access control is a policy-based method of granting access to network resources according to the current context of the user seeking access. For example, a user attempting to authenticate from an IP address that hasn't been whitelisted would be blocked.

• **Credential:** An identifier employed by the user to gain access to a network such as the user's password, public key infrastructure (PKI) certificate, or biometric information (fingerprint, iris scan).

• **De-provisioning:** The process of removing an identity from an ID repository and terminating access privileges.

• **Digital identity:** The ID itself, including the description of the user and his/her/its access privileges. ("Its" because an endpoint, such as a laptop or smartphone, can have its own digital identity.)

• **Entitlement:** The set of attributes that specify the access rights and privileges of an authenticated security principal.

• **Identity as a Service (IDaaS):** Cloud-based IDaaS offers identity and access management functionality to an organization's systems that reside on-premises and/or in the cloud.

• **Identity lifecycle management:** Similar to access lifecycle management, the term refers to the entire set of processes and technologies for maintaining and updating digital identities. Identity lifecycle management includes identity synchronization, provisioning, de-provisioning, and the ongoing management of user attributes, credentials and entitlements.

• **Identity synchronization:** The process of ensuring that multiple identity stores—say, the result of an acquisition—contain consistent data for a given digital ID.

• **Lightweight Directory Access Protocol (LDAP):** LDAP is open standards-based protocol for managing and accessing a distributed directory service, such as Microsoft's AD

• **Multi-factor authentication (MFA):** MFA is when more than just a single factor, such as a user name and password, is required for authentication to a network or system. At least one additional step is also required, such as receiving a code sent via SMS to a smartphone, inserting a smart card or USB stick, or satisfying a biometric authentication requirement, such as a fingerprint scan.

• **Password reset:** In this context, it's a feature of an ID management system that allows users to re-establish their own passwords, relieving the administrators of the job and cutting support calls. The reset application is often accessed by the user through a browser. The application asks for a secret word or a set of questions to verify the user's identity.

• **Privileged account management**: This term refers to managing and auditing

accounts and data access based on the privileges of the user. In general terms, because of his or her job or function, a privileged user has been granted administrative access to systems. A privileged user, for example, would be able set up and delete user accounts and roles.**Provisioning:** The process of creating identities, defining their access privileges and adding them to an ID repository.

- **Risk-based authentication (RBA):** Risk-based authentication dynamically adjusts authentication requirements based on the user's situation at the moment authentication is attempted. For example, when users attempt to authenticate from a geographic location or IP address not previously associated with them, those users may face additional authentication requirements.

- **Security principal:** A digital identity with one or more credentials that can be authenticated and authorized to interact with the network.

- **Single sign-on (SSO):** A type of access control for multiple related but separate systems. With a single username and password, a user can access a system or systems without using different credentials.

- **User behavior analytics (UBA):** UBA technologies examine patterns of user behavior and automatically apply algorithms and analysis to detect important anomalies that may indicate potential security threats. UBA differs from other security technologies, which focus on tracking devices or security events. UBA is alsosometimes grouped with entity behavior analytics and known as UEBA.

Intrusion and Detection and Prevention System

Intrusion Detection System: An intrusion detection system (IDS) is a device, typically another separate computer, that monitors activity to identify malicious or suspicious events. An IDS is a sensor, like a smoke detector, that raises an alarm if specific things occur. A model of an IDS is shown in below figure. The components in the figure are the four basic elements of an intrusion detection system, based on the Common Intrusion Detection Framework of [STA96]. An IDS receives raw inputs from sensors. It saves those inputs, analyzes them, and takes some controlling action.

Types of IDSs

The two general types of intrusion detection systems are signature based and heuristic. Signature-based intrusion detection systems perform simple patternmatching and report situations that match a pattern corresponding to a known attack type. Heuristic intrusion detection systems, also known as anomaly based, build a model of acceptable behavior and flag exceptions to that model; for the future, the administrator can mark a flagged behavior as acceptable so that the heuristic IDS will now treat that previously unclassified behavior as acceptable. Intrusion detection devices can be network based or host based. A network-based IDS is a stand-alone device attached to the network to monitor traffic throughout that network; a host-based IDS runs on a single workstation or client or host, to protect that one host.

Signature-Based Intrusion Detection:

A simple signature for a known attack type might describe a series of TCP SYN packets sent to many different ports in succession and at times close to one another, as would be the case for a port scan. An intrusion detection system would probably find nothing unusual in the first SYN, say, to port 80, and then another (from the same source address) to port 25. But as more and more ports receive SYN packets, especially ports that are not

open, this pattern reflects a possible port scan. Similarly, some implementations of the protocol stack fail if they receive an ICMP packet with a data length of 65535 bytes, so such a packet would be a pattern for which to watch.

Heuristic Intrusion Detection:

Because signatures are limited to specific, known attack patterns, another form of intrusion detection becomes useful. Instead of looking for matches, heuristic intrusion detection looks for behavior that is out of the ordinary. The original work in this area focused on the individual, trying to find characteristics of that person that might be helpful in understanding normal and abnormal behavior. For example, one user might always start the day by reading e-mail, write many documents using a word processor, and occasionally back up files. These actions would be normal. This user does not seem to use many administrator utilities. If that person tried to access sensitive system management utilities, this new behavior might be a clue that someone else was acting under the user's identity. Inference engines work in two ways. Some, called state-based intrusion detection systems, see the system going through changes of overall state or configuration. They try to detect when the system has veered into unsafe modes. Others try to map current activity onto a model of unacceptable activity and raise an alarm when the activity resembles the model. These are called model-based intrusion detection systems. This approach has been extended to networks in [MUK94]. Later work sought to build a dynamic model of behavior, to accommodate variation and evolution in a person's actions over time. The technique compares real activity with a known representation of normality. Alternatively, intrusion detection can work from a model of known bad activity. For example, except for a few utilities (login, change password, create user), any other attempt to access a password file is suspect. This form of intrusion detection is known as misuse intrusion detection. In this work, the real activity is compared against a known suspicious area.

Stealth Mode:

An IDS is a network device (or, in the case of a host-based IDS, a program running on a network device). Any network device is potentially vulnerable to network attacks. How useful would an IDS be if it itself were deluged with a denial-of-service attack? If an attacker succeeded in logging in to a system within the protected network, wouldn't trying to disable the IDS be the next step? To counter those problems, most IDSs run in stealth mode, whereby an IDS has two network interfaces: one for the network (or network segment) being monitored and the other to generate alerts and perhaps other administrative needs. The IDS uses the monitored interface as input only; it never sends packets out through that interface. Often, the interface is configured so that the device has no published address through the monitored interface; that is, a router cannot route anything to that address directly, because the router does not know such a device exists. It is the perfect passive wiretap. If the IDS needs to generate an alert, it uses only the alarm interface on a completely separate control network.

Goals for Intrusion Detection Systems:

1. Responding to alarms

Whatever the type, an intrusion detection system raises an alarm when it finds a match. The alarm can range from something modest, such as writing a note in an audit log, to something significant, such as paging the system security administrator. Particular

implementations allow the user to determine what action the system should take on what events.

In general, responses fall into three major categories (any or all of which can be used in a single response):

Monitor, collect data, perhaps increase amount of data collected

Protect, act to reduce exposure

Call a human

2. False Results

Intrusion detection systems are not perfect, and mistakes are their biggest problem. Although an IDS might detect an intruder correctly most of the time, it may stumble in two different ways: by raising an alarm for something that is not really an attack (called a false positive, or type I error in the statistical community) or not raising an alarm for a real attack (a false negative, or type II error). Too many false positives means the administrator will be less confident of the IDS's warnings, perhaps leading to a real alarm's being ignored. But false negatives mean that real attacks are passing the IDS without action. We say that the degree of false positives and false negatives represents the sensitivity of the system. Most IDS implementations allow the administrator to tune the system's sensitivity, to strike an acceptable balance between false positives and negatives.

IDS Strength and Limitations

On the upside, IDSs detect an ever-growing number of serious problems. And as we learn more about problems, we can add their signatures to the IDS model. Thus, over time, IDSs continue to improve. At the same time, they are becoming cheaper and easier to administer. On the downside, avoiding an IDS is a first priority for successful attackers. An IDS that is not well defended is useless. Fortunately, stealth mode IDSs are difficult even to find on an internal network, let alone to compromise. IDSs look for known weaknesses, whether through patterns of known attacks or models of normal behavior. Similar IDSs may have identical vulnerabilities, and their selection criteria may miss similar attacks. Knowing how to evade a particular model of IDS is an important piece of intelligence passed within the attacker community. Of course, once manufacturers become aware of a shortcoming in their products, they try to fix it. Fortunately, commercial IDSs are pretty good at identifying attacks. Another IDS limitation is its sensitivity, which is difficult to measure and adjust. IDSs will never be perfect, so finding the proper balance is critical. In general, IDSs are excellent additions to a network's security. Firewalls block traffic to particular ports or addresses; they also constrain certain protocols to limit their impact. But by definition, firewalls have to allow some traffic to enter a protected area. Watching what that traffic actually does inside the protected area is an IDS's job, which it does quite well.

Question Bank:

1. Categorize IT security mechanisms, which satisfies the confidentiality of the organizations.
2. Assess the major goals of intrusion detection systems and discuss how those are achieved?
3. Discuss various threats to IT security by giving proper example to each.
4. Explain the forward looking specifications of Web security.
5. Inspect the necessity of aligning information technology with business drivers for security.
6. Discuss the security gateways used in Web Security.
7. Examine the content matrix defines the scope of applicability between various security mentioned in ISO.
8. Discuss the framework of policies and procedures that includes all legal, physical and technical controls involved in an organisation's information risk management processes.
9. Elaborate online identification and user management tools, which ensures access control.
10. Appraise the categories of Intrusion detection system based on the working principle.
11. Propose the key terms of user identification platforms that plays vital role in authentication.
12. Explain the Codifying Best Practices for a Secure Service Oriented Architecture of an organization.
13. Construct the ISO Security model incorporated by the IT firms
14. Discuss the Security Employee Training and Awareness (SETA) goals
15. Elaborate the actions to be taken when Implementing a Critical Security Mechanism
16. Analyze different types of IDPS systems used by organisations

School of Computing
Department of Computer Science and Engineering

UNIT – IV - System Security and Practices - SCS7008

UNIT 4 CYBER SECURITY AND CRYPTOGRAPHY

Cyber Forensics- Cyber Forensics and Incidence Response - Security e-Discovery - Network Forensics - Data Encryption Satellite Encryption - Password based authenticated Key establishment Protocols.

Cyber Forensics

Security monitoring involves real-time or near-real-time monitoring of events and activities happening on all your organization's important systems at all times. To properly monitor an organization for technical events that can lead to an incident or an investigation, usually an organization uses a security information and event management (SIEM) and/or log management tool. These tools are used by security analysts and managers to filter through tons of event data and to identify and focus on only the most interesting events.

Understanding the regulatory and forensic impact of event and alert data in any given enterprise takes planning and a thorough understanding of the quantity of data the system will be required to handle. The better logs can be stored, understood, and correlated, the better the possibility of detecting an incident in time for mitigation. In this case, what you don't know will hurt you. The need to respond to incidents, identify anomalous or unauthorized behavior, and secure intellectual property has never been more important. Without a solid log management strategy, it becomes nearly impossible to have the necessary data to perform a forensic investigation, and without monitoring tools, identifying threats and responding to attacks against confidentiality, integrity, or availability become much more difficult. For a network to be compliant and an incident response or forensics investigation to be successful, it is critical that a mechanism be in place to do the following:

- Securely acquire and store raw log data for as long as possible from as many disparate devices as possible while providing search and restore capabilities of these logs for analysis.

- Monitor interesting events coming from all important devices, systems, and applications in as near real time as possible.

Fig 4.1: Security monitoring

- Run regular vulnerability scans on your hosts and devices and correlate these vulnerabilities to intrusion detection alerts or other interesting events, identifying high-priority attacks as they happen and minimizing false positives.

SIEM and log management solutions in general can assist in security information monitoring (see Figure 4.1) as well as regulatory compliance and incident response by doing the

following:

- Aggregating and normalizing event data from unrelated network devices, security devices, and application servers into usable information.
- Analyzing and correlating information from various sources such as vulnerability scanners, IDS/IPS, firewalls, servers, and so on, to identify attacks as soon as possible and help respond to intrusions more quickly.

- Conducting network forensic analysis on historical or real-time events through visualization and replay of events.

- Creating customized reports for better visualization of your organizational security posture.

- Increasing the value and performance of existing security devices by providing a consolidated event management and analysis platform.

- Improving the effectiveness and helping focus IT risk management personnel on the events that are important.

- Meeting regulatory compliance and forensics requirements by securely storing all event data on a network for long-term retention and enabling instant accessibility to archived data.

Incidence Response and Forensic Investigations

Network forensic investigation is the investigation and analysis of all the packets and events generated on any given network in the hope of identifying the proverbial needle in a haystack. Tightly related is incident response, which entails acting in a timely manner to an identified anomaly or attack across the system. To be successful, both network investigations and incident response rely heavily on proper event and log management techniques. Before an incident can be responded to, there is the challenge of determining whether an event is a routine system event or an actual incident. This requires that there be some framework for incident classification (the process of examining a possible incident and determining whether or not it requires a reaction). Initial reports from end users, intrusion detection systems, host- and network-based malware detection software, and system administrators are all ways to track and detect incident candidates.

As mentioned in earlier sections, the phases of an incident usually unfold in the following order: preparation, identification (detection), containment, eradication, recovery, and lessons learned. The preparation phase requires detailed understanding of information systems and the threats they face; so to perform proper planning, an organization must develop predefined responses that guide users through the steps needed to properly respond to an incident. Predefining incident responses enables rapid reaction without confusion or wasted time and effort, which can be crucial for the success of an incident response. Identification occurs once an actual incident has been confirmed and properly classified as an incident that requires action. At that point the IR team moves from identification to containment. In the containment

phase, a number of action steps are taken by the IR team and others. These steps to respond to an incident must occur quickly and may occur concurrently, including notification of key personnel, the assignment of tasks, and documentation of the incident. Containment strategies focus on two tasks: first, stopping the incident from getting any worse, and second, recovering control of the system if it has been hijacked.

Once the incident has been contained and system control regained, eradication can begin, and the IR team must assess the full extent of damage to determine what must be done to restore the system. Immediate determination of the scope of the breach of confidentiality, integrity, and availability of information and information assets is called incident damage assessment.

Those who document the damage must be trained to collect and preserve evidence in case the incident is part of a crime investigation or results in legal action.

At the moment that the extent of the damage has been determined, the recovery process begins to identify and resolve vulnerabilities that allowed the incident to occur in the first place. The IR team must address the issues found and determine whether they need to install and/or replace or upgrade the safeguards that failed to stop or limit the incident or were missing from the system in the first place. Finally, a discussion of lessons learned should always be conducted to prevent future similar incidents from occurring and review what could have been done differently.

Validating Security Effectiveness

The process of validating security effectiveness comprises making sure that the security controls that you have put in place are working as expected and that they are truly mitigating the risks they claim to be mitigating. There is no way to be sure that your network is not vulnerable to something if you haven't validated it yourself. The only way to have a concrete means of validation is to ensure that the information security policy addresses your organizational needs and assess compliance with your security policy across all systems, assets, applications, and people.

Here are some areas where actual validation should be performed; in other words, these are areas where assigned IT personnel should go with policy in hand, log in, and verify the settings and reports before the auditors do:

- Verifying operating system settings
- Reviewing security device configuration and management
- Establishing ongoing security tasks
- Maintaining physical security
- Auditing security logs
- Creating an approved product list
- Reviewing encryption strength
- Providing documentation and change control

Vulnerability Assessments and Penetration Tests

Validating security with internal as well as external vulnerability assessments and penetration tests is a good way to measure an increase or decrease in overall security, especially if similar assessments are conducted on a regular basis. There are several ways to test security of applications, hosts, and network devices. With a vulnerability assessment, usually limited

scanning tools or just one scanning tool is used to determine vulnerabilities that exist in the target system. Then a report is created and the manager reviews a holistic picture of security. With authorized penetration tests, the process is a little different. In that case, the data owner is allowing someone to use just about any means within reason (in other words, many different tools and techniques) to gain access to the system or information. A successful penetration test does not provide the remediation avenues that a vulnerability assessment does; rather, it is a good test of how difficult it would be for someone to truly gain access if he were trying.

Security e-Discovery

In virtually all distributed environments, electronic mail is the most heavily used network-based application. Users expect to be able to, and do, send e-mail to others who are connected directly or indirectly to the Internet, regardless of host operating system or communications suite. With the explosively growing reliance on e-mail, there grows a demand for authentication and confidentiality services. Two schemes stand out as approaches that enjoy widespread use: Pretty Good Privacy (PGP) and S/MIME. Both are examined in this chapter. The chapter closes with a discussion of Domain Keys Identified Mail.

PRETTY GOOD PRIVACY

PGP is a remarkable phenomenon. Largely the effort of a single person, Phil Zimmermann, PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications. In essence, Zimmermann has done the following:

1. Selected the best available cryptographic algorithms as building blocks.

2. Integrated these algorithms into a general-purpose application that is independent of operatingsystem and processor and that is based on a small set of easy-to-use commands.

3. Made the package and its documentation, including the source code, freely available via the Internet, bulletin boards, and commercial networks such as AOL (America On Line).

4. Entered into an agreement with a company (Viacrypt, now Network Associates) to provide a fully compatible, low-cost commercial version of PGP. PGP has grown explosively and is now widely used.

The PGP documentation often uses the term secret key to refer to a key paired with a public key in a public-key encryption scheme. As was mentioned earlier, this practice risks confusion with a secret key used for symmetric encryption. Hence, we use the term private key instead.

Operational Description

The actual operation of PGP, as opposed to the management of keys, consists of four services: authentication, confidentiality, compression, and e-mail compatibility

AUTHENTICATION Figure illustrates the digital signature service provided by PGP.

The sequence is as follows.

1. The sender creates a message.

2. SHA-1 is used to generate a 160-bit hash code of the message.

3. The hash code is encrypted with RSA using the sender's private key, and the result is prepended to the message.

4. The receiver uses RSA with the sender's public key to decrypt and recover the hash code.

5. The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

CONFIDENTIALITY

Another basic service provided by PGP is confidentiality, which is provided by encrypting messages to be transmitted or to be stored locally as files. In both cases, the symmetric encryption algorithm CAST-128 may be used. Alternatively, IDEA or 3DES may be used.The 64-bit cipher feedback (CFB) mode is used. As always, one must address the problem of key distribution. In PGP, each symmetric key is used only once.That is, a new key is generated as a random 128- bit number for each message.Thus, although this is referred to in the documentation as a session key, it is in reality a one-time key. Because it is to be used only once, the session key is bound to the message and transmitted with it.To protect the key, it is encrypted with the receiver's public key. Figure illustrates the sequence, which can be described as follows.

5. The sender generates a message and a random 128-bit number to be used as a session key for this message only.

6. The message is encrypted using CAST-128 (or IDEA or 3DES) with the session key.

7. The session key is encrypted with RSA using the recipient's public key and is prepended to the message.

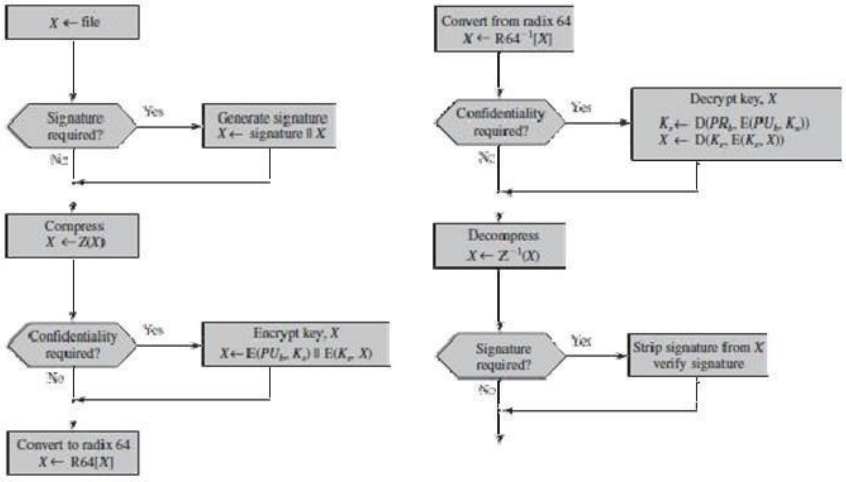


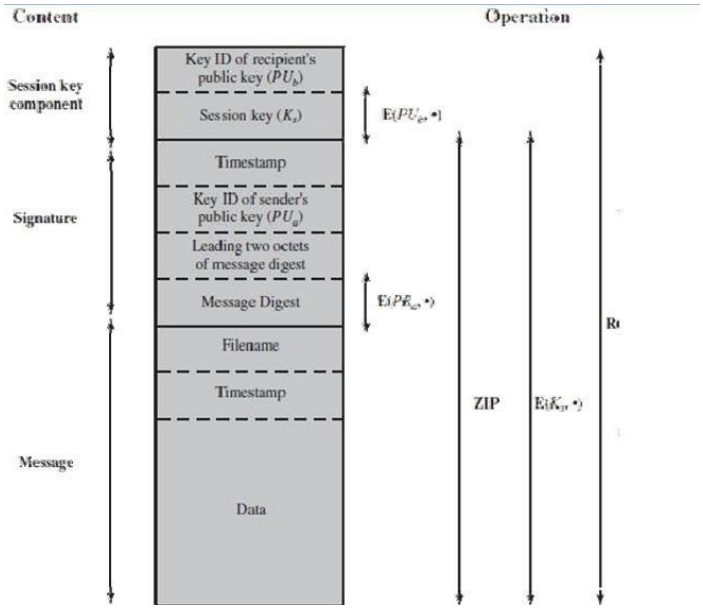
Figure 4.2 RSA Algorithm

8. The receiver uses RSA with its private key to decrypt and recover the session key.

9. The session key is used to decrypt the message.

The **message component** includes the actual data to be stored or transmitted, as well as a filename and a timestamp that specifies the time of creation. The **signature component** includes the following.

- **Timestamp:** The time at which the signature was made.
- **Message digest:** The 160-bit SHA-1 digest encrypted with the sender's private signature key. The digest is calculated over the signature timestamp concatenated with the data portion of the message component. The inclusion of the signature timestamp in the digest insures against replay types of attacks. The exclusion of the filename and timestamp portions of the message component ensures that detached signatures are exactly the same as attached signatures prefixed to the message. Detached signatures are calculated on a separate file that has none of the message component header fields.
- **Leading two octets of message digest:** Enables the recipient to determine if the correct public key was used to decrypt the message digest for authentication by comparing this plaintext copy of the first two octets with the first two octets of the decrypted digest. These octets also serve as a 16-bit frame check sequence for the message.
- **Key ID of sender's public key:** Identifies the public key that should be used to decrypt the message digest and, hence, identifies the private key that was used to encrypt the message digest.



S/MIME

Secure/Multipurpose Internet Mail Extension (S/MIME) is a security enhancement to the MIME Internet e-mail format standard based on technology from RSA Data Security. Although both PGP and S/MIME are on an IETF standards track, it appears likely that S/MIME will emerge as the industry standard for commercial and organizational use, while PGP will remain the choice for personal e-mail security for many users.

Multipurpose Internet Mail Extensions

Multipurpose Internet Mail Extension (MIME) is an extension to the RFC 5322 framework that is intended to address some of the problems and limitations of the use of Simple Mail Transfer Protocol (SMTP), defined in RFC 821, or some other mail transfer protocol and RFC 5322 for electronic mail.

1. SMTP cannot transmit executable files or other binary objects. A number of schemes are in use for converting binary files into a text form that can be used by SMTP mail systems, including thepopular UNIX UUencode/Uudecode scheme. However, none of these is a standard or even a de facto standard.

2. SMTP cannot transmit text data that includes national language characters, because these are represented by 8-bit codes with values of 128 decimal or higher, and SMTP is limited to 7-bit ASCII.

3. SMTP servers may reject mail message over a certain size.

4. SMTP gateways that translate between ASCII and the character code EBCDIC do not use a consistent set of mappings, resulting in translation problems.

5. SMTP gateways to X.400 electronic mail networks cannot handle nontextual data included in X.400 messages.

6. Some SMTP implementations do not adhere completely to the SMTP standards defined in RFC 821. Common problems include:

   - Deletion, addition, or reordering of carriage return and linefeed
   - Truncating or wrapping lines longer than 76 characters
   - Removal of trailing white space (tab and space characters)
   - Padding of lines in a message to the same length
   - Conversion of tab characters into multiple space characters

OVERVIEW The MIME specification includes the following elements.

1. Five new message header fields are defined, which may be included in an RFC 5322 header.These fields provide information about the body of the message.

2. A number of content formats are defined, thus standardizing representations that support multimedia electronic mail.

3. Transfer encodings are defined that enable the conversion of any content format into a form that is protected from alteration by the mail system.

In this subsection, we introduce the five message header fields. The next two subsections deal with content formats and transfer encodings.

The five header fields defined in MIME are

- MIME-Version: Must have the parameter value 1.0. This field indicates that the message conforms to RFCs 2045 and 2046.

- Content-Type: Describes the data contained in the body with sufficient detail that the receiving user agent can pick an appropriate agent or mechanism to represent the data to the user or otherwise deal with the data in an appropriate manner.

- Content-Transfer-Encoding: Indicates the type of transformation that has been used to represent the body of the message in a way that is acceptable for mail transport.

- Content-ID: Used to identify MIME entities uniquely in multiple contexts.

- Content-Description: A text description of the object with the body; this is useful when the object is not readable (e.g., audio data).

S/MIME Messages

S/MIME makes use of a number of new MIME content types. All of the new application types use the designation PKCS. This refers to a set of public-key cryptography specifications issued by RSA Laboratories and made available for the S/MIME effort.

We examine each of these in turn after first looking at the general procedures for S/MIME message preparation.

SECURING A MIME ENTITY S/MIME secures a MIME entity with a signature, encryption, or both. A MIME entity may be an entire message (except for the RFC 5322 headers), or if the MIME content type is multipart, then a MIME entity is one or more of the subparts of the message.The MIME entity is prepared according to the normal rules for MIME message preparation. Then the MIME entity plus some security-related data, such as algorithm identifiers and certificates, are processed by S/MIME to produce what is known as a PKCS object. A PKCS object is then treated as message content and wrapped in MIME (provided with appropriate MIME headers). This process should become clear as we look at specific objects and provide examples.

In all cases, the message to be sent is converted to canonical form. In particular, for a given type and subtype, the appropriate canonical form is used for the message content. For a multipart message, the appropriate canonical form is used for each subpart.

The use of transfer encoding requires special attention. For most cases, the result of applying the security algorithm will be to produce an object that is partially or totally represented in arbitrary binary data. This will then be wrapped in an outer MIME message, and transfer encoding can be applied at that point, typically base64. However, in the case of a multipart signed message (described in more detail later), the message content in one of the subparts is unchanged by the security process. Unless that content is 7bit, it should be transfer encoded using base 64 or quoted- printable so that there is no danger of altering the content to which the signature was applied.

Network Forensics

Network forensics is the idea of being able to resolve network problems through captured network traffic

- Previous methods focused on recreating the problem
- New technologies eliminate the time-consuming task of having to recreate the issue
- Allows IT professionals to go immediately to problem resolution mode

Internal and governmentally mandated compliancy

- Provides enforcement of acceptable use policies
- Helps fight industrial espionage
- Assists with Sarbanes Oxley compliance
- Provides pre-intrusion tracking and identification

Security

- Helps deliver a post-intrusion “paper-trail”

Network Troubleshooting

- Performs root-cause analysis
- Allows for historical problem identification

With internal compliancy, some of the most common issues are…

Acceptable Use

- Internal organizational policy that applies to use of all company systems, including e-mail and Internet access
- Challenge –organizations cannot adequately enforce these policies

Industrial espionage

- In today's competitive world, espionage is a continuous threat
- Challenge –W ith the advent of e-mail and IM, perpetrating acts of espionage has become far easier than ever before.

IT administrators can assist SOX (Sarbanes-Oxley) compliancy in a number of ways…

SOX requires documentation of information flowing to and from devices which store company information

Network forensics can be used to track all communication to and from any device or segment of interest (SOX ACT, section 302) SOX references the COSO (Committee of Sponsoring Organizations of the Treadway Commission), and their framework which helps businesses to assess and align their IT governance policies with SOX

- One frameworks focuses on network monitoring
- Network forensics can ensure real-time and continued network monitoring

Health Insurance Portability and Accountability Act HIPAA (Healthcare industry)

- Requires that patient data be protected from unauthorized access
- This means ensuring that the data is secure as it traverses the network
- Should a security breach happen, regulations provide for large fines of the organization UNLESS they can prove that no data was transferred
- Network forensics can record all transactions occurring over the wire and thus prove if data transfer took place

The Situation:

- At a large financial organization, an employee is being reviewed for possible termination by HR. Among the offenses the employee is accused of is browsing inappropriate websites on company equipment.
- IT has been tasked with researching these possible offenses. However, providing only domain names or URLs is not acceptable according to the HR policy. The offense has to have been documented in some way that will reflect the activity the employee perpetrated.

Data Encryption Satellite Encryption

Cryptography

The many schemes used for encryption constitute the area of study known as cryptography

Crypt analysis
Techniques used for deciphering a message without any knowledge of the enciphering details fall into the area of cryptanalysis. Cryptanalysis is what thelayperson calls "breaking the code."

Cryptology
The areas of cryptography and cryptanalysis together are called cryptology

Cipher
Encryption scheme is known as a cryptographic system or a cipher

Plain Text
This is the original intelligible message or data that is fed into the algorithm as input.

Cipher Text
This is the scrambled message produced as output. It depends on the plaintext and the secret key. For a given message, two different keys will produce two different cipher texts. The cipher text is an apparently random stream of data and, as it stands, is unintelligible.

Secret key
The secret key is also input to the encryption algorithm. The key is a value independent of the plaintext and of the algorithm. The algorithm will produce a different output depending on the specific key being used at the time. The exact substitutions and transformations performed by the algorithm depend on the key.

Encryption
The process of converting from plaintext to cipher text

Decryption
The process of restoring the plaintext from the cipher text

Enciphering Algorithm
The encryption algorithm performs various substitutions and transformations on the plaintext

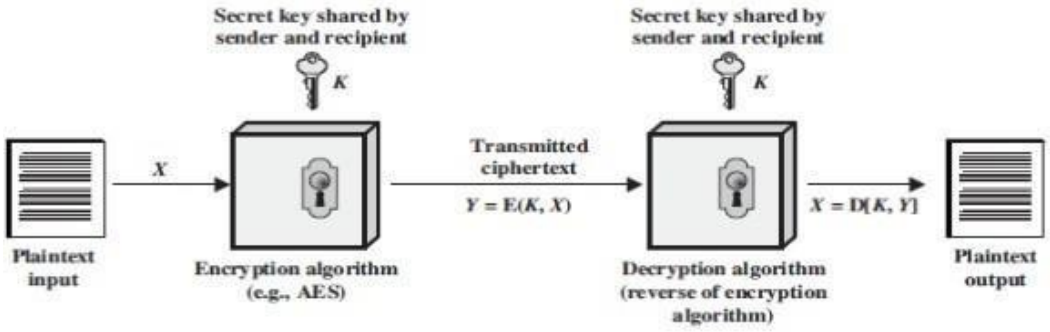


Fig. 1.1a Model of Symmetric Encryption

Deciphering Algorithm
This is essentially the encryption algorithm run in reverse. It takes the cipher text and the secret key and produces the original plaintext.

Threat
A potential for violation of security which exists when there is a circumstance,capability, action, or event, that could breach security and cause harm. That is, a threat is a possible danger that might exploit vulnerability.

Attack
An assault on system security that derives from an intelligent threat; that is, an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system.

Security attack: Any action that compromises the security of information owned by an organization.

Security mechanism: A process (or a device incorporating such a process) that is designed to detect, prevent, or recover from a security attack.

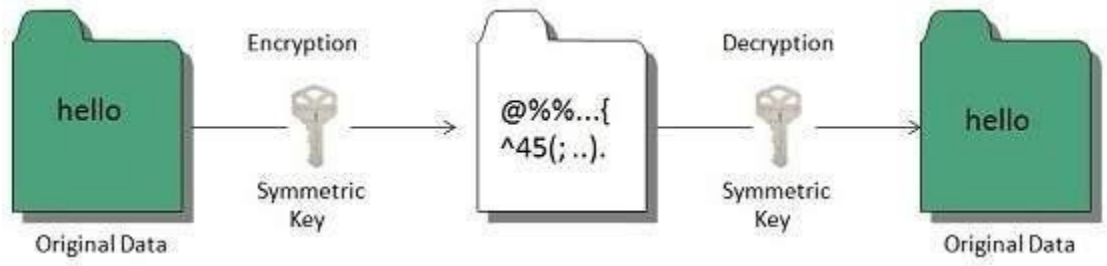
Security service: A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization. The services are intended to counter security attacks, and they make use of one or more security mechanisms to provide the service.

Types of Encryption

There are two types of encryptions schemes as listed below:

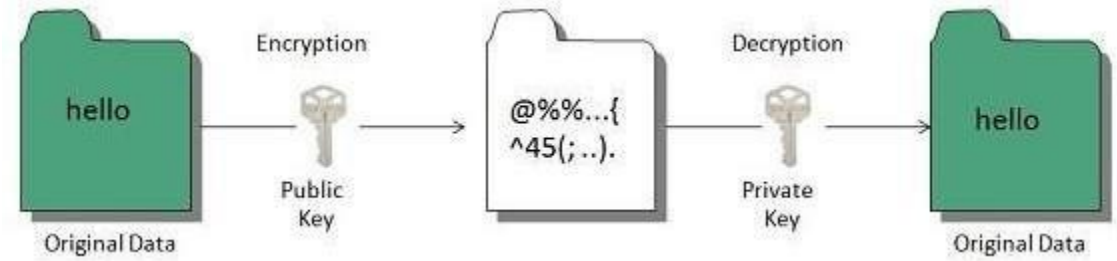
- Symmetric Key encryption
- Public Key encryption

Symmetric Key encryption



Public Key encryption

Public key encryption algorithm uses pair of keys, one of which is a secret key and one of which is public. These two keys are mathematically linked with each other.



Hashing

In terms of security, hashing is a technique used to encrypt data and generate unpredictable hash values. It is the hash function that generates the hash code, which helps to protect the security of transmission from unauthorized users.

Hash function algorithms

Hashing algorithm provides a way to verify that the message received is the same as the message sent. It can take a plain text message as input and then computes a value based on that message.

- The length of computed valueis much shorter than the original message.
- It is possible that different plain text messages could generate the same value.

Here we will discuss a sample hashing algorithm in which we will multiply the number of a's, e's and h's in the message and will then add the number of o's to this value.

For example, the message is " the combination to the safe is two, seven, thirty-five". The hash of this message, using our simple hashing algorithm is as follows:

$$(2 \times 6 \times 3) + 4 = 40$$

The hash of this message is sent to John with cipher text. After he decrypts the message, he computes its hash value using the agreed upon hashing algorithm. If the hash value sent by Bob doesn't match the hash value of decrypted message, John will know that the message has been altered.

For example, John received a hash value of 17 and decrypted a message Bob has sent as "You are being followed, use backroads, hurry"

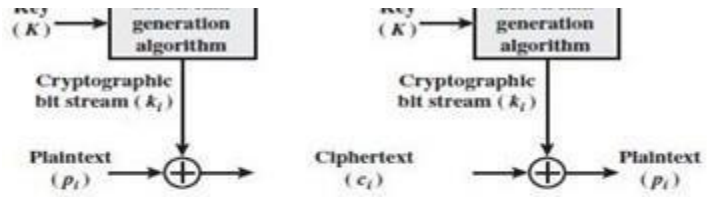
He could conclude the message had been altered, this is because the hash value of the message he received is:

$$(3 \times 4 \times 1) + 4 = 16$$

This is different from then value 17 that Bob sent.

Stream Ciphers and Block Ciphers

A **stream cipher** is one that encrypts a digital data stream one bit or one byte at a time. Examples of classical stream ciphers are the autokeyed Vigenère cipher and the Vernam cipher. Accordingly, for practical reasons, the bit-stream generator must be implemented as an algorithmic procedure, so that the cryptographic bit stream can be produced by both users.

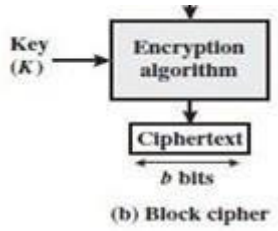


(a) Stream cipher using algorithmic bit-stream generator

In this approach, the bit-stream generator is a key-controlled algorithm and must produce a bit stream that is cryptographically strong. Now, the two users need only share the generating key, and each can produce the keystream.

A **block cipher** is one in which a block of plaintext is treated as a whole and used to produce a ciphertext block of equal length. Typically, a block size of 64 or 128 bits is used. As with a stream cipher, the two users share a symmetric encryption key.

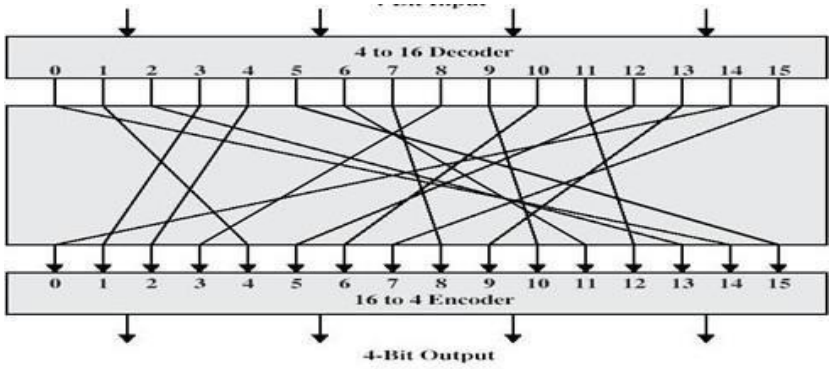
A block cipher can be used to achieve the same effect as a stream cipher. In general, they seem applicable to a broader range of applications than stream ciphers. The vast majorityof network-based symmetric cryptographic applications make use of block ciphers.



(b) Block cipher

Motivation for the Feistel Cipher Structure

Encryption should be reversible. Figure shows the logic of a general substitution cipher forn=4 (block size).



General n-bit-n-bit Block Substitution (shown with $n = 4$)

In general the logic of a general substitution cipher for n=4 with 4-bit input produces one of 16 possible input states, which is mapped by the substitution cipher into a unique one of 16 possible output states, each of which is represented by 4 cipher text bits. The encryption and decryption mappings can be defined by tabulation, as shown below.

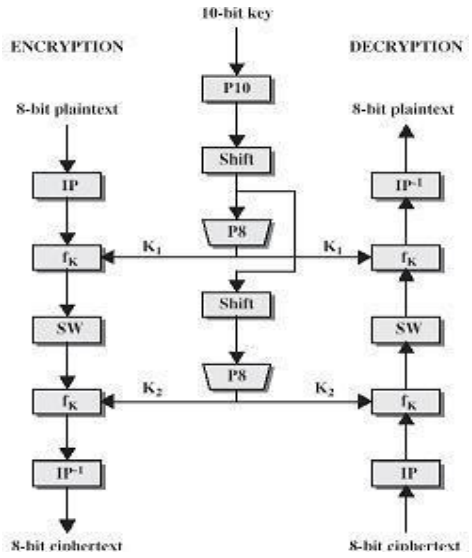
Encryption and Decryption Tables for Substitution

Plaintext	Ciphertext
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111

Ciphertext	Plaintext
0000	1110
0001	0011
0010	0100
0011	1000
0100	0001
0101	1100
0110	1010
0111	1111
1000	0111
1001	1101
1010	1001
1011	0110
1100	1011
1101	0010
1110	0000
1111	0101

SIMPLIFIED DES

S-DES encryption algorithm takes 8-bit block of plaintext and a 10-bit key, and produces 8-bit ciphertext block. Encryption algorithm involves 5 functions: an initial permutation (IP); a complex function f_K, which involves both permutation and substitution and depends ona key input; a simple permutation function that switches (SW) the 2 halves of the data; the function f_K again; and finally, a permutation function that is the inverse of the initial permutation (IP-1). Decryption process is similar.



The function fK takes 8-bit key which is obtained from the 10-bit initial key. The key is first subjected to a permutation P10. Then a shift operation is performed. The output of the shift operation then passes through a permutation function that produces an 8-bit output (P8) for the first subkey (K1). The output of the shift operation also feeds into another shift and another instance of P8 to produce the 2nd subkey K2. We can express encryption algorithm as superposition:

$$\text{Ciphertext} = IP^{-1}\left(f_{K_2}\left(SW\left(f\quad\left(IP(pla\,\text{int}\,ext)\right)\right)\right)\right)$$

Where,

$$K_1 = P8(Shift(P10(key)))$$

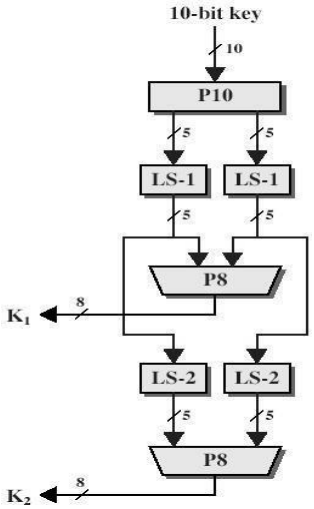
$$K_2 = P8(Shift(Shift(P10(key))))$$

Decryption is the reverse of encryption:

$$\text{Plaintext} = IP^{-1}\left(f_{K_1}\left(SW\left(f\quad\left(IP(ciphertext)\right)\right)\right)\right)$$

We now examine S-DES in more details

Scheme of key generation



First, permute the 10-bit key k1,k2, ... ,k10:

P10(k1,k2,k3,k4,k5,k6,k7,k8,k9,k10) = (k3,k5,k2,k7,k4,k10,k1,k9,k8,k6)

Or it may be represented in such a form:

P10

3	5	2	7	4	10	1	9	8	6

Each position in this table gives the identity of the input bit that produces the output bit in this position. So, the 1st output bit is bit 3 (k3), the 2nd is k5 and so on. For example, the key (1010000010) is permuted to (1000001100).

Next, perform a circular shift (LS-1), or rotation, separately on the 1st 5 bits and the 2nd 5 bits. In our example, the result is (00001 11000) Next, we apply P8, which picks out and permutes 8 out of 10 bits according to the following rule:

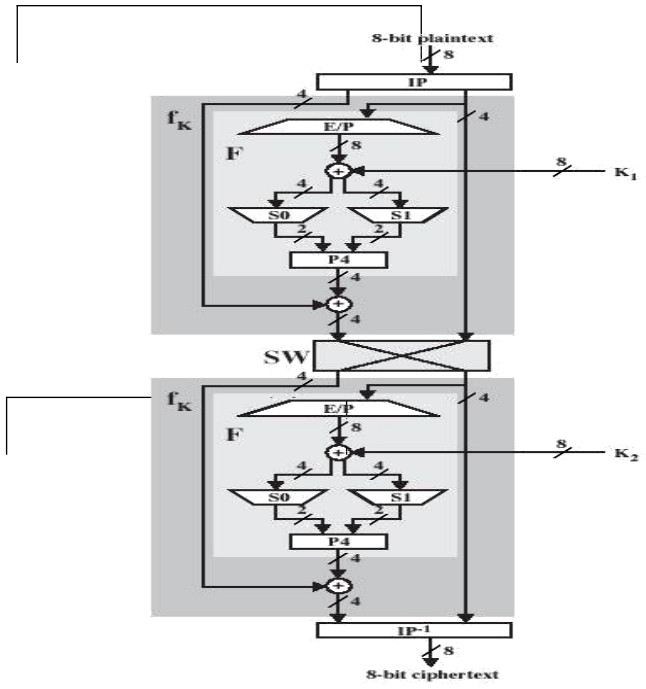
P8

6	3	7	4	8	5	10	9

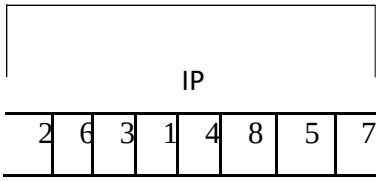
The result is subkey is K1. In our example, this yields (10100100)

We then go back to the pair of 5-bit strings produced by the 2 LS-1 functions and performa circular left shift of 2 bit positions on each string. In our example, the value (00001 11000) becomes (00100 00011). Finally, P8 is applied again to produce K2. In our example, the result is (01000011).

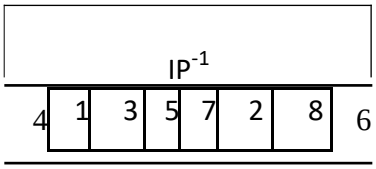
S-DES ENCRYPTION



The input to the algorithm is an 8-bit block of plaintext, which is permuted by IP function:



At the end of the algorithm, the inverse permutation is used:



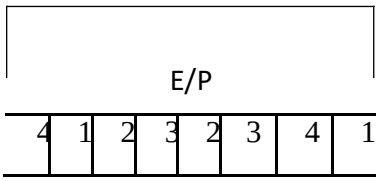
It may be verified, that $IP^{-1}(IP(X)) = X$.

The most complex component of S-DES is the function f_K, which consists of a combination of permutation and substitution functions. The function can be expressed as follows. Let L and R be the leftmost 4 bits and rightmost 4 bits of the 8-bit input to f_K, and let F be a mapping (not necessarily one to one) from 4-bit strings to 4-bit strings. Then we let

$$f_K(L,R) = (L \oplus F(R,SK),R)$$

where SK is a subkey and $\oplus$ is the bit-by-bit XOR operation. For example, suppose the output of the IP stage in Fig.3.3 is (1011 1101) and F(1101,SK) = (1110) for some key SK. Then f_K(1011 1101) = (0101 1101) because (1011) $\oplus$(1110) = (0101).

We now describe the mapping F. The input is a 4-bit number (n1 n2 n3 n4). The 1st operation is an expansion/permutation:



For what follows, it is clearer to depict result in this fashion:

n4|n1 n2|n3

n2|n3 n4|n1

The 8-bit subkey K1 = (k11, k12, k13, k14, k15, k16, k17, k18) is added to this value usingXOR:

n4+k11|n1+k12

n2+k13|n3+k14

n2+k15|n3+k16

n4+k17|n1+k18

Let us rename these bits:

p00|p01

p02|p03

p10|p11

p12|p13

The 1st 4 bits (1st row of the preceding matrix) are fed into the S-box S0 to produce a 2-bit output, and the remaining 4 bits (2nd row) are fed into S1 to produce another 2-bit output. These 2 boxes are defined as follows:

The S-boxes operate as follows. The 1st and 4th input bits are treated as a 2-bit number that specify a row of the S-box, and the 2nd and 3rd input bits specify a column of the S- box. The entry in that row and column, in base 2, is the 2-bit output. For example, if (p00, p03) = (00) and (p01, p02) = (10), then the output is from row 0, column 2 of S0, which is 3, or (11) in binary. Similarly, (p10, p13) and (p11, p12) are used to index into arow and column of S1 to produce an additional 2 bits.Next, the 4 bits produced by S0 and S1 undergo a further permutation as follows:

The

P4			
2	4	3	1

output of P4 is the output of function F.

The function f_K only alters the leftmost 4 bits of input. The switch function SW interchanges the left and right bits so that the 2nd instance of f_K operates on a different 4 bits. In the 2nd instance, the E/P, S0, S1, and P4 functions are the same. The key input is K2.

THE DATA ENCRYPTION STANDARD (DES)

It was adopted in 1977 by the National Bureau of Standards (NBS), now National Institute of Standards and Technology (NIST), as Federal Information Processing Standard 46 (FIPS PUB 46). In 1971, IBM's team under Horst Feistel leadership developed algorithm LUCIFER,operating on 64-bit blocks with 128-bit key. Further, IBM's team leaded by Walter Tuchman and Carl Meyer revised LUCIFER to make it more resistant to cryptanalysis, but they reduced key size to 56 bits. In 1973, NBS issued a request for proposals for a national cipher standard. IBM submitted results of its Tuchman-Meyer project.

This was by far the best algorithm proposed and was adopted in 1977 as Data Encryption Standard. In 1994, NIST reaffirmed DES for federal use for another 5 years. In 1999, NIST issued a new versionof its standard (FIPS PUB 46-3) that indicated that DES should only be used for legacy systems and that triple DES be used.

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).

DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only).

The 64 bit input enters into initial permutation and the permutated output is fed into sixteen rounds with kay values and then 32-bit swap swaps left and 32-bit halves obtainedafter Round 16, we get preoutput. Finally, preoutput passes through a permutation IP-1, that is inverse to initial permutation IP, to produce the 64-bit ciphertext. The right-hand portion of Fig. 3.7 shows the way in which 56-bit is used. For each of 16 rounds a subkey Ki

is produced by the combination of a left circular shift and a permutation. The permutation function is the same for each round**.**

INITIAL PERMUTATION AND ITS INVERSE

It effects on 64-bit input : IP

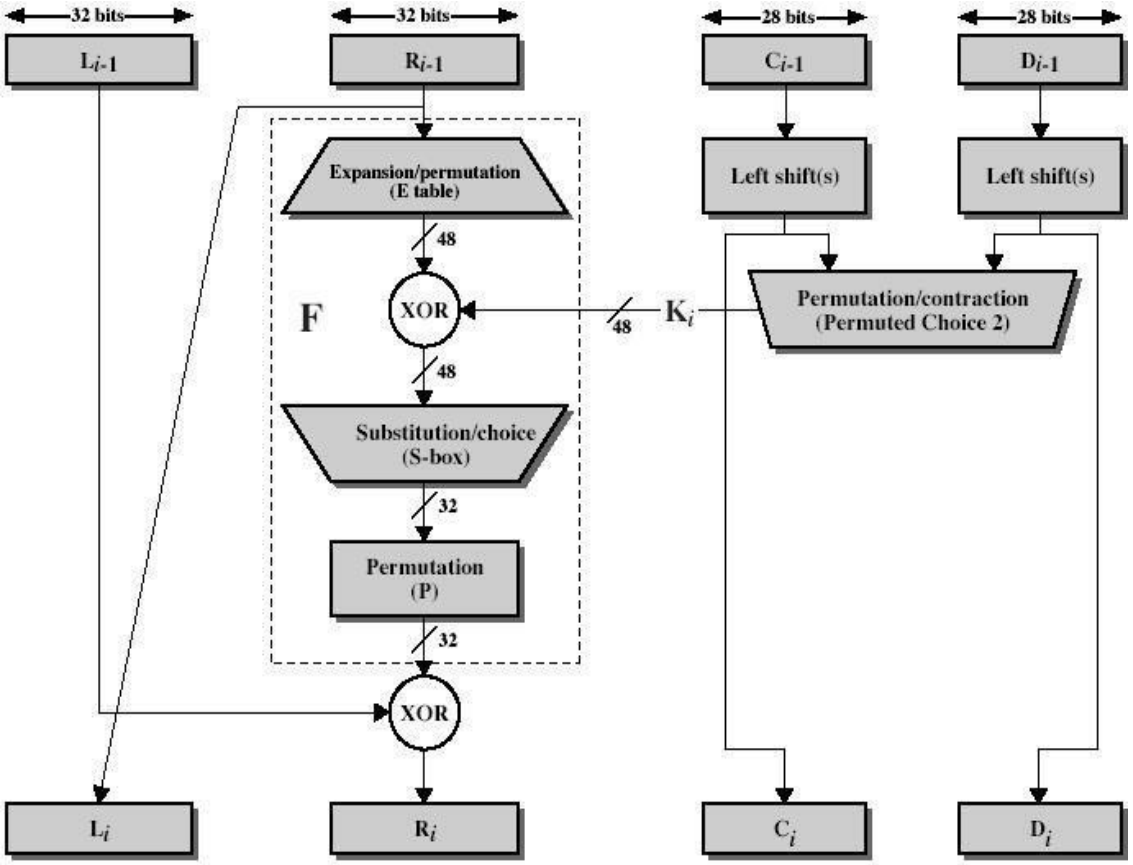
:

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

DETAILS OF SINGLE ROUND



Single Round of DES Algorithm

The left and right halves of each 64-bit intermediate value are treated as separate 32-bit quantities, labeled L and R. As in the classic Feistel cipher, the overall process at each round is summarized as follows:

$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$$

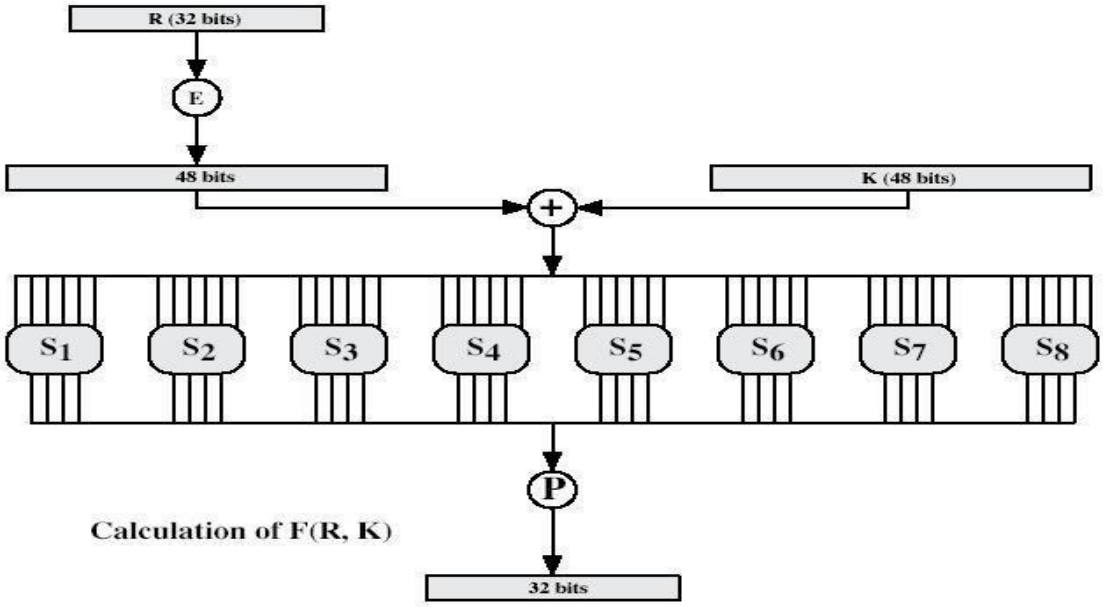
The round key Ki is 48 bits. The R input is 32 bits. This R input is first expanded to 48 bits by

Expansion/Permutation (E table):

Expansion/Permutation (E / P table)					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

The resulting 48 bits are XORed with Ki. This 48 bit result passes through a substitution function that produces 32-bit output, which is permuted by Permutation function (P):

Permutation function (P)							
16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25



Calculation of F(R, K)

The substitution consists of a set of 8 S-boxes, each of which accepts 6 bits input and produces 4 bits as output. These transformations are:

Each row of an S-box defines a general reversible substitution: middle 4 bits of each group of 6-bit input are substituted by S-box output, 1st and last 6th bits define what particular substitution out of to use.

S_1	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S_2	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
S_5	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S_6	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

KEY GENERATION

Input key has 64 bits. But each 8th bit is not used: bits 8,16,24,32,40,48,56,64 are notfurther used. The 56-bit key is first subjected to permutation Permuted Choice 1:

Permuted Choice 1 (PC-1) 5749413325179
    1585042342618 1025951433527 1911360524436 63 55 47 39 31 23 15 7625446383022 1466153453729

211352820124

The resulting 56-bit key is then treated as 2 28-bit quantities, labeled C0 and D0. At each round, C_{i-1} and D_{i-1} are separately subjected to a circular left shift, or rotation, of1 or 2 bits as governed by the following:

Schedule of Left Shifts
Round number 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Bits rotated 1 122222212 2 2 2 2 2 1

These shifted values serve as input to the next round. They also serve as input to Permuted Choice 2, which produces a 48-bit output that serves as input to the function

$F(R_{i-1}, K_i)$.

Permuted Choice 2 (PC-2)

141711241 5 3 28

156 21102319124

268 167 2720132

41 52 31 37 47 55 30 40

51 45 33 48 44 49 39 56

34 53 46 42 50 36 29 32

DES DECRYPTION

As with any Feistel cipher, decryption uses the same algorithm as encryption,except that the application of subkeys is reversed.

THE AVALANCE EFFECT IN DES

Avalanche effect − A small change in plaintext results in the very grate change in the ciphertext. 1 bit change in the plaintext leads to 34 bit difference in the ciphertext. 1 bit change in the key leads to 35 bit difference in the ciphertext.

THE STRENGTH OF DES

DES proved insecure in July 1998, when the Electronic Frontier Foundation (EFF) announced that it had broken a DES encryption using a special-purpose "DES cracker" machine that was built for less than $250 000. The attack took less than 3 days.

Design criteria for S-boxes were not made public, so there was a concern that cryptanalysis is possible for an opponent who knows the weaknesses in S-boxes. Up to now, there are no published results about such weaknesses in S-boxes.

DES also appears to be resistant to timing attack but suggest some avenues to explore. Timing attack tries to understand essence of algorithm by analysis of time of its work on different inputs.

One of such approaches yields a Hamming weight (number of bits equal to 1) of the secret key. brute-force attack on S-DES is feasible since with a 10-bit key there are only 1024 possibilities.

Password based authenticated Key establishment Protocols.

For symmetric encryption to work, the two parties to an exchange must share the same key, and that key must be protected from access by others. Furthermore, frequent key changes are

usually desirable to limit the amount of data compromised if an attacker learns the key. Therefore, the term that refers to the means of delivering a key to two parties who wish to exchange data, without allowing others to see the key. For two parties A and B, key distribution can be achieved in a number of ways, as follows:

1. A can select a key and physically deliver it to B.

2. A third party can select the key and physically deliver it to A and B.

3. If A and B have previously and recently used a key, one party can transmit the new key to the other, encrypted using the old key.

4. If A and B each has an encrypted connection to a third party C, C can deliver a key on the encrypted links to A and B.

Physical delivery (1 & 2) is simplest - but only applicable when there is personal contact between recipient and key issuer. This is fine for link encryption where devices & keys occur in pairs, but does not scale as number of parties who wish to communicate grows. 3 is mostly based on 1 or 2 occurring first.

A third party, whom all parties trust, can be used as a trusted intermediary to mediate the establishment of secure communications between them (4). Must trust intermediary not to abuse the knowledge of all session keys. As number of parties grow, some variant of 4 is only practical solution to the huge growth in number of keys potentially needed.

Key distribution centre:

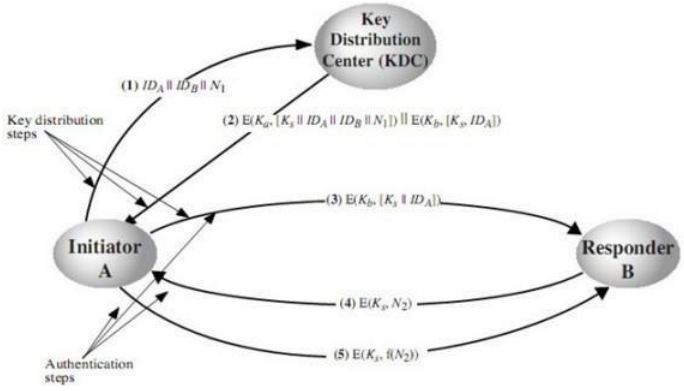
• The use of a key distribution center is based on the use of a hierarchy of keys. At a minimum, two levels of keys are used.

• Communication between end systems is encrypted using a temporary key, often referred to as a session key.

• Typically, the session key is used for the duration of a logical connection and then discarded

• master key is shared by the key distribution center and an end system or user and used to encrypt the session key.

Key Distribution Scenario:



Let us assume that user A wishes to establish a logical connection with B and requires a one-time session key to protect the data transmitted over the connection. A has a master key, Ka, known only to itself and the KDC; similarly, B shares the master key Kb with the KDC. The following steps occur:

1. A issues a request to the KDC for a session key to protect a logical connection to B. The message includes the identity of A and B and a unique identifier, N1, for this transaction, which we refer to as a nonce. The nonce may be a timestamp, a counter, or a random number; the minimum requirement is that it differs with each request. Also, to prevent masquerade, it should be difficult for an opponent to guess the nonce. Thus, a random number is a good choice for a nonce.

2. The KDC responds with a message encrypted using Ka Thus, A is the only one who can successfully read the message, and A knows that it originated at the KDC. The message includes two items intended for A:

• The one-time session key, Ks, to be used for the session

• The original request message, including the nonce, to enable A to match this response with the appropriate request Thus, A can verify that its original request was not altered before reception by the KDC and, because of the nonce, that this is not a replay of some previous request. In addition, the message includes two items intended for B:

• The one-time session key, Ks to be used for the session

• An identifier of A (e.g., its network address), IDA These last two items are encrypted with Kb (the master key that the KDC shares with B). They are to be sent to B to establish the connection and prove A's identity.

3. A stores the session key for use in the upcoming session and forwards to B the information that originated at the KDC for B, namely, E(Kb, [Ks || IDA]). Because this information is encrypted with Kb, it is protected from eavesdropping. B now knows the session key (Ks), knows that the other party is A (from IDA), and knows that the information originated at the KDC (because it is encrypted using Kb). At this point, a session key has been securely delivered to A and B, and they may begin their protected exchange.

However, two additional steps are desirable:

4. Using the newly minted session key for encryption, B sends a nonce, N2, to A.

5. Also using Ks, A responds with f(N2), where f is a function that performs some transformation on N2 (e.g., adding one).

These steps assure B that the original message it received (step 3) was not a replay. Note that the actual key distribution involves only steps 1 through 3 but that steps 4 and 5, as well as 3, perform an authentication function.

Major Issues with KDC:

For very large networks, a hierarchy of KDCs can be established. For communication among entities within the same local domain, the local KDC is responsible for key distribution. If two entities in different domains desire a shared key, then the corresponding local KDCs can communicate through a (hierarchy of) global KDC(s) To balance security & effort, a new session key should be used for each new connection-oriented session. For a connectionless protocol, a new session key is used for a certain fixed period only or for a certain number of transactions. An automated key distribution approach provides the flexibility and dynamic characteristics needed to allow a number of terminal users to access a number of hosts and for the hosts to exchange data with each other, provided they trust the system to act on their behalf. The use of a key distribution center imposes the requirement that the KDC be trusted and be protected from subversion. This requirement can be avoided if key distribution is fully decentralized. In addition to separating master keys from session keys, may wish to define different types of session keys on the basis of use. KEY DISTRIBUTION

For symmetric encryption to work, the two parties to an exchange must share the same key, and that key must be protected from access by others. Furthermore, frequent key changes are usually desirable to limit the amount of data compromised if an attacker learns the key. Therefore, the term that refers to the means of delivering a key to two parties who wish to exchange data, without allowing others to see the key. For two parties A and B, key distribution can be achieved in a number of ways, as follows:

1. A can select a key and physically deliver it to B.

2. A third party can select the key and physically deliver it to A and B.

3. If A and B have previously and recently used a key, one party can transmit the new

key to the other, encrypted using the old key.

4. If A and B each has an encrypted connection to a third party C, C can deliver a key

on the encrypted links to A and B.

Physical delivery (1 & 2) is simplest - but only applicable when there is personal contact between recipient and key issuer. This is fine for link encryption where devices & keys occur in pairs, but does not scale as number of parties who wish to communicate grows. 3 is mostly based on 1 or 2 occurring first.

A third party, whom all parties trust, can be used as a trusted intermediary to mediate the establishment of secure communications between them (4). Must trust intermediary not to abuse the knowledge of all session keys. As number of parties grow, some variant of 4 is only practical solution to the huge growth in number of keys potentially needed.

Key distribution centre:

• The use of a key distribution center is based on the use of a hierarchy of keys. At a minimum, two levels of keys are used.

• Communication between end systems is encrypted using a temporary key, often referred to as a session key.

• Typically, the session key is used for the duration of a logical connection and then

discarded

• Master key is shared by the key distribution center and an end system or user and used to encrypt the session key.

Key Distribution Scenario:
Let us assume that user A wishes to establish a logical connection with B and requires a one-time session key to protect the data transmitted over the connection. A has a master key, Ka, known only to itself and the KDC; similarly, B shares the master key Kb with the KDC.

The following steps occur:

1. A issues a request to the KDC for a session key to protect a logical connection to B. The message includes the identity of A and B and a unique identifier, N1, for this transaction, which we refer to as a nonce. The nonce may be a timestamp, a counter, or a random number; the minimum requirement is that it differs with each request.

Also, to prevent masquerade, it should be difficult for an opponent to guess the nonce. Thus, a random number is a good choice for a nonce.

2. The KDC responds with a message encrypted using Ka Thus, A is the only one who can successfully read the message, and A knows that it originated at the KDC. The message includes two items intended for A:

• The one-time session key, Ks, to be used for the session

• The original request message, including the nonce, to enable A to match this response with the appropriate request

Thus, A can verify that its original request was not altered before reception by the KDC and, because of the nonce, that this is not a replay of some previous request.

In addition, the message includes two items intended for B:

• The one-time session key, Ks to be used for the session

• An identifier of A (e.g., its network address), IDA These last two items are encrypted with Kb (the master key that the KDC shares withB). They are to be sent to B to establish the connection and prove A's identity.

3. A stores the session key for use in the upcoming session and forwards to B the information that originated at the KDC for B, namely, E(Kb, [Ks || IDA]). Because this information is encrypted with Kb, it is protected from eavesdropping. B now knows the session key (Ks), knows that the other party is A (from IDA), and knows that the Kb).

At this point, a session key has been securely delivered to A and B, and they may begin their protected exchange. However, two additional steps are desirable:

4. Using the newly minted session key for encryption, B sends a nonce, N2, to A.

5. Also using Ks, A responds with f(N2), where f is a function that performs some transformation on N2 (e.g., adding one).

These steps assure B that the original message it received (step 3) was not a replay.

Note that the actual key distribution involves only steps 1 through 3 but that steps 4 and 5, as well as 3, perform an authentication function.

Major Issues with KDC:

For very large networks, a hierarchy of KDCs can be established. For communication among entities within the same local domain, the local KDC is responsible for key distribution. If two entities in different domains desire a shared key, then the corresponding local KDCs can communicate through a (hierarchy of) global KDC(s). To balance security & effort, a new session key should be used for each new connection-oriented session. For a connectionless protocol, a new session key is used for a certain fixed period only or for a certain number of transactions.

An automated key distribution approach provides the flexibility and dynamic characteristics needed to allow a number of terminal users to access a number of hosts and for the hosts to exchange data with each other, provided they trust the system to act on their behalf.

The use of a key distribution center imposes the requirement that the KDC be trusted and be protected from subversion. This requirement can be avoided if key distribution is fully decentralized. In addition to separating master keys from session keys, may wish to define different types of session keys on the basis of use.

Question Bank:

1. Explain security-monitoring model in Cyber forensics.
2. List the confidentiality and authentication service provided by PGP that are used for electronic mail and file storage Applications.
3. Analyse the problems and limitations of simple mail transfer protocol.
4. Differentiate symmetric key cryptography and asymmetric key cryptography.
5. Propose the model of symmetric key cryptography and identify few algorithms, which uses this model foe encryption and decryption.
6. Identify the strength and weakness of Data Encryption Standard.
7. Examine the issues with Key distribution centres.
8. Build the digital signature model used for authentication in PGP service.
9. Elaborate the cryptographic security services provided by S/MIME for electronic messaging applications.
10. Discuss the key distribution scenario in symmetric key cryptography environment with detailed diagram.
11. Explain the model elliptic curve key distribution protocol in detail.
12. Paraphrase trusted system. Explain the basic concept of data access control in trusted systems.
13. Discuss the importance of encryption in satellite data and explain the process.
14. Explain how the network forensics is witnessed in various situations with required diagrams
15. Discuss any one key distribution protocol and the possible attacks
16. Analyze the importance of encryption in satellite data and explain the process.

School of Computing
Department of Computer Science and Engineering

UNIT V- System Security and Practices - SCS7008

Perfect Forward Secrecy

- *Encryption can be used to keep communications secret.*
  - But what if someone forces you to disclose the key?

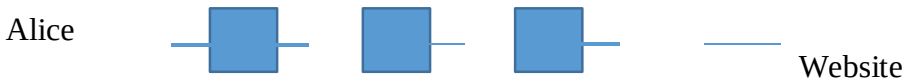
- *Perfect Forward Secrecy (PFS): gold standard of encrypted communications.*
  - Start with keys that allow Alice to authenticate Bob.
  - Alice and Bob create fresh public keys and exchange them.
  - They establish fresh shared keys, and talk secretly.
  - Once done, they delete the shared keys.

- *Result: after a conversation is over, no-one can decrypt what was said.*
  - Additional property: plausible deniability.
  - Illustrates: using only end devices, no long-term keys.

Protecting communications meta-data

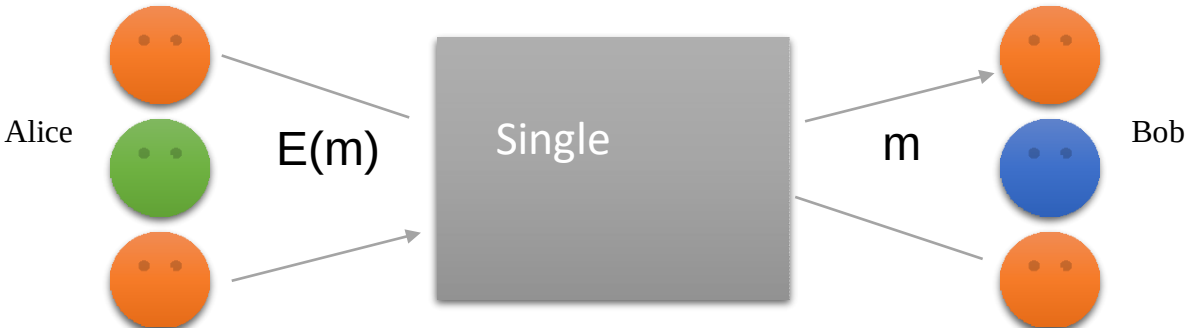
- *Who talks with whom, and what you browse is sensitive.*
  - Alice talks to Bob, Bob is a cancer doctor.
  - Alice Browses the NHS website, looking at pages on STDs.
  - *Extensive research shows a lot can be inferred from meta-data:*
    - Sender, receiver, length & time of communication, pattern.
  - Eg. mental condition, personality, language, emotions, political opinion.
  - Even if the content is encrypted!

- *Anonymous communication systems hide such information:*
  - Best known: Tor – The Onion Router.
  - How? Use a set of relays:



  - Illustrates: distribute trust, chose who to trust, crypto …

Proxies for Anonymous Communications



- *Alice wants to hide the fact she is sending a message to Bob.*
  - The proxy decrypts the message.
  - The proxy batches many messages.
  - The proxy is the TCB.
- *Problem:*
  - Low throughput.
  - Corrupt Proxy or Proxy hacked / coerced.
  - Real case: Penet.fi vs the church of scientology (1996)

Private Information Retrieval

- *Key problem: which database record you access is sensitive!*
  - Example: which book you are looking at the library? Which friend you checkif they are on-line?

    What music you are listening?

    Which minister you look up in your online address book?
- *PETs Solutions:*

- Private information retrieval: access a public record without leaking which – even to the provider! (Is that evenpossible?)
- ORAM: access your own private encrypted records, without divulging which (cheap) to cloud store.

- *Techniques: distribute trust, rely on client (e2e).*

Private Computations in general

Alice and Bob want to work out who is older, without telling each other their age – can they do that?

- Amazing result: any function that could be privately computed by providing the private inputs to a trusted thirdparty, can also be computed privately.
- Ie. Alice and Bob simply exchange cryptographic messages, and end up with the result! Neither of them learns theother's age!
- Also enables secure outsourcing.

Two families of techniques:

- Secure Multiparty Computation: well established and understood techniques based on secret sharing data.Commercial support (eg. Cybernetica's Sharemind).
- Homomorphic Encryption: allows operations on encrypted data. Toy Prototypes.

- *Warning: slow for generic computations.*
- Normal CPU 1,000,000,000s (GHz) of operations a second.
- Secure computations 1-10 per second (Hz) in general.

Specific Private Computations

- *Generic Private Computations slow – but specific ones can be fast.*
- Smart metering examples: aggregation, fraud detection, billing.
- Private road tolls.
- Private authentication and authorization.
- Simple private statistics.
- Detecting common elements in sets.

- *Application specific protocols can be practical.*
- But they need to be evaluated, and the computation needs to be simple.
- High-value simple computations are commonplace.

How mature are PETs?

• Not all PETs are equally well understood and mature for use.
• PFS: download “Signal” now. Demand it everywhere. 1B users (Whatsapp).
• Anonymity: Tor provides a weak form of anonymity, 1M users.
• ZKP: Pilots (EU Prime, Primelife, ABC4Trust)
• Specific Private Computations: pilots (Tor statistics & ENCS smart metering)
• PIR / ORAM: we can build it, not large scale deployments.
• Generic Private Computations: start-ups & pilots (Cybernetica & Microsoft)

• *Performance:*
• Encryption of communications and storage: super-fast, will not slow down anything you care about.
• ZKP: slow, but usually need to prove simple things.
• Anonymity / PIR / ORAM: is slower than normal communications.
• Private Computations: much slower – 6-9 orders of magnitude.

Other ways to protect privacy

Non-cryptographic technologies are also used to protect privacy.
They have their uses, particularly where a trusted third party exists.
• Remember the 5Cs: cost, compulsion, collusion, corruption, carelessness.

• *However some mechanisms are misunderstood:*
• Dataset anonymization.
• Query Privacy / Privacy in statistical databases.
• Restricting use of collected data.
• Logging to detect privacy compromises.

Data Anonymization

• Magical thinking: this cannot happen in general.

• *The problem of de-anonymization:*
• Any aspect of the “anonymized” dataset can be used to link the records to known named records.
• Example of Netflix (anonymous) vs. DBLP (named) de-anonymization.
• In general it is impossible to sanitise only the private information without severely scrubbing all the usefulness out of the dataset.
• Removing PII is not enough!

Query Privacy

• *“Would it not be nice if I could send complex queries to a database to extract statistics, and it returned results that are informative, but leak very little information about any individual?”*

• Possible: state of the art are “differential privacy” mechanisms.

- *Why is that possible (while anonymization was impossible):*
- The final result depends on multiple personal records.

• However it does not depend much on any particular one (sensitivity).
• Therefore adding a little bit of noise to the result, suffices to hide any record contribution.
• In the case of anonymization: need to add a lot of noise to all the entries.

• *Example: average height in the room via anonymization or query privacy.*

• Public policy:
• Notice the difference in the share of the architecture to provide robust privacy.
• Notice that a TTP holds the data.

Controls on usage of collected data

• *"Limiting collection is not practical, so why not place stricter limits on use instead?" - Favourite of Craig Mundie(ex-Microsoft)*

- In practice: use some access control mechanism to ensure that once collected the data in only used for somethings.
- Problems of this approach:
- How does the user, or anyone else, gets assurance of robustness?
- Abuses are invisible making this more difficult to police.
- Technically need to keep track of private data and policies – even more complex than ensuring it is not collected.
- Need to ensure consent for use, even more difficult than consent for collected (since user may not even beavailable – bulk datasets).

- *Nearly no research on how to robustly achieve this, and prevent abuse.*
- Basically: "trust us we would never do anything wrong".
- No clear direction to design such robust technologies

A cautionary note on more logs

- *"Well it is simple: you collect all the data, and then you audit all operations and access to it. Thus if anyone abuses ityou can find them and punish them"*
- So many problems with this … *Issues:*
- Authorized accesses are themselves sensitive: eg. accesses to medical records. Access to contacts.
- It is not guaranteed that the unauthorized access was not itself the result of a compromised user in theorganization.
- Once personal data leaks it cannot be put back in the bottle. Eg. the leakage of private pictures of celebrities.

- *Public Policy: detecting compromises after the fact is one of the weakest*

security mechanism, and a weak privacy mechanism. It is not even clear someone can get punished.

Public verifiability and protection

• "How do I know this software I am using provides a gold standard level of privacy protection through PETs?"

• Key question!

• *Answer 1: we leave it up to everyone to examine!*

• Enormous externality – each user must be an expert and check.

• *Answer 2: provide clear specifications, require applications providing privacy to provide transparency in their code &operations.*

• Concept of "Public verifiability" of both code and operations.

• Gold Standard in the world of PETs (PGP, Tor, Signal,…)

• Reluctance from industries to adopt for PETsor anything else.

• Serious public policy issue beyond PETs (VW scandal).

• *At what point does society have a right to know how key machinery works?*

• Remember: this was the aim of patents, originally.

• This space will require serious regulation & consumer protection.

Personal privacy Policies

privacy is even more complex than security, involving protection of sensitive data in both electronic and physical forms. Federal law recognizes no difference in the levels of protection expected for physical and electronic data. Privacy also involvesprotecting that which is personal, including an individual's body, belongings, and private life. Theft and stalking are clearly the responsibility of the campus police, but matters such as who should have access to the list of visitors to a dorm does not fall under their auspices. Other privacy matters that don't involve the campus police include access to e-mail and voicemail. Privacy is addressed by both policy and law.

There's an "expectation of privacy" at most universities. Gaps exist in definitions of what should be considered sensitive or personal. How to apply these principles in practical, operational terms challenges most universities.

Protecting the Sensitive

A privacy policy dictates who should know what. Policies and procedures supported by system enhancements can largely address protection of sensitive information, often identified or implied by federal laws or community expectations. Privacy is more important now because of linkages and access to data that weren't available before. Examples of potentially sensitive information include the following:

- Social Security numbers

- Grades
- Financial aid
- Research
- Donor information

- Health records
- Physical activity (such as garage or shuttle use)
- Student information
- Employee information
- Applicant information
- Credit card information
- Names
- Addresses
- Communications (who sends to who)
- E-mail content
- Network logins

Protecting the Personal

Protecting personal information has little to do with system automation, being primarily a matter of policies and procedures that govern human interaction. Privacy violations are not broadcasted or publicly disclosed but instead are reported to ombudsmen at many universities. Privacy concerns range from trivial matters to potential criminal violations. Examples include:

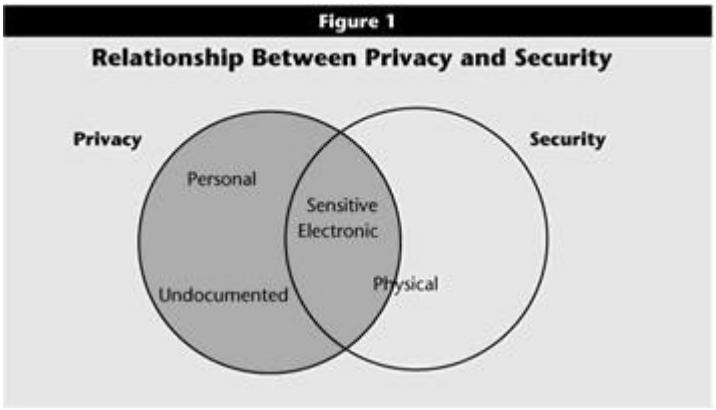
- Access to e-mail and voicemail
- Access to data on borrowed or loaned computers
- Access to an individual's desk
- Hacking
- Use of Social Security numbers on forms
- Salary questions
- Nosy supervisors
- Discomfort with undressing in certain areas due to physical abnormalities
- Inquiries about personal health
- Inquiries about reasons for time off
- Disability needs
- Stalking

Parents, students, university staff, and faculty report these concerns. Often, a conversation initiated by the ombudsperson with the relevant party resolves these matters simply.

Privacy and Security Intersect

Several areas of concern are common to both privacy and security: policy establishment, communication, training and enforcement, procedures, detection/discovery of intrusions, notification of victims, and response to intrusions. Theoretically, security should protect privacy. However, they don't match perfectly—they overlap (see Figure 1). Security involves protection of the physical and virtual realms. Sensitive information in a form that could be accessed by others (such as paper or electronic documentation) might be protected by security. Security measures typically do not protect those things that are personal

and not documented, however. These matters should be protected by privacy policies.



The generally accepted role of information security is to support information privacy, but in some situations, one might be compromised for the sake of the other. For example, threatening e-mails might be accessed (a violation of privacy) to protect the security of potential victims. This interrelationship implies that one needs to be considered "superior" to the other, or at a minimum a plan established to decide which is more important.

1. Conduct Research

Many universities have assembled a task force to assess risks and areas for improvement. Potential areas for investigation include usage of Social Security numbers, community expectations for privacy, a resource audit (to determine whether the university has the system and human resources to adequately address privacy), and development of metrics to measure the effectiveness of information security and privacy programs.

2. Appoint a Privacy Officer

Create a privacy officer position to serve as a full-time resource exclusively dedicated to privacy. This person can address the diverse privacy issues that either are neglected or only partially addressed by different departments having no common policiesor comprehensive reporting and tracking of issues. To ensure the goals of legal compliance and electronic security, this officer must build a strong alliance between the legal department and central IT. Individuals responsible for compliance with specific regulations (such as HIPAA or FERPA) should report to this person, who will provide general oversight of all privacy-related matters at the university. The privacy officer should be supported by designated compliance representatives as well as a privacy advisory board. This position should report to the president as a signal of the importance given to privacy and to ensure impartiality. A conflict of interest could result if the privacy officer reported to the IT or legal departments.

3. Establish a Privacy Advisory Board

Just as security experts exist, so do privacy experts. A group of experts and high-ranking representatives of the administration, academic departments, and the student body should be appointed to a privacy advisory board chaired by the privacy officer. The board should meet once a month, at a minimum, to proactively manage privacy at the university, including providing education and awareness programs to the community, reviewing regulations, establishing policies, and creating task forces to manage specific initiatives.

4. Establish an Insider Network of Privacy Advocates

Security is effectively addressed by IT systems and physical security teams. Privacy, however, requires many more manual adjustments in processes that must be performed by people. For maximum acceptance of privacy policies, tap into graduate students, faculty, and administrators with a passion for and expertise in this subject. These individuals could be used as researchers, privacy board members, or privacy advocates. Universities with successful privacy programs rely heavily on a network of liaisons inside each department who have a personal interest in privacy.

5. Launch Information Security and Privacy Campaigns

Create a culture where the community has the knowledge (what to do), skill (how to do it), and attitude (desire to do it) that support information security and privacy objectives. Security and privacy awareness must be part of an intentional, systematic, organizational change effort that adjusts attitudes and reshapes values and norms. These campaigns should be separate and led by the information security officer and privacy officer, with annual events to continually promote awareness and education.

Security and Privacy Issues in Environmental Monitoring

Environmental monitoring systems and the data they collect can be vulnerable to security and privacy risks. In particular, security risks are related to the threats that can undermine confidentiality, integrity, and availabilityof both the data and the monitoring systems in their entirety (e.g., system architecture and communication infrastructure). Conversely, privacy risks are related to those threats that can allow an adversary to use the environmental data for inferring sensitive information, which is not intended for disclosure and should be kept private.

Security and privacy risks are not independent:
they are often correlated, and an adversary can exploit a security violation for

breaching data privacy. As an example, suppose that Alice successfully violates the physical security of processing node PN, causing a security violation that can allow her to access private information related to the pollutant levels in the air of San Francisco. This security violation can allow Alice to infer pathologies of the citizens of a given area of the city, violating therefore their privacy.

Security risks

our environmental monitoring scenario security risks are related to all threats that can:

i) damage the infrastructure of the monitoring system;

ii) violate communication channels connecting different components of the monitoring system;

iii) allow unauthorized parties to intrude into the monitoring system for malicious purposes. We now describe in details these threats. Damages to the system infrastructure. Any attack performed with the aim of physically damaging the monitoring system can put at risk the confidentiality, integrity, and availability of the collected environmental data. For instance, suppose that the local municipality of San Francisco wants tobuild a new playground for children and, to determine the safest location, it analyzes the collected environmental data to discard polluted areas of the city. Suppose also that Alice maliciously damages the sensor nodes close to factory A, to hide evidences of the pollutants and production rejects release. Clearly, this compromises the collection of the environmental data, since these sensor nodes become not available (data availability violation). An analysis of the partial environmental data available to the local municipality can erroneously identify an area close to factory As the safest area where building the new playground. If this were to happen, children would be exposed to pollutants and production rejects.

iv) Privacy risks

Privacy risks are related to all threats that can allow an adversary to infer sensitive information from the collected environmental data. Such inferences can be direct, that is, caused by observations in the data collection (e.g., an adversary observing production rejects can discover confidential details of the productive processes of a company), or indirect (e.g.,studies on the presence of polluting substances in geographical areas or workplaces can be correlated with studies on the relationship between correlating pollutants and diseases, revealing possible illnesses of individuals living in those areas).

•**Data correlation and association**.

A possible means through which sensitive information can be inferred is represented by the natural correlations existing among different phenomena. To illustrate, consider a life and sickness insurance company in San Francisco. Suppose that a third-party organization releases a study illustrating the relationship existing between pollutants and rare diseases. Suppose also that the insurance company accesses this study.

By analyzing environmental data collected by the local municipality, and

comparing them with the study, the insurance company can decide to increase the risk associated with citizens living in polluted areas of San Francisco and re-compute their insurance policies. In addition to correlation, also the association of environmental data with other information coming from different sources can be exploited for inferring sensitive information. For instance, suppose that Alice can access a collection of data recording the medical histories of a community of patients

•**Data evolutions**.

To obtain more meaningful data, sensor nodes can perform several measurements of quantities of interest over time. For instance, a measuring station can continuously record the noise level in a given area of a city. While a high number of samples allows for better analysis of a given phenomenon, such repeated measurements can open the door to possible inference channels leaking sensitive information. For instance, suppose that Alice wants to discover the timetable of the freight trains traversing the railroad in San Francisco, which is kept secret by the local train company. Suppose also that the environmental monitoring of the local municipality includes the measurements of the noise pollution in the city. Having access to the measurements collected close to the railway, Alice can notice peaks in the noise levels and correlate this information with the public timetables of passenger trains, thus re-constructing the freight trains timetable.

• **Unusual data**.

Intuitively, if the measurements obtained from an environmental monitoring system deviate from what is expected or considered as usual, a high risk of sensitive information inference can arise. To illustrate, suppose that the results of the environmental monitoring of the San Francisco city area show a high level of radioactivity. If the neighbor cities do not show such a high level of radioactivity, then these values can be considered surprising, and may witness the existence of a neighbor location storing radioactive material (e.g., nuclear weapons, or rejects of nuclear power plants). Otherwise, if the same level of radioactivity is observed also in other cities, the radioactivity in San Francisco can be due to some peculiarities of the soil.

• **Users' locations**.

Mobile phones and smartphones are portable computers that more and more users have and carry with them all times. In the near future, we can imagine that our phones will be equipped with sensors and applications specifically targeted to the environmental monitoring, leading to a pervasive environmental monitoring where the sensing will be directly performed by users who will collect data related to the locations they visit. Since users move around the space, measurements have to be tagged with the location in which they have been captured. An adversary able to track the movements of a given user can violate her privacy discovering her frequent addresses (e.g., home and workplace), usual movements (e.g., from home to work), habits, and, accordingly, infer sensitive information about her. For

instance, suppose that Alice gains access to the set oflocation-tagged environmental measurements

Storage Area Network Security

Storage area network (SAN) **security** refers to the collective measures, processes, tools and technologies that enable the securing of a SAN infrastructure. It is a broad process that ensures that the SAN infrastructure operates securely and is protected from any vulnerabilities.

Storage Area Network Architecture (SAN Architecture)

Storage area network (SAN) architecture refers to the logical layout of a SAN infrastructure.
This architecture defines:

- How the SAN is logically created
- Components used
- Data storage and retrieval frameworks
- Device/host interconnectivity
- Other parameters/components essential for a SAN

Explains Storage Area Network Architecture (SAN Architecture)

There are two types of SAN architecture, storage-centric SAN

architecture and network-centric SAN architecture.A SAN generally

consists of three core components; therefore, SAN architecture is

composed of:

- Hosts: These are the system/end devices that use the SAN services. This can include servers and computers on thenetwork.
- Fabric: This consists of the interfaces such as fiber channel and host bus adapter that enable connectivity between thehosts and SAN infrastructure.
- Storage: This is the physical storage drives.

Typically, SAN architecture defines:

- Pool of storage used and how it is shared in between different servers or computers connected via the network
- Type of network or data transmission connection used between the keySAN infrastructure and all connecting nodes
- Placement of data depending upon the type of SAN architecture or topology
- Type of SAN topology being used

SAN architecture also includes SAN management applications and the overall data storage, consumption and retrieval policy that governs the SAN's resources.

Storage Area Network Management (SAN Management)

Storage area network (SAN) management refers to the collective measures, processes, tools and technologies that enable the operation, administration and maintenance of a SAN infrastructure.

It is a broad term that utilizes a layered approach that starts from the lower hardware level to the top software level in managing and maintaining a SAN infrastructure.

Explains Storage Area Network Management (SAN Management)

SAN management is generally performed through a central location, usually in the form of a SAN management application or a SAN server. This enables a central interface to allocate, manage and monitor storage. SAN management also includes monitoring the utilization of SAN resources, resolving issues, ensuring security and optimizing operations for best performance.

SAN management may also include:

- Planning for future expansion
- Capacity management
- Support for virtualization/cloud use
- Infrastructure management
- Creating and managing RAID levels
- LUN mapping
- Usage monitoring
- Backup management

Question Bank:

1. Discuss the Internet privacy risks in detail and plan how to minimize those risks?
2. Analyse various privacy policies must be followed in Internet communication.
3. Identify the steps in which the communication data can be protected?
4. Discuss the problems with controls on collected data approach of privacy mechanism.
5. Examine public verifiability and discussion method of data privacy.
6. Conclude the physical security essentials in risk management.
7. Elaborate what makes privacy enhancing technologies a different approach when compared with other approaches.
8. Build a model for setting proxies to secure from anonymous communications.
9. Create a model for private technologies for the smart grid to secure the data.

10. Categorize the sensitive information that must protected in privacy protection.
11. Security measures typically do not protect those things that are personal and not documented, however these matters should be protected by privacy policies, So what are the factors need to be considered to fulfill the above scenario.
12. Elaborate what makes privacy enhancing technologies a different approach when compared with other approaches.
13. Discuss the Internet privacy risks in detail and plan how to minimize those risks?
14. Conclude the physical security essentials in risk management.