ଓଡ଼ିଶା ରାଜ୍ୟ ମୁକ୍ତ ବିଶ୍ୱବିଦ୍ୟାଳୟ, ସମ୍ବଲପୁର, ଓଡ଼ିଶା
Odisha State Open University, Sambalpur, Odisha
Established by an Act of Government of Odisha.

M.Sc. in Cyber Security (MSCS)

CSPL-12 – INFORMATION SECURITY

LAB MANUAL

EXPERIMENT-1

Aim: To study the Private Key and Public Key cryptographic systems.

1.0 Learning Objective:

At the end of the session you should be able to:

- be familiar with basic terminologies of cryptography
- understand the private key and public key cryptography.

1.1 Basic Terminologies

Cryptography

The art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form is called cryptography.

Plaintext: This is the original intelligible message or data that is fed into the algorithm as input.

Encryption algorithm: The encryption algorithm performs various substitutions and transformations on the plaintext.

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.

Ciphertext: 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 ciphertexts. The ciphertext is an apparently random stream of data and, as it stands, is unintelligible.

Decryption algorithm: This is essentially the encryption algorithm run in reverse. It takes the ciphertext and the secret key and produces the original plaintext.

Cipher: An algorithm for transforming an intelligible message into one that is unintelligible by transposition and/or substitution methods

Cryptanalysis: The study of principles and methods of transforming an unintelligible message back into an intelligible message without knowledge of the key. It is also called code breaking.

1.2 Types of Cryptography

Based on the security models different cryptographic algorithms have been developed to prevent and defend attacks. Based on the type of algorithms cryptographic systems are categories in to two categories**.**

1. Symmetric/Private key Cryptography

In symmetric/private key Cryptography, the encryption and decryption keys are known both to sender and receiver. The encryption key is shared and the decryption key is easily calculated from it. In many cases, the encryption and decryption keys are the same.

2. Public key Cryptography

In public key cryptography, encryption key is made public, but it is computationally infeasible to find the decryption key without the information known to the receiver.

1.3 Symmetric/Private Key Cryptography

In private-key cryptography, the sender and recipient agree beforehand on a secret private key. The plaintext is somehow combined with the key to create the cipher text. The method of combination is such that, it is hoped, an adversary could not determine the meaning of the message without decrypting the message, for which he needs the key. The following diagram illustrates the encryption and decryption process.

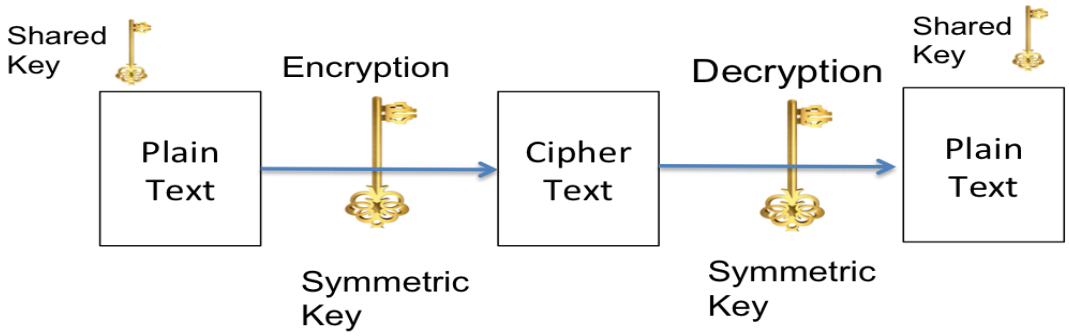


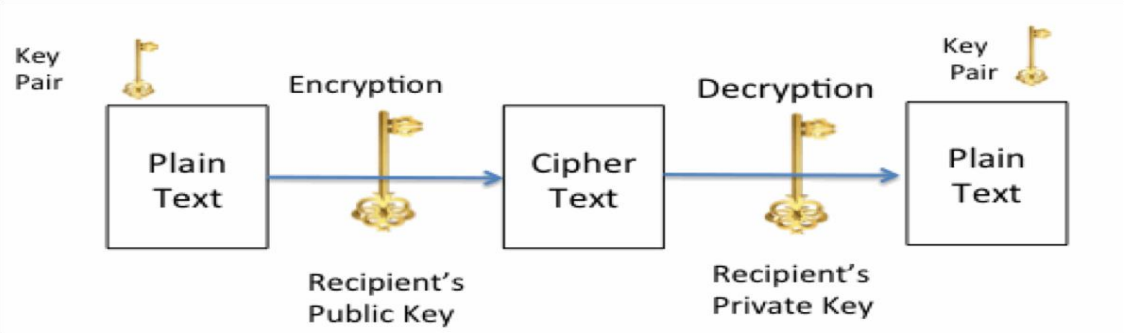
Fig.: Symmetric key Cryptography

To break a message encrypted with private-key cryptography, an adversary must either exploit a weakness in the encryption algorithm itself, or else try an *exhaustive search* of all possible keys (brute force method). If the key is large enough (*e.g.*, 128 bits), such a search would take a very long time (few years), even with very powerful computers. Private-key methods are efficient and difficult to break. However, one major drawback

is that the key must be exchanged between the sender and recipient beforehand, raising the issue of how to protect the secrecy of the key. When the President of the United States exchanges launch codes with a nuclear weapons site under his command, the key is accompanied by a team of armed couriers. Banks like wise use high security in transferring their keys between branches. These types of key exchanges are not practical, however, for e-commerce between, say, amazon.com and a casual web surfer.

1.4 Public Key Cryptography

Asymmetric encryption uses two keys instead of only one. These keys are mathematically related and are known as the public key and the private key. The public key is known to everyone and can be freely distributed, while the private key is known only to the individual to whom it belongs. The public key is used to encrypt the message and then it is sent to the recipient who can decrypt the message using the private key. The message encrypted with the public key cannot be decrypted with any other key except for its corresponding private key. The following Diagram illustrates the encryption and decryption process in the public key cryptography.



There are several important principles regarding asymmetric cryptography:

Key Pairs: Unlike symmetric cryptography that uses only one key, asymmetric cryptography requires a pair of keys.

Public Key: Public keys by their nature are designed to be public and do not need to be protected. They can be freely given to anyone or even posted on the Internet.

Private Key: The Private Key should be kept confidential and never shared. Both directions. Asymmetric cryptography keys can work in both directions.

A document encrypted with a public key can be decrypted with the corresponding private key.

EXPERIMENT-2

Aim: To study the classical encryption techniques: substitution and transposition.

1.0 Learning Objective:

At the end of the session you should be able to:

- understand the simple substitution techniques
- understand simple transposition techniques

1.1 Classical Encryption Techniques

The two basic building blocks of Classical encryption techniques are substitution and transposition.

1.2 Substitution Techniques

A substitution technique is one in which the letters of plaintext are replaced by other letters or by numbers or symbols to obtain the cipher text.

If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns.

1.2.1 Caesar Cipher

The earliest known, and the simplest, use of a substitution cipher was by Julius Caesar. The Caesar cipher involves replacing each letter of the alphabet with the letter standing three places further down the alphabet. Let us consider the following example.

Plaintext: meet me after the toga party

Cipher: PHHW PH DIWHU WKH WRJD SDUWB

Note: The alphabet is wrapped around, so the letter following Z is A.

We can define the transformation by listing all possibilities, as follows:

Plaintext: a b c d e f g h i j k l m n o p q r s t u v w x y z

Ciphertext: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C

Let us assign a numerical equivalent to each letter:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Then the algorithm can be expressed as follows. For each plaintext letter: p, substitute the cipher text letter: C

$$C = E(3, p) = (p + 3) \bmod 26$$

Encryption:

A shift may be of any amount, so that the general Caesar algorithm is

$$C = E(k, p) = (p + k) \bmod 26$$

Where, k takes on a value in the range 1 to 25.

Decryption:

The decryption algorithm for the Caesar Cipher is simply

$$p = D(k, C) = (C - k) \bmod 26$$

Limitations of the Caesar Cipher

If it is known that a given ciphertext is a Caesar cipher, then a brute-force cryptanalysis is easily performed: simply try all the 25 possible keys.

- ***Note:*** Three important characteristics of this problem enabled us to use a brute force

Cryptanalysis of Caesar Cipher:

1. The encryption and decryption algorithms are known.
2. There are only 25 keys to try.
3. The language of the plaintext is known and easily recognizable.

1.2.2 Play fair Cipher

The best-known multiple-letter encryption cipher is the Playfair, which treats digrams in the plaintext as single units and translates these units into ciphertext digrams. The Playfair algorithm is based on the use of a 5 × 5 matrix of letters constructed using a keyword.

Example: Let us consider the example solved by Lord Peter Wimsey in Dorothy Sayers's *Have His Carcase:*

M	*O*	*N*	*A*	*R*
C	*H*	*Y*	*B*	*D*
E	*F*	*G*	*I/J*	*K*
L	*P*	*Q*	*S*	*T*
U	*V*	*W*	*X*	*Z*

In this case, the keyword is *monarchy*. The matrix is constructed by filling in the letters of the keyword (minus duplicates) from left to right and from top to bottom, and then filling in the remainder of the matrix with the remaining letters in alphabetic order. The letters I and J count as one letter.

Plaintext is encrypted two letters at a time, according to the following rules:

1. Repeating plaintext letters that are in the same pair are separated with a filler letter, such as x, so that balloon would be treated as ba lx lo on.
2. Two plaintext letters that fall in the same row of the matrix are each replaced by the letter to the right, with the first element of the row circularly following the last. For example, ar is encrypted as RM.
3. Two plaintext letters that fall in the same column are each replaced by the letter beneath, with the top element of the column circularly following the last. For example, mu is encrypted as CM.
4. Otherwise, each plaintext letter in a pair is replaced by the letter that lies in its own row and the column occupied by the other plaintext letter. Thus, hs become BP and ea becomes IM (or JM, as the encipherer wishes).

Strengths of Playfair Cipher:

The Playfair cipher is a great advance over simple monoalphabetic ciphers.

As there are only 26 letters, there are 26 × 26 = 676 digrams, so that identification of individual digrams is more difficult.

Furthermore, the relative frequencies of individual letters exhibit a much greater range than that of digrams, making frequency analysis much more difficult.

Limitations of Playfair Cipher:

Despite this level of confidence in its security, the Play air cipher is relatively easy to break, because it still leaves much of the structure of the plaintext language intact. A few hundred letters of ciphertext are generally sufficient.

1.3 Transposition Techniques

All the techniques examined so far involve the substitution of a cipher text symbol for a plaintext symbol. A very different kind of mapping is achieved by performing some sort of permutation on the plaintext letters. This technique is referred to as a transposition cipher.

1.**3.1 Rail fence techniques**:

Rail fence is simplest of such cipher, in which the plaintext is written down as a sequence of diagonals and then read off as a sequence of rows.

Plaintext = meet at the school house

To encipher this message with a rail fence of depth 2, we write the message as follows:

m e a t e c o l o s

e t t h s h o h u e

The encrypted message is MEATECOLOSETTHSHOHUE

1.3.2 Row Transposition Cipher

A more complex scheme is to write the message in a rectangle, row by row, and read the message off, column by column, but permute the order of the columns. The order of columns then becomes the key of the algorithm.

e.g., plaintext = meet at the school house

Key = 4 3 1 2 5 6 7

Plain Text = m e e t a t t

h e s c h o o

l h o u s e

The cipher text is: esotcueehmhlahstoeto

Strength of Row Transposition Cipher

A pure transposition cipher is easily recognized because it has the same letter frequencies as the original plaintext. The transposition cipher can be made significantly more secure by performing more than one stage of transposition. The result is more complex permutation that is not easily reconstructed.

EXPERIMENT-3

Aim: To analyse the encryption and decryption of RSA – Public Key Cryptography Algorithm.

1.0 Learning Objective:

At the end of the session you should be able to:

- Understand the encryption and decryption algorithm using RSA algorithm.

1.1 Introduction to RSA Algorithm:

RSA is one of the most popular and successful public key cryptography algorithms. The algorithm has been implemented in many commercial applications. It is named after its inventor's Ronald L. Rivest, Adi Shamir, and Leonard Adleman. They invented this algorithm in the year 1977. They utilized the fact that when prime numbers are chosen as a modulus, operations behave "conveniently". They found that if we use a prime for the modulus, then raising a number to the power (prime - 1) is 1.

RSA algorithm simply capitalizes on the fact that there is no efficient way to factor very large integers. The security of the whole algorithm relies on that fact.

1.2 RSA Algorithm:

The encryption and decryption in the RSA algorithm is done as follows. Before encryption and decryption is done, we have to generate the key pair and then those keys are used for encryption and decryption.

1.2.1 Key Generation:

The first step in RSA encryption is to generate a key pair. Two keys are generated of which one is used as the public key and the other is used as the private key. The keys are generated with the help of two large prime numbers. The keys are generated as follows.

1. Generate two large random prime numbers p and q.
2. Compute n which is equal to product of those two prime numbers, $n = pq$
3. Compute $\varphi(n) = (p-1)(q-1)$.
4. Choose an integer e, $1 < e < \varphi(n)$, such that $\gcd(e, \varphi(n)) = 1$.
5. Compute the secret exponent d, $1 < d < \varphi(n)$, such that $ed \equiv 1 \pmod{\varphi(n)}$.
6. The public key is (n, e) and the private key is (n, d).

The values of p, q, and $\varphi(n)$ should also be kept secret.

- n is known as the *modulus*.
- e is known as the *public exponent* or *encryption exponent*.
- d is known as the *secret exponent* or *decryption exponent*

1.2.2 Encryption:

Encryption is done using the public key component e and the modulus n. To whomever we need to send the message, we encrypt the message with their public key (e,n). Encryption is done by taking an exponentiation of the message m with the public key e and then taking a modulus of it.

The following steps are done in encryption.

1. Obtain the recipient's public key (n,e)
2. Represent the plaintext message as a positive integer m < n
3. Compute the ciphertext $c = m^e \bmod n$.
4. Send the ciphertext c to the recipient.

1.2.3 Decryption:

Decryption is done using the Private Key. The person who is receiving the encrypted message uses his own private key to decrypt the message.

Decryption is similar to the encryption except that the keys used are different.

1. Recipient uses his private key (n,d) to compute $m = c^d \bmod n$.
2. Extract the plaintext from the integer representative m. The RSA algorithm has been implemented in many applications and it is currently one of the most popularly used encryption algorithm. RSA algorithm is based fully on mathematics and in the next section we will see the mathematics behind RSA.

1.2.4 Mathematics behind RSA:

The RSA algorithm works as follows. It first finds two prime numbers and generates a key pair using those two prime numbers. Then the encryption and decryption are done using the key pair. p and q are distinct primes

$N = p \times q$

Find a, b such that $a \times b = 1 \bmod (p-1)(q-1)$

Encryption Key: $e = (b, n)$

Decryption Key: $d = (a, n)$

Encryption of message m: $E_e(m) = m^b \bmod n = C$ (cipher)

Decryption of C: $D_d(C) = c^a \bmod n = m$

EXPERIMENT-4

Aim: To study working of Intrusion detection System (IDS) tool.

1.0 Learning Objective:

At the end of this session you should be able to

- Understand the functions of an intrusion detection system
- Explore the Intrusion Detection System "Snort".
- Know the approach of Snort IDS, a signature based intrusion detection system used to detect network attacks.

1.1 Software Requirements

All required files are packed and configured in the provided virtual machine image.

The VMWare Software

http://apps.eng.wayne.edu/MPStudents/Dreamspark.aspx

The Ubuntu 14.04 Long Term Support (LTS) Version

http://www.ubuntu.com/download/desktop

Snort: A signature-based Intrusion Detection System

https://www.snort.org/#get-started

1.2 Intrusion Detection System

An intrusion can be defined as a successful attack that exploits vulnerability in a system or network infrastructure resulting in the violation of the security policy.

An Intrusion Detection System (IDS) can prevent attacks by detecting an intrusion through vigilant monitoring of networks and computers and reporting alerts in the instance of an attack. This can be compared with that of a real time burglar alarm, where the alarm alerts the owner of a possible burglary. When the Intrusion Detection System spots a violation of security policy, it immediately logs details of the event and presents it in the format of a report to the system administrator.

1.3 Functions of IDS

Irrespective of the type, all the IDS typically have the same functionalities: x Observing and Monitoring: An IDS observes the system and the network for any suspicious events. The criteria for observation depend on the type of IDS. A detailed description

on the types of IDS and their modes of observations is provided under the Section 3.4.
x Logging of Events: On encountering a suspicious activity, the IDS records the information related to the observed activity. This is either performed locally by the concerned system (if the IDS has been installed on a single system) or by a centralized logging server (if the IDS has been set up for monitoring an entire network).
Alerting System Administrators: Once the events have been logged onto a database, the IDS can be set up to send alerts to the System Administrator. An IDS can send alerts through web pages, emails, messages, etc.

1.4 Reports on Intrusions: A detailed report is prepared listing the details on the events, which had been captured and logged. These reports can be used by System Administrators to analyze the security setup for the organization and for determining vulnerabilities.

1.5 Need for IDS

An Intrusion Detection System can be implemented by an organization for the following reasons.

- An IDS tends to act as an extra layer of protection along with the other security mechanisms in an organization.
- It aids in the process of detecting intrusions and other malicious events when other security measures in the organization fail.
- It can detect an attack in its preliminary stages when the attacker initiates a port scan to determine vulnerable ports.
- It can log events and present reports that can be used by the system administrator to determine the existing threats to an organization.
- It acts as quality control tool and aids in strengthening the vulnerabilities in an organization.
- It provides an easy technique for analyzing the security measures of an organization.

1.6 Intrusion Detection using SNORT

SNORT is a signature-based Network Intrusion Detection System tool, developed by Martin Roesch in 1998 that will be used in this project. SNORT was specifically chosen as it is a light weight IDS that is suitable for a small network. The rules used in SNORT are predefined and they can easily be edited as per the security needs of the organization. Currently, it is one of the most widely used IDS as it can be

utilized to detect Denial of Service (DoS) attacks, buffer overflow attacks, port scanning, scanning worms and viruses, SQL injections, HTTP injections, etc .

One of the major advantages of SNORT is that it could be used along with an external database such as MySQL in order to log the alerts and an output system such as Barnyard. Barnyard can read the output of SNORT, which is in a special binary called unified and send back the data in to the database.

In general, SNORT plays the combined roles of a packet sniffer, packet logger and a network monitor.

1.7 Components of SNORT

In order to understand the functioning of SNORT as IDS, it is important for the reader to understand about the different components of SNORT and the tasks performed by each. SNORT consists of five different components, which together can detect intrusions and log alerts.

The five major components of SNORT are: packet decoder, pre-processor, detection engine, logging and alert systems and output module. On close examination, it is evident that SNORT is designed based on the CIDF model. The diagrammatic representation of the different components is shown below. All the packets originating from the network are initially sent to the packet decoder and depending on the nature of the packet, it is either dropped or an alert is reported to the System Administrator.

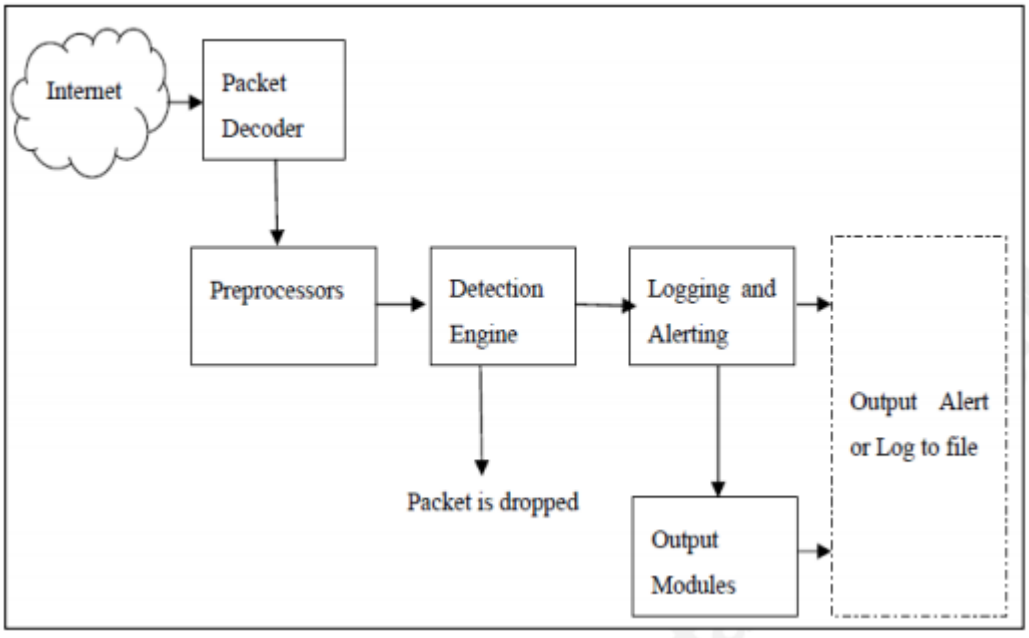


Fig: Components of SNORT

Packet Decoder: The packet decoder captures all the packets from the network and these packets are sent to the pre-processor.

Pre-processor: A pre-processor is used to aid the process of detection before sending the packets to the detection engine. An attacker can create a malicious packet in such a way that the IDS does not detect it. An attacker can disguise a packet to escape a signature. The pre-processor is responsible for detecting such disguised packets. For instance, if a rule has been written to detect packets with string ”htttpd/conf” the attacker can try to escape being detected by modifying it as ”htttpd/./conf” of the pre-processor to identify the forged packet and rearrange to its original format before sending the packet to the detection engine. Similarly, an attacker can fragment a single packet into several smaller packets in order to hide the signature from being detected by the IDS. A pre-processor puts back the fragments as one single malicious packet.

Detection Engine: The detection engine is the most important component in the SNORT. The detection engine utilizes the rules in determining whether a packet contains a signature or not. A rule can be divided into two parts namely the rule header and the options. The rule header carries details about the action that needs to be executed by the rule and the criteria for matching an incoming packet. The rules are matched against all the incoming packets. The options field carries additional information that can be used to for rule matching against packets and also to determine which portion of the packet can be used to trigger the alert message. If a packet matches a rule, then an alert is generated by the next component (Logging and Alerting System) of SNORT, else the packet is dropped by the detection engine.

Logging and Alert System: Once, the detection engine detects a malicious packet after matching against the rules, details about the type of packet, its signature, source and destination addresses, the time of generation, etc. are logged. The logs generated by SNORT are known as unified and are in a binary format. These can be stored as text files or tcp-dump-style files. A report can be sent listing the number of alerts to the System administrator via email, message or a webpage.

Output Modules: An output module is used to decide how the generated logs are saved as an output. There are many options based on how the System Administrator wishes to view the output.

EXPERIMENT-5

Aim: To study the prevention mechanisms to avoid Virus and other Malware in one's PC.

1.0 Learning Objective: To know different counter measures of to avoid Virus and Malware in one's PC**.**

1.1 Introduction

There are many ways in which Virus and other Malware can enter into one's PC. Accordingly there are many preventive measures. Some of the preventive measures include the following.

- Use of Antivirus
- Use of Firewall
- Keeping Software updated
- Download only trusted programs
- Avoid pirated software
- Be cautious about Phishing and Social Engineering
- Be wise with Passwords

1.2 **Use of Antivirus** - The first line of defines in PC security is installing an antivirus. It is vital to choose an antivirus that best suits your requirements to protect your computer even from the most dangerous zero day threats, those that have not yet been diagnosed by security analyst. There are many antivirus products, but only Comodo is architected to thwart even zero day threats.

1.3 **Use of Firewall** - Firewall along with antivirus stands as the first line of defines in the mechanism of PC security. Always ensure that the built-in firewall is enabled. It is also important to configure the Firewall correctly. *Firewall* blocks suspicious programs and hence provides PC security.

1.4 **Keeping Software updated** - It is not just about installing a software and application in your system, be it software or an operating system or any other application. Software that you install can end up having some security issues. Hence updating the software that is already installed in your system can prevent the hackers from exploiting the system and thereby making it prone to malicious attacks.

1.5 **Download only trusted programs -** Ensure that you download and run programs and applications from a trusted source. Also make sure you do not open any executable software that comes as email attachments.

1.6 **Avoid pirated software -** if you want to ensure *PC security*. It is the best choice to avoid cracked or pirated software. There is a culprit who provides malicious programs in the form of useful software that might let a less experienced user to run the program, thereby getting infected.

1.7 **Be cautious about Phishing and Social Engineering -** Do not share sensitive and personal information online. Though there is a protective measure that is offered by Browsers to protect from phishing attacks, they cannot be perfect any day. Make sure that you share your details over the network only to authenticated individuals and websites. Do not dare to click on links that are sent to you through mails, they might direct you to malicious sites.

1.8 **Be wise with Passwords -** It is advisable to use a wise combination of alphabets either big or small, numerals, characters and so. Do not use the same password for all the accounts that you have over the network. Implement the use of unique password for each of your accounts.

EXPERIMENT-6

Aim: To study the prevention mechanisms to protect one's PC from Hackers.

1.0 Learning Objective:

At the end of the session you should be able to know the simple steps in which you can protect your PC from Hackers.

1.1 Introduction

In this world of ubiquitous computers and persistent threats from hackers, protecting your computer is a must. The key pathway through which malware attacks the system is the Internet and its popular service, the Web.

There are numerous ways to protect and remove malware from our computers. No one method is enough to ensure your computer is secure. The more layers of defense, the harder for hackers to use your computer. Here are five simple, but critical steps to protect your computer,

- Install Firewall
- Install Antivirus Software
- Install Anti-Spyware Software
- Use Complex and Secure Passwords
- Check on the Security Settings of the Browser

1.2 Install Firewall

A **firewall** enacts the role of a security guard. There are of two types of firewalls: a software firewall and hardware firewall. Each serves similar, but different purposes. A firewall is the first step to provide security to the computer. It creates a barrier between the computer and any unauthorized program trying to come in through the Internet. If you are using a system at home, turn on the firewall permanently. It makes you aware if there are any unauthorized efforts to use your system.

1.3 Install Antivirus Software:

Antivirus is one other means to protect the computer. It is software that helps to protect the computer from any unauthorized code or software that creates a threat to the system. Unauthorized software includes viruses, key loggers, Trojans etc. This might slow down the processing speed of your computer, delete important files and access personal information. Even if your system is virus free, you must install antivirus software to prevent the system from further attack of virus.

Antivirus software plays a major role in real time protection; its added advantage of detecting threats helps computer and the information in it to be safe. Some advanced antivirus programs provide automatic updates, this further helps to protect the PC from newly created viruses.

Antivirus for Windows 8 software may include advanced features such as email protection, blocking of pop-ups and identity theft.

1.4 Install Anti-Spyware Software:

Spyware is a software program that collects personal information or information about an organization without their approval. This information is redirected to a third party website. Spyware are designed in such a way that they are not easy to be removed. Anti-Spyware software is solely dedicated to combat spyware. Similar to antivirus software, anti-spyware software offers real time protection. It scans all the incoming information and helps in blocking the threat once detected.Comodo Free Antivirus comes with spyware protection built in.

1.5 Use Complex and Secure Passwords:

The first line of defense in maintaining system security is to have strong and complex passwords. Complex passwords are difficult for the hackers to find. Use a password that is at least 8 characters in length and include a combination of numbers, letters that are both upper and lower case and a special character. Hackers use certain tools to break easy passwords in few minutes. One recent study showed that a 6 character password with all lower case letters can be broken in less than 6 minutes!

1.6 Check on the Security Settings of the Browser:

Browsers have various security and privacy settings that you should review and set to the level you desire. Recent browsers give you ability to tell web sites to not track your movements, increasing your privacy and security.