

**NEHRU COLLEGE OF ENGINEERING AND RESEARCH CENTRE
(NAAC Accredited)
(Approved by AICTE, Affiliated to KTU University, Kerala)**

ELECTRONICS AND COMMUNICATION ENGINEERING DEPARTMENT
Course Material

S7:EC366: Cyber Security

About the Department:

Department of ECE established in 2002 with an intake of 60 students to undergraduate (B.Tech) programme and enhanced to an intake of 120 students from 2006. The department offers two Postgraduates(M.Tech) programmes in “Electronics”. “Applied Electronics & Communication System” from 2011 with an intake of 18 students and “ VLSI Design” from 2012 with an intake of 18. Highly qualified, experienced and dedicated staff members are the backbone of the Department. The Department always strive hard to satisfy the knowledge thirst of both students and faculties by organizing workshops / technical talks / conferences etc. The faculty members are actively involved in research work and regularly present/ publish their work in various national and international conferences / journals. The ECE Department is proud to host state-of- the art Laboratories in the area of VLSI, Embedded Systems, Microprocessor and Microcontrollers, Circuits, Analog and Digital Communication and Microwave and Optical communication. The ECE department formally inaugurated the ECHOS (The ECE Association) in 2009 and under this banner many extra-academic activities have been conducted such as paper presentation, quiz competition, workshops and seminars. Also the department has two magazines that have been developed on the basis of the creative skills of our imaginative students. There is an Embedded Club that meets on monthly basis to discuss innovative projects and publication based activities. Department is closely associated with INSTITUTE OF ELECTRONICS & TELECOMMUNICATION ENGINEERS (IETE) Palakkad Centre to organize technical events like guest lecture, seminars and conferences.

Vision of the institute:

To mould true citizens who are millennium leaders and catalysts of change through excellence in education.

Mission of the institute:

NCERC is committed to transform itself into a center of excellence in Learning and Research in Engineering and Frontier Technology and to impart quality education to mould technically competent citizens with moral integrity, social commitment and ethical values. We intend to facilitate our students to assimilate the latest technological know-how and to imbibe discipline, culture and spiritually, and to mould them in to technological giants, dedicated research scientists and intellectual leaders of the country who can spread the beams of light and happiness among the poor and the underprivileged.

Vision of the department:

Providing Universal Communicative Electronics Engineers with corporate and social relevance towards sustainable developments through quality education.

Mission of the department:

- 1) Imparting Quality education by providing excellent teaching, learning environment.
- 2) Transforming and adopting students in this knowledgeable era, where the electronic gadgets (things) are getting obsolete in short span.
- 3) To initiate multi-disciplinary activities to students at earliest and apply in their respective fields of interest later.
- 4) Promoting leading edge Research & Development through collaboration with academia & industry.

Program Educational Objectives (PEOs)

- I. To prepare students to excel in postgraduate programmes or to succeed in industry / technical profession through global, rigorous education and prepare the students to practice and innovate recent fields in the specified program/ industry environment.
- II. To provide students with a solid foundation in mathematical, Scientific and engineering fundamentals required to solve engineering problems and to have strong practical knowledge required to design and test the system.
- III. To train students with good scientific and engineering breadth so as to comprehend, analyze, design, and create novel products and solutions for the real life problems.
- IV. To provide student with an academic environment aware of excellence, effective communication skills, leadership, multidisciplinary approach, written ethical codes and the life-long learning needed for a successful professional career.

Program Outcomes (Pos):

Engineering Graduates will be able to

1. **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem Analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of Solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct Investigations of Complex Problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern Tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and

modeling to complex engineering activities with an understanding of the limitations.

6. **The Engineer and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and Sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and Team Work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project Management and Finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long Learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSO):

Engineering Students will be to

1. Facility to apply the concepts of Electronics, Communications, Signal processing, VLSI, Control systems etc., in the design and implementation of engineering systems.

2. Facility to solve complex Electronics and communication Engineering problems, using latest hardware and software tools, either independently or in team.

Mapping of PEOs with the Program Outcomes (POs):

The Electronics and Communication Engineering Program outcomes leading to the achievement of the objectives can be summarized in the following Table.

		Program Outcomes										
		a	b	c	d	e	f	g	h	i	j	k
PEOs	1	X	X	X								X
	2	X	X	X	X		X					X
	3		X	X	X	X					X	
	4				X	X	X	X	X	X	X	X

Course Outcome:

1. To familiarize various types of cyber security methods and cyber-attacks.
2. To give an overview of the Mathematical approaches such as modular and polynomial arithmetic.
3. To study and expose to the different approaches that handle security and protect themselves and entire internet community from cyber-attacks.
4. To study and expose various crypt analysis for cyber security.
5. To familiarize the techniques and various algorithm in use for maintain data integrity and authenticity.
6. To enable the students to appreciate the practical aspects of security feature design and their implementation regarding intrusion technique and password management.

CO- PO Mapping:

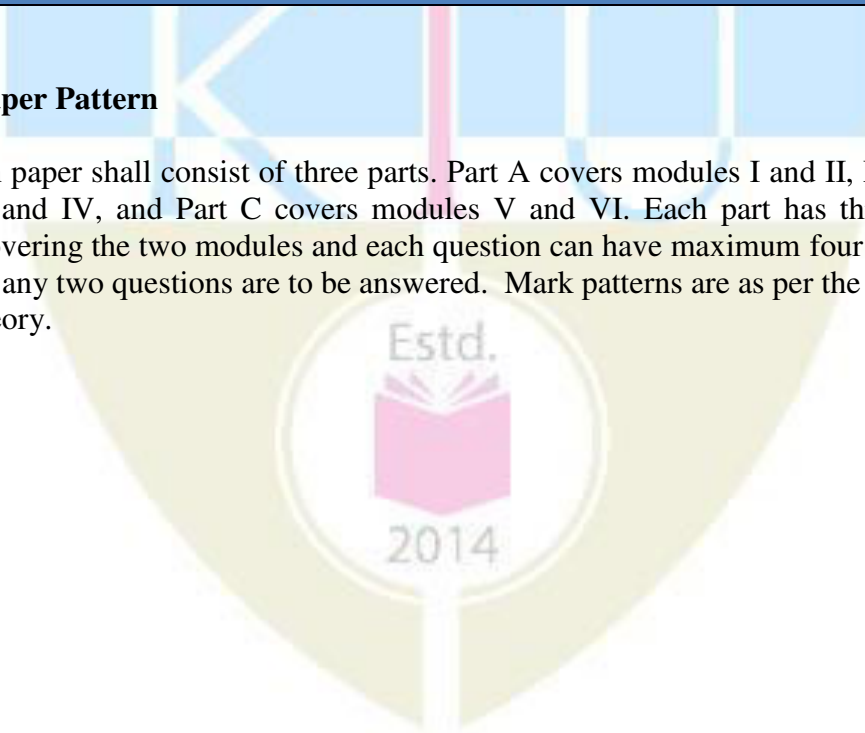
CO	PO	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	Avg
C468.1	3	3	2	3	2	3	3	3	3	3	1	3	2.66
C468.2	3	2	3	3	1	2	1	3	2	2	1	3	2.16
C468.3	3	3	3	3	3	3	3	3	3	3	1	3	2.83
C468.4	3	3	3	2	2	3	3	3	2	3	1	3	2.58
C468.5	3	3	3	3	2	2	2	3	2	3	1	3	2.5
C468.6	3	2	3	2	3	3	3	3	3	3	1	3	2.66
Avg	3	2.66	2.83	2.66	2.16	2.66	2.5	3	2.5	2.83	1	3	2.56

COURSE CODE	COURSE NAME	L-T-P-C	YEAR OF INTRODUCTION
EC466	CYBER SECURITY	3-0-0 -3	2016
Prerequisite: EC407 Computer Communication			
Course objectives: <ul style="list-style-type: none"> To familiarize various types of cyber-attacks and cyber-crimes. To give an overview of the cyber laws To study the defensive techniques against these attacks 			
Syllabus:			
Vulnerability scanning, tools for scanning, Network defense tools, Firewalls and Intrusion Detection Systems, Virtual Private Networks, Scanning for web vulnerabilities tools, Cyber crimes and law, cyber crime investigation			
Expected outcome: The students will be able to understand cyber-attacks, types of cybercrimes, cyber laws and also how to protect them self and ultimately the entire Internet community from such attacks			
Text Books: <ol style="list-style-type: none"> Mike Shema , Anti-Hacker Tool Kit, Mc Graw Hill Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley 			
References: <ol style="list-style-type: none"> Achyut S.Godbole Data Communication and Networking,2e, McGraw –Hill Education New Delhi,2011 Forouzan, Data Communication and Networking (Global Edition) 5/e, McGraw Hill Education India, 2013. Forouzan,TCP/IP Protocol Suite 4e, McGraw Hill Education India, 2010 			
Course Plan			
Module	Course contents	Hours	End Sem. Exam Marks
I	Introduction to Vulnerability Scanning Overview of vulnerability scanning, Open Port / Service Identification, Banner / Version Check, Traffic Probe, Vulnerability Probe, Vulnerability Examples, OpenVAS, Metasploit.	7	15%
II	Network Vulnerability Scanning Networks Vulnerability Scanning - Netcat, Socat, understanding Port and Services tools - Datapipe, Fpipe, WinRelay, Network Reconnaissance – Nmap, THC-Amap and System tools, Network Sniffers and Injection tools – Tcpdump and Windump, Wireshark, Ettercap, Hping, Kismet	7	15%
FIRST INTERNAL EXAM			
III	Network Defense tools Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection	8	15%

IV	Web Application Tools Scanning for web vulnerabilities tools: Nikto, W3af, HTTP utilities - Curl, OpenSSL and Stunnel, Application Inspection tools – Zed Attack Proxy, Sqlmap. DVWA, Webgoat, Password Cracking and Brute-Force Tools – John the Ripper, L0htcrack, Pwdump, HTC-Hydra	6	15%
SECOND INTERNAL EXAM			
V	Introduction to Cyber Crime and law Cyber Crimes, Types of Cybercrime, Hacking, Attack vectors, Cyberspace and Criminal Behavior, Clarification of Terms, Traditional Problems Associated with Computer Crime, Introduction to Incident Response, Digital Forensics, Computer Language, Network Language, Realms of the Cyber world, A Brief History of the Internet, Recognizing and Defining Computer Crime, Contemporary Crimes, Computers as Targets, Contaminants and Destruction of Data, Indian IT ACT 2000.	8	20%
VI	Introduction to Cyber Crime Investigation Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus and Worms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks	6	20%
END SEMESTER EXAM			

Question Paper Pattern

The question paper shall consist of three parts. Part A covers modules I and II, Part B covers modules III and IV, and Part C covers modules V and VI. Each part has three questions uniformly covering the two modules and each question can have maximum four subdivisions. In each part, any two questions are to be answered. Mark patterns are as per the syllabus with 100% for theory.



4.2 COURSE PLAN

Day		Status
<u>MODULE -1</u>		
1.	Introduction to Information Theory, Concept of information, units,	
2.	Entropy, marginal, conditional and joint entropies	
3.	Relation among entropies, mutual information, information rate.	
4.	Source coding: Instantaneous codes	
5.	Construction of instantaneous codes	
6.	Kraft's inequality	
7.	Coding efficiency and redundancy	
<u>MODULE – 2</u>		
9.	Noiseless coding theorem, construction of basic source codes	
10.	Shannon-Fano Algorithm	
11.	Huffman coding	
12.	Channel capacity – redundancy and efficiency of a channel	
13.	Binary symmetric channel (BSC)	
14.	Binary erasure channel (BEC	
15.	Capacity of band limited Gaussian channels	
<u>MODULE – 3</u>		
18.	Continuous Sources and Channels	

19.	Differential Entropy, Mutual information,	
20.	Waveform channels, Gaussian channels	
21.	Shannon – Hartley theorem, bandwidth	
22.	SNR trade off	
23.	Capacity of a channel of infinite bandwidth	
24.	Shannon's limit	
<u>MODULE – 4</u>		
27.	Introduction to rings, fields, and Galois fields	
28.	Codes for error detection and correction	
29.	Parity check coding – linear block codes	
30.	Error detecting and correcting capabilities	
31.	Generator and parity check matrices	
32.	Standard array and syndrome decoding	
<u>MODULE – 5</u>		
34.	Perfect codes, Hamming codes	
35.	Encoding and decoding Cyclic codes	
36.	Polynomial and matrix descriptions	
37.	Generation of cyclic codes,	
38.	Decoding of cyclic codes BCH codes	
39.	Construction and decoding	

40.	Reed Solomon codes	
<u>MODULE – 6</u>		
43.	Convolutional Codes ,encoding – time and frequency domain approaches	
44.	State Tree & Trellis diagrams	
45.	transfer function and minimum free distance	
46.	Maximum likelihood decoding of convolutional codes	
47.	The Viterbi Algorithm	
48.	Sequential decoding.	

QUESTION BANK

Module 04:

1. Briefly explain about Web Application Tools.
2. Explain the Scanning for web vulnerabilities tool of Nikto.
3. Explain the Scanning for web vulnerabilities tool of W3af.
4. Detailed explanation of HTTP utilities of Curl, OpenSSL and Stunnel.
5. Write short notes on HTTP utilities of Stunnel.
6. Write short notes on Application Inspection tools of Zed Attack Proxy & Sqlmap.
7. Write short notes on Application Inspection tools of DVWA & Webgoat.
8. Explain with example of Password Cracking and Brute-Force Tools.
9. Write short notes about John the Ripper & LOhtcrack
10. Write short notes about John the Pwdump & HTC Hydra
11. Briefly differentiate about the Zed Attack proxy and sqlmap.

Module05:

1. Explain Cyber Crimes with types and examples.
2. What is Hacking? List out Attack vectors, Cyberspace and Criminal Behavior in the hacking and Clarification the Terms with examples.
3. List out the Traditional Problems Associated with Computer Crimes.
4. Define Digital Forensics. Explain with examples.
5. Compare the Computer Language & Network Language with examples.
6. Summarize Realms of the Cyber world
7. Describe a Brief History of the Internet related to use of Cyber Crime.
8. Recognizing and Defining Computer Crime in our day to day life.
9. List out the Contemporary Crimes with examples.
10. How Computers are Targets in the cyber-crime.
11. Contaminants and Destruction of Data in the computer related to cyber-crime.
12. List the Indian IT ACT 2000 related to Cyber-Law.

Module 06:

1. List the procedure related to Cyber Crime Investigation.
2. Briefly explain about Firewalls and Packet Filters in the cyber crime investigation.
3. Explain Password Cracking with examples.
4. Briefly explain with examples of Key loggers and Spyware.
5. Differentiate Virus and Worms.
6. List out the difference between Trojan and backdoors.
7. Define and explain Steganography with examples.
8. What is DOS and DDOS attack? Explain with practical examples.
9. Define SQL injection. Explain with example.
10. What is Buffer Overflow? Explain with example.
11. How the Attack on wireless Networks will be happen and protection methods.

Module 1

CYBER SECURITY

VULNERABILITY SCANNERS

- A vulnerability scanner is software application that assesses security vulnerabilities in networks or host systems and produces a set of scan results.
- Both administrators and attackers can use the same tool for fixing or exploiting a system
- Administrators need to conduct a scan and fix problems before an attacker can do the same scan and exploit any vulnerabilities found

OPEN PORT

It is a TCP or UDP port number that is configured to accept packets. In contrast, a port which rejects connections or ignores all packets directed at it is called a **closed port**.

A port being "open" is not enough for a communication channel to be established. There needs to be an application (service) listening on that port, accepting the incoming packets and processing them.

Ports can be "closed" through the use of a firewall.

OPEN PORT

The design and operation of the Internet is based on the Internet Protocol Suite, commonly also called TCP/IP. In this system, network services are referenced using two components:

- A host address
- And a port number. - 65536 distinct and usable port numbers. Most services use a limited range of port numbers.

Some port scanners scan only the most common port numbers, or ports most commonly associated with vulnerable services, on a given host.

OPEN PORT

The result of a scan on a port is usually generalized into one of three categories:

- Open or Accepted: The host sent a reply indicating that a service is listening on the port.
- Closed or Denied or Not Listening: The host sent a reply indicating that connections will be denied to the port.
- Filtered, Dropped or Blocked: There was no reply from the host.

Open ports present two vulnerabilities -

- Security and stability concerns associated with the program responsible for delivering the service - Open ports.
- Security and stability concerns associated with the [operating system](#) that is running on the host - Open or Closed ports.
- Filtered ports do not tend to present vulnerabilities.

Open Port/Service Identification

Some services are insecure.

Ex - Telnet (port 23) - exposes passwords.

Fortunately, the widespread adoption of Secure Shell (SSH) has diminished the presence of telnet on the Internet.

Services do not always run on default ports; hence the scanner must rely on banners to get a response from a listening port.

A telnet service could be configured to listen on port 24601. If the scanner doesn't check that port, then it would miss the vulnerability.

Also, services do not always announce themselves. Telnet and SMTP (port 25) are promiscuous services; they return text-based banners upon receiving a connection, without waiting for any particular incoming data on that connection.

Conversely, HTTP (port 80) won't respond with data until the service receives a request that contains data (valid or otherwise)

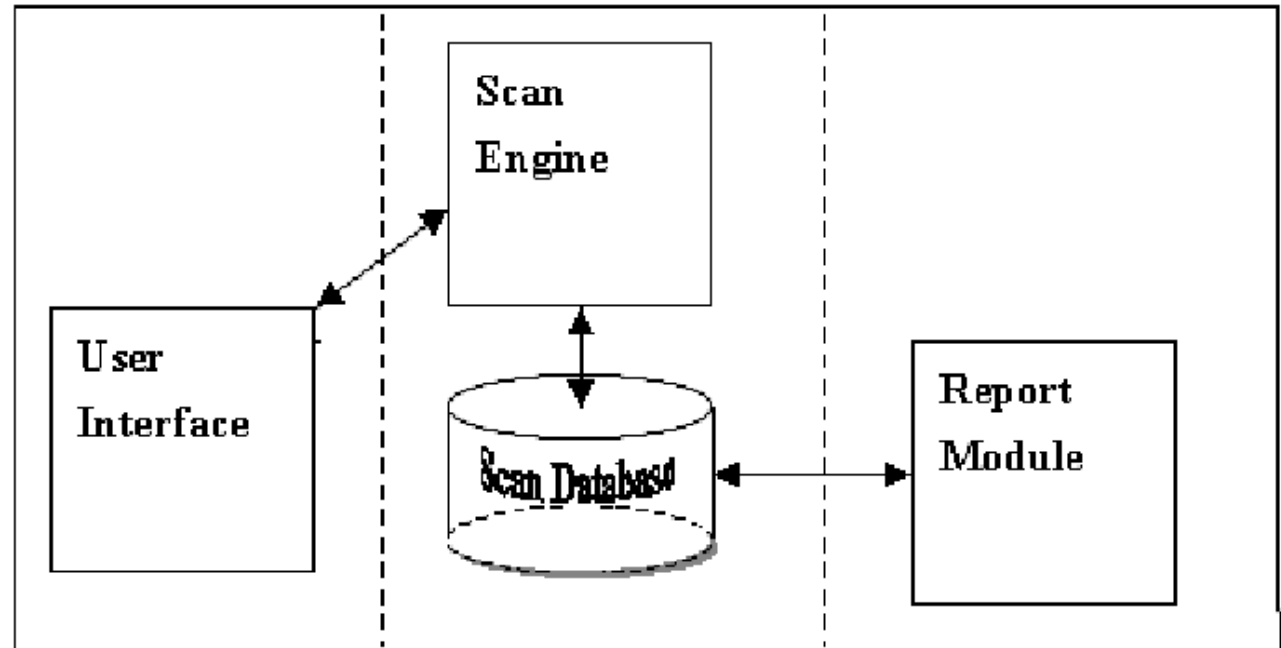
VULNERABILITY SCANNERS

- Vendor-originated.
- System administration-originated.
- User-originated.

ARCHITECTURE OF VULNERABILITY SCANNERS

It has four main modules

- A Scan Engine,
- A Scan Database,
- A Report Module
- And a User Interface.



ARCHITECTURE OF VULNERABILITY SCANNERS

- Scan Engine - Security checks according to its installed plug-ins, identifying system information and vulnerabilities.
- Scan Database - Stores vulnerability information, scan results, and other data used by scanner. Scanners with an "auto-update" feature can download and install the latest set of plug-ins to the database automatically.
- Report Module - Provides different levels of reports on the scan results, such as detailed technical reports with suggested remedies for system administrators, summary reports for security managers, and high-level graph and trend reports for executives.
- User Interface - allows the administrator to operate the scanner. It may be either a Graphical User Interface (GUI), or just a command line interface.

TYPES OF VULNERABILITY SCANNER

NETWORK-BASED SCANNERS - Usually installed on a single machine that scans a number of other hosts on the network. It helps detect critical vulnerabilities such as mis-configured firewalls, vulnerable web servers, risks associated with vendor-supplied software, and risks associated with network and systems administration.

HOST-BASED SCANNERS - A host-based scanner is installed in the host to be scanned, and has direct access to low level data, such as specific services and configuration details of the host's operating system

VULNERABILITY SCANNERS

BENEFITS OF VULNERABILITY SCANNERS

- Allows early detection and handling of known security problems.
- A new device or even a new system may be connected to the network without authorization.
- Helps to verify the inventory of all devices on the network.

VULNERABILITY SCANNERS

LIMITATIONS OF VULNERABILITY SCANNERS

- Can only assess a "snapshot of time" in terms of a system or network's security status. Therefore, scanning needs to be conducted regularly.
- Human judgement is needed: Vulnerability scanners can only report vulnerabilities according to the plug-ins installed in the scan database. They cannot determine whether the response is a false negative or a false positive.
- Others: a vulnerability scanner is designed to discover known vulnerabilities only. It cannot identify other security threats

EXAMPLES OF COMMON VULNERABILITY SCANNERS

1) Network-based scanners

a) Port scanners

- Nmap : <http://insecure.org/nmap/>
- Superscan: <http://www.foundstone.com/us/resources/proddesc/superscan4.htm>

b. Network vulnerability scanners

- Nessus : <http://www.nessus.org/nessus/>
- GFI LANguard Network Security Scanner (N.S.S.) (commercial) : <http://www.gfi.com/languard/>

c. Web server scanners

- Nikto : <http://www.cirt.net/code/nikto.shtml>
- Wikto : <http://www.sensepost.com/research/wikto/>

d. Web application vulnerability scanners

- Paros : <http://parosproxy.org/index.shtml>
- Acunetix Web Vulnerability Scanner (commercial) : <http://www.acunetix.com/>

EXAMPLES OF COMMON VULNERABILITY SCANNERS

2. Host-based scanners

a. Host vulnerability scanners

- Microsoft Baseline Security Analyser (MBSA):

<http://www.microsoft.com/technet/security/tools/mbsahome.msp>

- Altiris Security Expressions (commercial):

<http://www.altiris.com/Products/SecurityExpressions.aspx>

Banner grabbing

It is a technique used to gain information about a computer system on a network and the services running on its open ports. Administrators can use this to take inventory of the systems and services on their network. However, an intruder can use banner grabbing in order to find network hosts that are running versions of applications and operating systems with known exploits.

Some examples of service ports used for banner grabbing are those used by

- Hyper Text Transfer Protocol (HTTP) - ports 80
- File Transfer Protocol (FTP) - ports 21
- and Simple Mail Transfer Protocol (SMTP); ports 25.
- Tools commonly used to perform banner grabbing are Telnet, nmap, zmap and Netcat

Vulnerability Probe

Some security bugs can't be identified without sending a payload that exploits a suspected vulnerability.

These types of probes are more accurate—they rely on direct observation as opposed to inferring problems based on port numbers or service banners. But they also carry more risk of interrupting the service, because the test payload must be trying to either produce or take advantage of an error in the service's code.

An easy-to-understand example of a vulnerability probe is an HTML injection check for a web application. Imagine a web app that has a search box for users to find text within its pages.

Banner grabbing

Banner grabbing is a technique used to gain information about a computer system on a network and the services running on its open ports. Administrators can use this to take inventory of the systems and services on their network.

An intruder can use banner grabbing in order to find network hosts that are running versions of applications and operating systems with known exploits.

This information may be used by an administrator to catalog this system, or by an intruder to narrow down a list of applicable exploits.

To prevent this, network administrators should restrict access to services on their networks and shut down unused or unnecessary services running on network hosts.

Some examples of service ports used for banner grabbing are those used by

Hyper Text Transfer Protocol (HTTP), File Transfer Protocol (FTP), and Simple Mail Transfer Protocol (SMTP); ports 80, 21, and 25 respectively. Tools commonly used to perform banner grabbing are Telnet, nmap, zmap and Netcat.

For example, one could establish a connection to a target web server using Netcat, then send an HTTP request. The response will typically contain information about the service running on the host:

OpenVAS

The Open Vulnerability Assessment System (OpenVAS) collects and manages security information for networks, devices, and systems.

At its core, OpenVAS sweeps through a network to identify known network misconfigurations and known vulnerabilities associated with common services and software.

Vulnerability detections are defined in scripts called Network Vulnerability Tests (NVTs).

OpenVAS uses a client/server architecture to separate the duties of data collection from those of data management. The `openvasd` server (primarily a Linux executable) does the dirty work of keeping track of all of the different vulnerability results against the systems it discovers.

The server uses its own database to manage users independently of the server's host operating system.

OpenVAS

OpenVAS is smart. It uses a variety of probing techniques to recognize services running on any port, rather than just assume a service's identity based on the default Internet Assigned Numbers Authority (IANA) port number. If you have a web server running on TCP port 8888

The OpenVAS user interface displays the aggregated information from all tasks that populate its knowledge base

Metasploit

Vulnerability scanners rely on service banners, version numbers, and network responses to guess whether a particular application or service has a vulnerability that's been publicly reported.

Metasploit (www.metasploit.com) expands on the detection phase by actively exploiting a vulnerability to verify its existence. Not only do the exploits confirm whether or not a vuln exists, but they compose a larger framework that abstracts the hacking process into a sequence of menu options.

It's basically a hacking group at your beck and call. You might say Metasploit dumbs down the hacking process so that anyone who can drive a mouse or tap a keyboard can take over a vulnerable system.

Metasploit is an open source project written in Ruby. Commercial support and extensions are available for it. This section focuses on its open source components.

Metasploit

Metasploit uses the PostgreSQL database (www.postgresql.org) to manage data for scans, sessions, and post-hack information. The database is installed separately from Ruby and Metasploit.

Metasploit uses the PostgreSQL database (www.postgresql.org) to manage data for scans, sessions, and post-hack information. The database is installed separately from Ruby and Metasploit.

Module 2

CYBER SECURITY

Netcat

- Netcat is a wonderfully versatile tool which has been dubbed the “hackers' Swiss army knife”.
- Netcat is a computer networking service for reading from and writing network connections using TCP or UDP.
- This dual functionality suggests that Netcat runs in two modes: “client” and “server”.
- Netcat is designed to be a dependable “back-end” device that can be used candidly or easily driven by other programs and scripts.
- At the same time, it is a feature-rich network debugging and investigation tool, since it can produce almost any kind of correlation you would need and has a number of built-in capabilities.
- Its list of features includes port scanning, transferring files, and port listening, and it can be used as a backdoor.

Netcat

- Major features of Netcat are:
 - Outbound or inbound connections, TCP or UDP, to or from any ports.
 - Full DNS forward/reverse checking, with appropriate warnings.
 - Ability to use any local source port .
 - Ability to use any locally-configured network source address.
 - Built-in port-scanning capabilities, with randomization.
 - Built-in loose source-routing capability.
 - Can read command line arguments from standard input.
 - Hex dump of transmitted and received data.
 - Optional ability to let another program service established connections.
 - Optional telnet-options responder.
 - Featured tunneling mode which allows also special tunneling such as UDP to TCP, with the possibility of specifying all network parameters (source port/interface, listening port/interface, and the remote host allowed to connect to the tunnel).

Netcat

- Netcat can also be used to transfer files from one computer to another. This applies to text and binary files.
- One of Netcat's neat features is command redirection. This means that Netcat can take an exe file and redirect the input, output and error messages to a TCP/UDP port, rather than to the default console.
- Take for example the `cmd.exe` executable. By redirecting the `stdin/stdout/stderr` to the network, we can bind `cmd.exe` to a local port. Anyone connecting to this port will be presented with a command prompt belonging to this computer.
- Another interesting Netcat feature is the ability to send a command shell to a listening host

Netcat

- **-e command** - If Netcat was compiled with the `GAPING_SECURITY_HOLE` option, this option causes a listening Netcat to execute `command` any time someone makes a connection on the port to which it is listening.
- **-g and -G** - Affect loose source routing used to attempt to hide or spoof the source of traffic.
- **-i seconds** - Specifies the delay interval that Netcat waits between sending data.
- **-l** - Toggles Netcat's "listen" mode. This binds Netcat to a local port to await incoming TCP connections, making it act as a server.
-

Netcat

- **-n** Tells Netcat to forego hostname lookups. If you use this option, you must specify an IP address instead of a hostname.
- **-o file** - Dumps communications over this channel to the specified file. The output is written in hexadecimal format. This option records data going in both directions and begins each line with < or > to indicate incoming or outgoing data, respectively.
- **-p port** - Lets you specify the local port number Netcat should use, also referred to as the source port of a connection. For the original Netcat, this argument is required when using the -l or -L option to start listen mode. If it's not specified for outgoing connections, Netcat will use whatever port is given to it by the system, just as most other TCP or UDP clients do. On Unix-based systems, only users with root privileges may specify a port number less than 1024.

Socat

- Socat is a Netcat clone with extensive configuration options. It supports several protocols, from OpenSSL to proxies to IPv4 and IPv6. Its home page is at www.dest-unreach.org/socat/.
- It uses word-based directives on the command line. Socat is part of the BSD ports collection and available as a package for most Linux distributions.

DATA PIPE

- Datapipe is a Unix-based port redirection tool. The original version was written by Todd Vierling in 1995
- A port redirection tool passes TCP/IP traffic received by the tool on one port to another port to which the tool points.
- the tool does not care whether you pass encrypted SSH traffic or plain-text e-mail through it.

DATA PIPE

- **Implementation** - Datapipe in the Unix world are distributable as source code. This enables users to adapt a program to a variety of hardware platforms and Unix versions.
- **Compiling from Source** - You must compile Datapipe for your platform. Often, it is useful for you to have precompiled binaries for several types of Unix: Solaris, AIX, Linux, FreeBSD, OSX, and so on. Use gcc to compile for Linux distributions and the BSD family:
 - `// gcc -o datapipe datapipe.c //`
- **Redirecting Traffic** - Using Datapipe is straightforward in spite of the complicated port redirection tunnels that you can create with it:
 - `// ./datapipe Usage: ./datapipe localhost localport remotehost remote port//`
 - The localhost argument indicates the IP address on which to open the listening port.

DATA PIPE

- The localport argument indicates the listening port on the local system connections will be made to this port number.
- On Unix systems, you must have root privileges to open a listening port below 1024.
- If you receive an error similar to “**bind: Permission denied**,” your account may not have privileges to open a reserved port.
- The remote port argument indicates the port to which data is to be forwarded.
- For example, in most cases if the target is a web server, the remote port value will be 80.
- The remote host argument indicates the hostname or IP address of the target.

DATA PIPE

- The easiest conceptual example of port redirection is forwarding HTTP traffic. Here we set up a datapipe to listen on a high port, 9080 in this example, that redirects to a web site of our choice:
 - `$./datapipe my.host 9080 80 www.google.com`
- Now, we enter this URL into a web browser:
 - `http://my.host:9080/`
 - *You should see Google's home page. By design, Datapipe places itself in the background. So we'll have to use the `ps` and `kill` commands to find the process ID to stop it:*
 - `$ ps auxww | grep datapipe root 21570 0.0 0.1 44 132 ?? Is 8:45PM 0:00.00 ./datapipe my.host 9080 80 ... $ kill -9 21570`

FPipe

- FPipe, from McAfee, implements port redirection techniques natively in Windows.
- It also adds User Datagram Protocol (UDP) support, which Datapipe lacks.
- FPipe is available at
 - www.mcafee.com/us/downloads/free-tools/fpipe.aspx.
- FPipe does not require any support DLLs or privileged user access.
- It runs on all Windows platforms. The lack of support DLLs or similar files makes it easy to pick up fpipe.exe and drop it onto a system.
- FPipe also adds more capability than Datapipe in its ability to use a source port and bind to a specific interface.

FPipe

- Implementation: Whereas Datapipe's options are few, FPipe's increased functionality necessitates some more command-line switches:

FPipe

FPipe Option	Description
-?	Prints the help text.
-h	
-c	Maximum number of simultaneous TCP connections. The default is 32. Note that this has no bearing (and doesn't make sense!) for UDP connections.
-i	The IP address of the listening interface.
-l	The listening port number.
-r	The remote port number (the port to which traffic is redirected).
-s	The source port used for outbound traffic.
-u	UDP mode.
-v	Prints verbose connection information.

- As a port redirector, FPipe works like Datapipe. Here is the Datapipe version:
 - `$./datapipe my.host 9080 80 www.google.com`
- Here's FPipe's equivalent, with connection logs as new clients access the listening port:
 - `C:\> fpipe -l 9080 -r 80 www.google.com Pipe connected: In: 10.0.1.12:57990 --> 10.0.1.5:9080 Out: 10.0.1.5:49433 --> 72.233.2.58:80`
- FPipe does not run as a background process.
- It continues to report connections until you press ctrl-c. Notice that FPipe also indicates the peer IP addresses and the source port number of each connection.
- The `-s` option allows FPipe to take further advantage of port specification:
 - `C:\> fpipe -l 139 -r 139 -s 88 192.168.97.154`

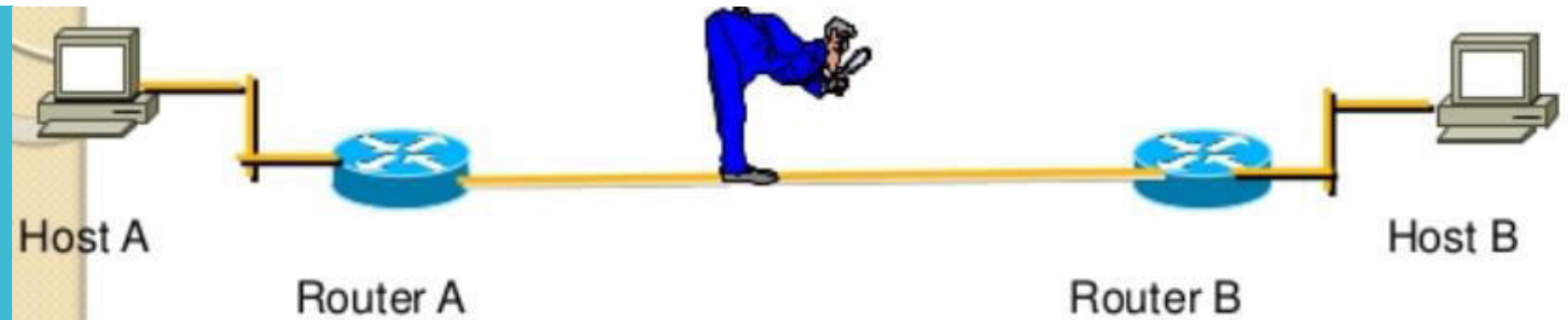
WinRelay

- WinRelay is another Windows-based port redirection tool.
- It and FPipe share the same features, including the ability to define a static source port for redirected traffic.
- Consequently, it can be used interchangeably with FPipe on any Windows platform.
- It is available at www.ntsecurity.nu/toolbox/winrelay/.
- An antivirus or antimalware mechanism may report the WinRelay binary as malicious because it considers this tool's sole purpose (or nearly so) to be part of an exploit kit for compromising systems

Network Sniffing

- Process of capturing, decoding, and analyzing network traffic is called network sniffing.
- Network sniffing is a tool that can help you locate network problems by allowing you to capture and view packet level data on your network.
- The Colasoft Capsa Network sniffer is a powerful network analyzer
- A packet sniffer is a software application that uses a network adapter card in promiscuous mode to capture all network packets

Network Sniffing



Packet sniffing is a technique of monitoring every packet that crosses the network.

Network sniffer is also called as Packet sniffer

Network Sniffing

- A packet analyzer (also known as a packet sniffer) is a piece of software or hardware designed to intercept data as it is transmitted over a network and decode the data into a format that is readable for humans.
- Wireless sniffers are packet analyzers specifically created for capturing data on wireless networks. Wireless sniffers are also commonly referred to as wireless packet sniffers or wireless network sniffers.

Network Sniffing

- Sniffers also work differently depending on the type of network they are in.
 1. Shared Ethernet
 2. Switched Ethernet

Network Sniffing

Packet Sniffer Mitigation



- The following techniques and tools can be used to mitigate sniffers:
 - Authentication—Using strong authentication, such as one-time passwords, is a first option for defense against packet sniffers.
 - Switched infrastructure—Deploy a switched infrastructure to counter the use of packet sniffers in your environment

Network Sniffing

- The following techniques and tools can be used to mitigate sniffers:
 - Antisniffer tools—Use these tools to employ software and hardware designed to detect the use of sniffers on a network.
 - Cryptography—The most effective method for countering packet sniffers does not prevent or detect packet sniffers, but rather renders them irrelevant.

Nmap

- Nmap (Network Mapper) is a security scanner, used to discover hosts and services on a computer network, thus building a "map" of the network.
- To accomplish its goal, Nmap sends specially crafted packets to the target host(s) and then analyzes the responses.
- The software provides a number of features for probing computer networks, including host discovery and service and operating-system detection.

THC-Amap

- THC-Amap, or amap for short, is an advanced port scanner with service identification.
- It probes open ports to determine the listening service's type and, when possible, specific version information.
- This is identical to Nmap's -sV option.
- THC-Amap is available from www.thc.org/thc-amap/. It installs with the usual GNU process (./configure, make, make install) under Unix-based systems, including Cygwin. Note that the web update feature has been disabled, as amap is outdated and not supported anymore.

THC-Amap

- Amap is a great tool for determining what application is listening on a given port.
- It is a scanner that identifies applications/services installed on a remote machine.
- Amap even knows how to parse Nmap output files and it can be used to confirm or complete a Nmap analysis

THC-Amap

- Implementation
 - Amap interrogates ports with various alphanumeric and hexadecimal (i.e., binary) payloads.
 - This interrogation is done after the TCP handshake has been completed. Much of Nmap's port scanning relies on manipulating TCP flags and options that could be spoofed.
 - With Amap, you must interact with the unknown service. Spoofed and decoy traffic is not possible here.

THC-Amap

Mode Option	Description
-A	Identifies the service associated with the port. This identification is based on an analysis of responses to various triggers sent by amap.
-B	Reports banners. Does not perform identification or submit triggers to the service.
-P	Conducts a port scan. Amap performs full connect scans. Use Nmap for advanced options if you just want to discover ports.

TcpDump

- TcpDump is a common packet analyzer that runs under the command line.
- It allows the user to display TCP/IP and other packets being transmitted or received over a network to which the computer is attached.
- TcpDump is free software and it works on most Unix like operating systems.

WinDump

- WinDump is the Windows version of tcpdump, the command line network analyzer for UNIX.
- It is fully compatible with tcpdump and can be used to watch, diagnose and save to disk network traffic according to various complex rules.
- It can run under Windows 95, 98, ME, NT, 2000, XP, 2003 and Vista.

Wireshark

- Wireshark is a free and open source packet analyzer.
- It is used for network troubleshooting, analysis, software and communications protocol development, and education.
- Originally named Ethereal, the project was renamed Wireshark in May 2006 due to trademark issues.
- Wireshark is very similar to tcpdump, but has a graphical frontend, plus some integrated sorting and filtering options.

Ettercap

- Ettercap is a free and open source network security tool for man-in-the-middle attacks on LAN.
- It can be used for computer network protocol analysis and security auditing.
- It runs on various Unix-like operating systems including Linux, Mac OS X, BSD and Solaris, and on Microsoft Windows.
- It is capable of intercepting traffic on a network segment, capturing passwords, and conducting active eavesdropping against a number of common protocols.

hping

- **hping** is a free [packet generator](#) and analyzer for the [TCP/IP](#) protocol distributed by Salvatore Sanfilippo (also known as Antirez).
- It is a one type of a tester for network security. It is one of the *de facto* tools for security auditing and testing of [firewalls](#) and networks, and was used to exploit the [idle scan](#) scanning technique (also invented by the hping author), and now implemented in the [Nmap Security Scanner](#).
- The new version of hping, hping3, is scriptable using the [Tcl](#) language and implements an engine for string based, human readable description of [TCP/IP](#) packets, so that the programmer can write scripts related to low level [TCP/IP](#) packet manipulation and analysis in very short time.

Kismet

- Kismet is a network detector, packet sniffer, and intrusion detection system for 802.11 wireless LANs.
- Kismet will work with any wireless card which supports raw monitoring mode, and can sniff 802.11a, 802.11b, 802.11g, and 802.11n traffic.
- The program runs under Linux, FreeBSD, NetBSD, OpenBSD, and Mac OS X.
- Kismet is used in a number of commercial and open source projects and it is distributed with Kali Linux.

The background features a light gray gradient with several realistic water droplets of varying sizes scattered across the surface. A faint, stylized globe is centered in the background, showing latitude and longitude lines.

CYBER SECURITY MODULE 3 NETWORK DEFENSE TOOLS

FIREWALLS AND PACKET FILTERS-BASICS

What Is a Firewall?

- **Firewalls are either a hardware device or software which deny or accept traffic**
- **It is often built in to devices like wireless access points and cable and DSL modems. It's also a part of almost all operating systems.**
- **At its core, firewall software examines traffic on a network interface to determine whether packets should be allowed to enter or leave the interface.**
- **Thus, firewall software blocks inbound connections to a system's services that shouldn't be exposed to other systems on a public network**
- **It can also be used to block outbound traffic from a system to a network.**
- **For example it block traffic to known malware sites to try and limit the potential damage of downloading an infected file.**
- **Firewalls also manage traffic between two or more different networks.**

CONTD...

- **Firewalls help keep internal traffic internal and safe from malicious external traffic.**
- **Firewalls take the direction of traffic into consideration when filtering packets. It uses two main categories of filters**
- **An **ingress filter** affects packets that arrive on a protected interface (or network, system, etc.). For a firewall that protects a web site, this would be inbound traffic such as HTTP requests from anywhere on the Internet to the web server.**
- **An **egress filter** affects packets that leave the interface. For a web site, this would be responses to incoming HTTP requests.**
- **An ingress filter might ensure that only HTTP traffic comes into the web server. An egress filter might ensure that no traffic is initiated from the web server to the Internet**

CONTD..

- **Two common network security software components that can be part of firewall are**
 - **Personal firewalls**
 - These firewalls primarily protect a system's services or file sharing from unauthorized access.
 - **Parental control software**
 - Parental control software blocks outbound traffic (usually web) to sites excluded from access based on appropriateness (e.g., porn), ideology (e.g., politics), safety (e.g., malware), or other reasons. This requires a privileged account (such as root or Administrator) to define the controls for a lower-privilege account.
- **Other filtering software tools such as spam blockers and virus scanners are similar to firewalls in the sense that they accept or deny traffic based on content inspection.**
- **Spam blockers and virus scanners operate “higher up the stack” on application layer content such as e-mail or web traffic, whereas firewalls typically operate at the level of IP address and port numbers in packet headers.**

CONTD..

- **A packet-level filter might only be able to filter based on source or destination properties (e.g., port 80).**
- **The application layer (or “deep inspection”) firewall might be able to tell the difference between valid and spoofed e-mail.**
- **If encryption is employed, then to most firewalls, an HTTPS connection just looks like random traffic over port 443.**
- **SSL certificate-switching tricks to peek into the encrypted data stream.**

PACKET FILTER VS FIREWALL

- **Packet filters inspect traffic based on characteristics such as protocol, source or destination addresses, and other fields in the TCP/IP (or other protocol) packet header. Firewalls are packet filters, but application layer firewalls may examine more than just packet headers; they may examine packet data (or payloads) as well.**
- **For example, a packet filter may monitor connections to ports 20 and 21 (FTP ports), whereas a firewall may be able to establish criteria based on the FTP port numbers as well as FTP payloads, such as the PORT command or filenames that include the text passwd. A web application firewall (WAF) watches incoming connections for tell-tale signs of SQL injection attacks and outbound traffic for sensitive information being leaked from the web app.**
- **The term packet filter refers to software that makes decisions based on protocol attributes: addresses, ports, and flags. Packet filtering provides coarse (but effective) security to a network routing device. It is simplistic because the access control is limited to a handful of protocols like TCP/IP, UDP, and ICMP.**

CONTD..

- **The term firewall is usually reserved for software or devices whose primary purpose is to apply security decisions to network traffic.**
- **Intrusion-prevention system (IPS). This usually refers to hardware and software that combines packet filtering, content filtering, intrusion-detection system (IDS) capabilities, and other security functions.**

HOW A FIREWALL PROTECTS A NETWORK

- Firewalls are only as effective as the rules they're configured to enforce.
- Most firewalls have three ways to enforce a rule for network traffic:
 - **Accept the packet** and pass it on to its intended destination.
 - **Deny the packet and indicate the denial** with an Internet Control Message Protocol (ICMP) message or similar acknowledgment to the sender. This provides explicit feedback that such traffic is not permitted through the firewall.
 - **Drop the packet without any acknowledgment.** This ends the packet's life on the network. No information is sent to the packet's sender. This method minimizes the sender's ability to deduce information about the protected network, but it may also adversely impact network performance for certain types of traffic. For example, a client may repeatedly attempt to connect to a service because it hasn't received an explicit message that the service isn't available.

CONTD..

- **Most firewalls drop packets as their default policy for traffic that isn't permitted.**
- **When building a ruleset, start with the concept of least privilege or deny all.**

PACKET CHARACTERISTICS TO FILTER

- **Most firewalls and packet filters have the ability to examine the following characteristics of network traffic:**

- **Type of protocol (IP, TCP, UDP, ICMP, IPSec, etc.)**
- **Source IP address and port**
- **Destination IP address and port**
- **ICMP message type and code**
- **TCP flags (ACK, FIN, SYN, etc.)**
- **Network interface on which the packet arrives**

- **To block incoming ping packets (ICMP echo requests) to our home network of 192.168.1.0/24, we can write something like the following rule.**

“deny proto icmp type 8:0 from any to 192.168.1.0/24”

- **The important components of the rule are the action (deny), the packet attributes (ICMP protocol, specifically “ping” types), the direction of the rule (packets “from” one source “to” another), and the type of source (a network address range like 192.168.1.0/24).**

CONTD..

- To allow incoming web traffic to 192.168.1.50 but deny everything else, we can create two rules.

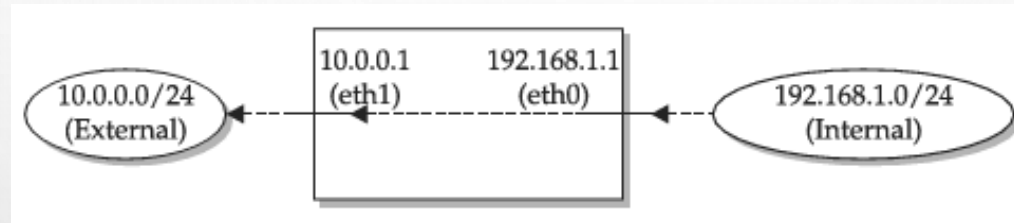
“ allow proto tcp from any:any to 192.168.1.50:80 “

“ deny proto all from any to 192.168.1.0/24 “

- The first one would specify the direction of web traffic to a specific TCP port on a specific host. The second one would make sure all other traffic is denied.
- It is necessary that one should understand the order in which the firewall interprets rules. One firewall may take a **“first match” approach** that permits (or denies) a packet as soon as it encounters a matching rule. A **“last match”** firewall may traverse every rule and apply the final, most specific match to a packet.
- We can also use a firewall to protect our network from IP spoofing.

CONTD..

- For example, imagine our firewall's external interface (called eth1) has an IP address of 10.0.0.1 with a netmask of 255.255.255.0. Our firewall's internal interface (called eth0) has an IP address of 192.168.1.1 with a netmask of 255.255.255.0. Any traffic from the 192.168.1.0 network destined to the 10.0.0.0 network will come in to the eth0 interface and go out of the eth1 interface.



- Conversely, traffic from the 10.0.0.0/24 network destined for the 192.168.1.0/24 network will come in to the eth1 interface and go out of the eth0 interface. Therefore, traffic with a source address in the 192.168.1.0/24 range coming inbound on the eth1 interface should be never seen. If we do, it means someone on the external 10.0.0.0/24 network is attempting to spoof an address in our local IP range. The firewall can stop this kind of activity by using a rule like the following:

“ deny proto any from 192.168.1.0/24 to any on eth1 “

STATELESS VS. STATEFUL FIREWALLS

- A stateless firewall examines individual packets in isolation from each other; it doesn't track whether related packets have arrived before or are coming after.
- A stateful firewall places that packet in the context of related traffic and within a particular protocol, such as TCP/IP or FTP. This enables stateful firewalls to group individual packets together into concepts like connections, sessions, or conversations.
- A stateful firewall is able to filter traffic based not only on a packet's characteristics, but also on the context of a packet according to a session or conversation.
 - For example, a TCP ACK packet will be denied if the protected service hasn't set up the SYN and SYN-ACK handshake to establish a connection
- Stateful firewalls also allow for more dynamic rulesets.

NETWORK ADDRESS TRANSLATION (NAT) AND PORT FORWARDING

- **Networking devices, whether a consumer-level wireless access point or an enterprise-grade firewall, are the gateways between networks. They separate external networks like the Internet from private networks like those used by the systems in our home.**
- **Systems on the Internet must have unique, public (i.e., “routable”) IP addresses. This ensures that packets for a web site or a gaming server always go to the right destination.**
- **Internal networks, on the other hand, use “non-routable” IP addresses, referred to as private or RFC 1918 addresses.**
- **RFC 1918 refers to the document that explicitly defines the address space of the following networks:**
 - **192.168.0.0 through 192.168.255.255 (written 192.168.0.0/16 or 192.168.0.0/255.255.0.0)**
 - **172.16.0.0 through 172.31.255.255 (written 172.16.0.0/12 or 172.16.0.0/255.240.0.0)**
 - **10.0.0.0 through 10.255.255.255 (written 10.0.0.0/8 or 10.0.0.0/255.0.0.0)**

CONTD..

- **The Internet Assigned Numbers Authority (IANA) reserved those IP address blocks for private networks.**
- **This enables organizations large and small to build networks whose traffic will not leak onto the Internet unless it passes through a gateway device like a router or firewall.**
- **Internet traffic should never accommodate packets whose source contains an RFC 1918 address.**
- **It also means that organizations are free to use addresses within these networks without worrying about whether other networks are using the same IP addresses.**
- **IPv4 supports about 4 billion devices theoretically due to its 32-bit address field, but much of that space cannot be used for practical addressing. IPv6 uses a 128-bit address field, enough for roughly 3.4×10^{38} unique devices.**
- **The “non-routable” nature of private address spaces poses a problem once a device needs to access the Internet.**

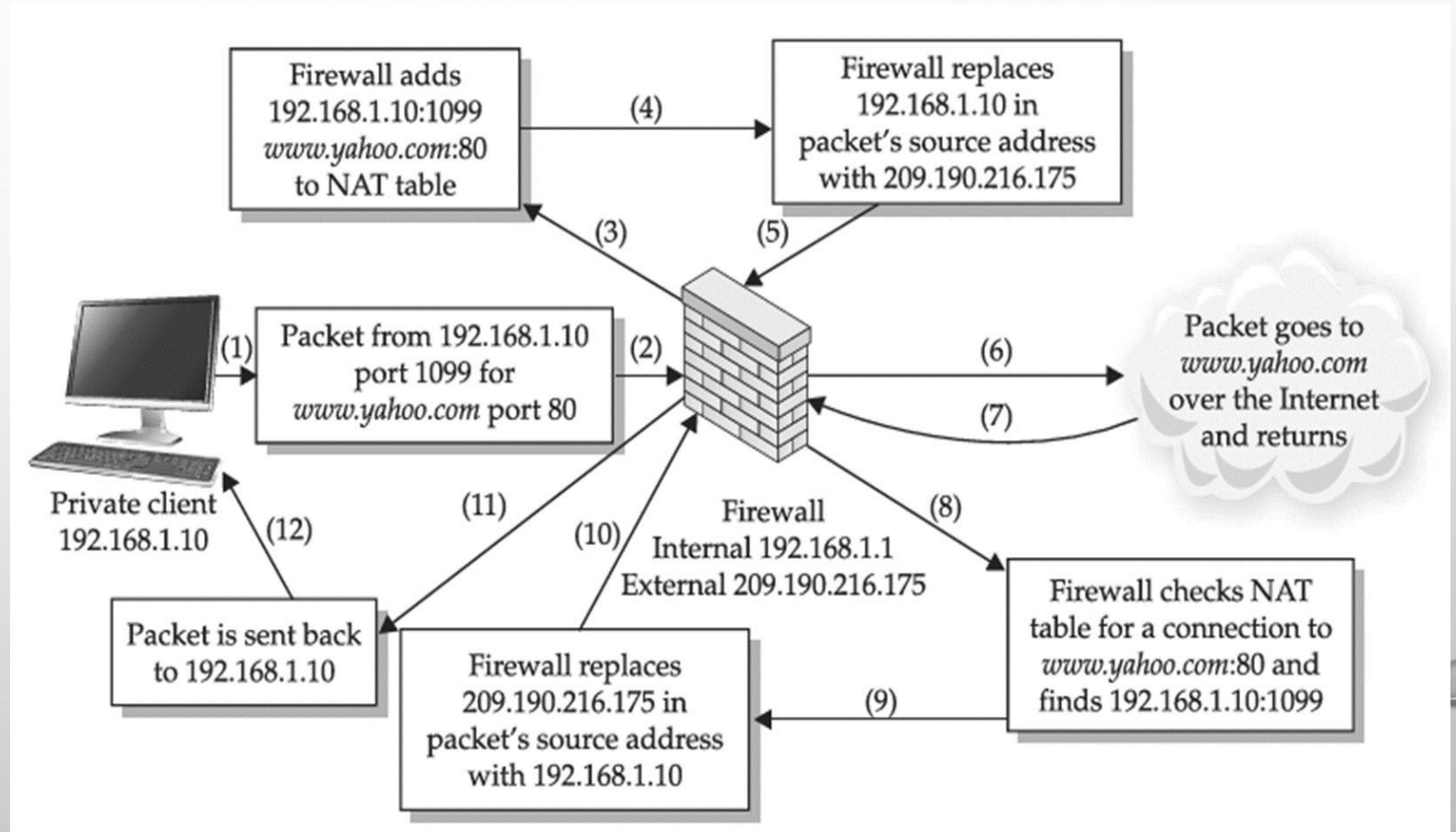
CONTD..

- **The addresses are fine for syncing our data stored on the local network, but they won't work when our device with address 10.0.1.42 needs to retrieve data from storage on the Internet.**
- **The storage service needs to know the difference between our device using the 10.0.1.42 address and someone else's device using the same private IP address on their private network.**
- **Network Address Translation (NAT) solves this routing problem by translating packets from private to public addresses.**
- **NAT is usually performed by a networking device on its external interface for the benefit of the systems on its internal interface.**
- **Private systems can communicate with the Internet using the routable, publicly accessible IP address on the NAT device's external interface.**

CONTD..

- **When a NAT device receives traffic from the private network destined for the external network (Internet), it records the packet's source and destination details. The device then rewrites the packet's header such that the private source IP address is replaced with the device's external, public IP address.**
- **Then the device sends the packet to the destination IP address. From the destination system's point of view, the packet appears to have come directly from the NAT device. The destination system responds as necessary to the packet, sending it back to the NAT device's IP address.**
- **When the NAT device receives the response packet, it checks its address translation table to see if the address and port information of the packet match any of the packets that had been sent out.**
- **If no match is found, the packet is dropped or handled according to any firewall rules operating on the device. If a match is found, the NAT device rewrites the packet's destination IP address with the private IP address of the system that originally sent the packet.**

- Finally, the NAT device sends the packet to its internal destination. The network address translation is completely transparent to the systems on the internal, private IP address and the Internet destination. The private system can access the Internet, but an Internet system cannot directly address it.



CONTD..

- **NAT has a few limitations with regard to the kinds of traffic it may successfully translate. The packet header manipulation will interfere with any protocol that requires the use of true IP addresses, such as IPSec.**
- **Also, any protocols that require a separate, reverse incoming connection, such as active mode FTP, will not work. The outgoing FTP control connection to the FTP server will make it through the NAT device just fine, but when the FTP server attempts to establish the data connection, the NAT device won't know what to do because it doesn't have a corresponding entry in its translation table.**
- **NAT has become integral to firewalls and network security. It provides an added layer of security to a firewall appliance, as it not only protects machines behind its internal interface, but also hides them.**

CONTD..

- **NAT has a few limitations with regard to the kinds of traffic it may successfully translate. The packet header manipulation will interfere with any protocol that requires the use of true IP addresses, such as IPSec.**
- **Also, any protocols that require a separate, reverse incoming connection, such as active mode FTP, will not work. The outgoing FTP control connection to the FTP server will make it through the NAT device just fine, but when the FTP server attempts to establish the data connection, the NAT device won't know what to do because it doesn't have a corresponding entry in its translation table.**
- **NAT has become integral to firewalls and network security. It provides an added layer of security to a firewall appliance, as it not only protects machines behind its internal interface, but also hides them.**
- **if we decide we'd like to expose a particular service on our private network to the Internet ,then we can use a technique called Port forwarding**

CONTD..

- **The NAT device may forward traffic received on a particular port on the device's external interface to a port on a system on the private, internal network. A remote system on the Internet that connects to the NAT device on this port effectively connects to the port on the internal system and only needs to know the public IP address of the NAT device.**
- **Now we've made our private network a little less private by exposing the service listening on that port. Now anyone on the Internet can access our internal web server by connecting to the port on our NAT device.**
- **If our NAT device is a firewall, we can use firewall rules to limit which IP addresses are allowed to access it.**

THE BASICS OF VIRTUAL PRIVATE NETWORKS

- A VPN establishes an encrypted channel between two networks (or single systems, or a combination thereof) that is overlaid on a public network.
- It's designed to mitigate the impact of using a hostile network like a public Wi-Fi connection where data may be sniffed or intercepted by an attacker. The VPN's encrypted traffic is meant to be **opaque to anyone who tries to monitor or interfere** with it. The VPN provides **confidentiality and integrity**.
- A VPN connection usually works like this:
 - Data is transmitted from your client machine to a point in your VPN network. The VPN point encrypts your data and sends it through the internet.
 - Another point in your VPN network decrypts your data and sends it to the appropriate internet resource, such as a web server, an email server, or your company's intranet.
 - Then the internet resource sends data back to a point in your VPN network, where it gets encrypted.
 - That encrypted data is sent through the internet to another point in your VPN network, which decrypts the data and sends it back to your client machine

LINUX SYSTEM FIREWALL

- All Linux distributions rely on the kernel's netfilter software to provide firewall capabilities.
- Netfilter is part of the kernel. The command-line interface for administering netfilter rules is the iptables command. The following example shows what may be the default rules for your system:

```
$ sudo iptables -list
```

```
Chain INPUT (policy ACCEPT)
```

```
Target      prot opt  source      destination
```

```
Chain FORWARD (policy ACCEPT)
```

```
target      prot opt  source      destination
```

```
Chain OUTPUT (policy ACCEPT)
```

```
target      prot opt  source      destination
```

- Netfilter builds tables of rules based on chains. As in the previous example, the iptables command lists the three default chains for netfilter: INPUT, FORWARD, and OUTPUT. These chains reflect the direction of traffic into or out of the network interface monitored by netfilter. The FORWARD chain is a special case for supporting NAT.

WINDOWS SYSTEM FIREWALL

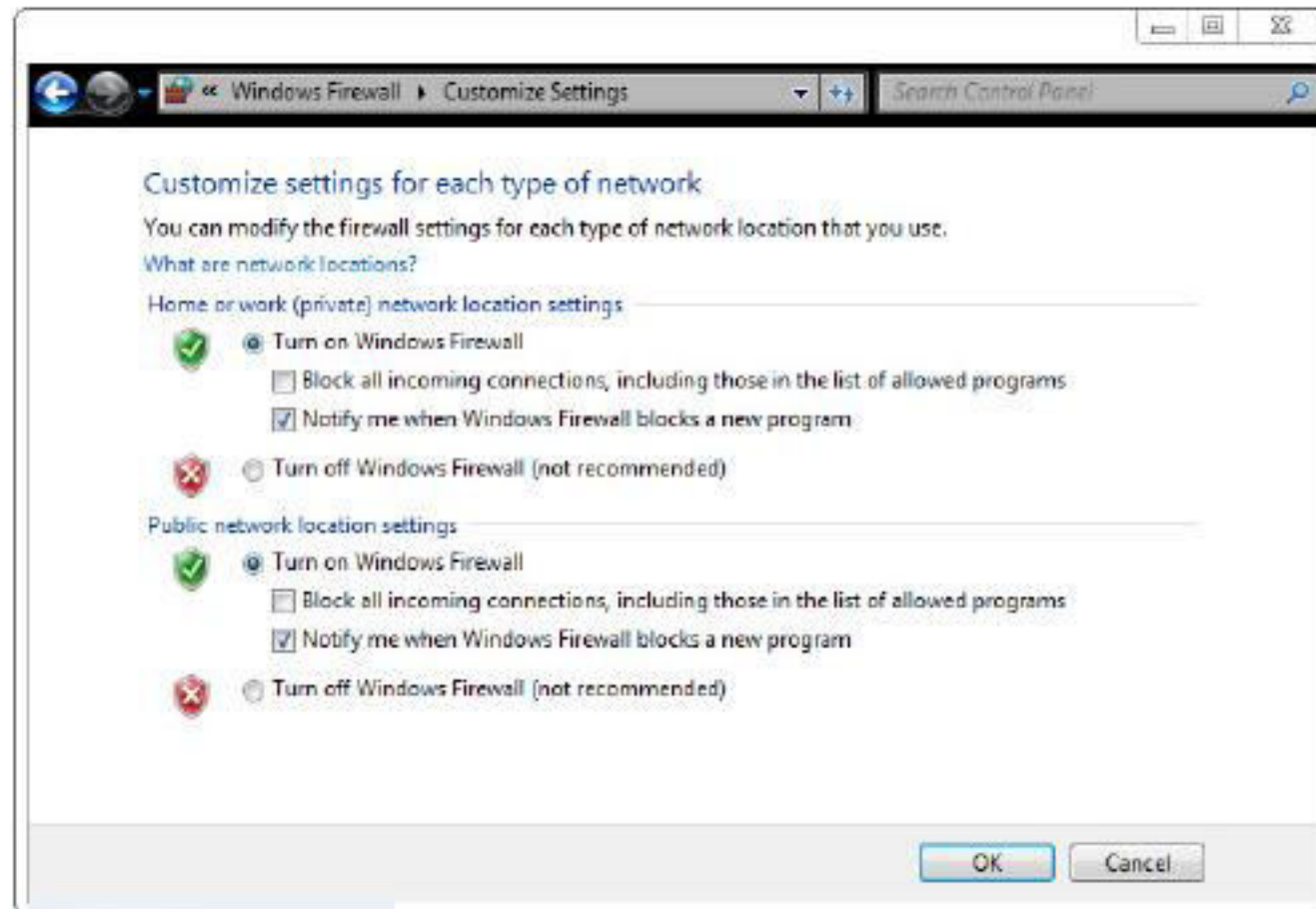
- One important thing Windows OS included is the System Firewall. The interface for enabling and configuring the firewall is in the Control Panel, as shown in Figure



- It allows us to define different trust levels based on your network location

WINDOWS SYSTEM FIREWALL

- In a roaming environment, this interface helps us to determine when and what to share .
- Figure shows the basic options available to you for these different locations.

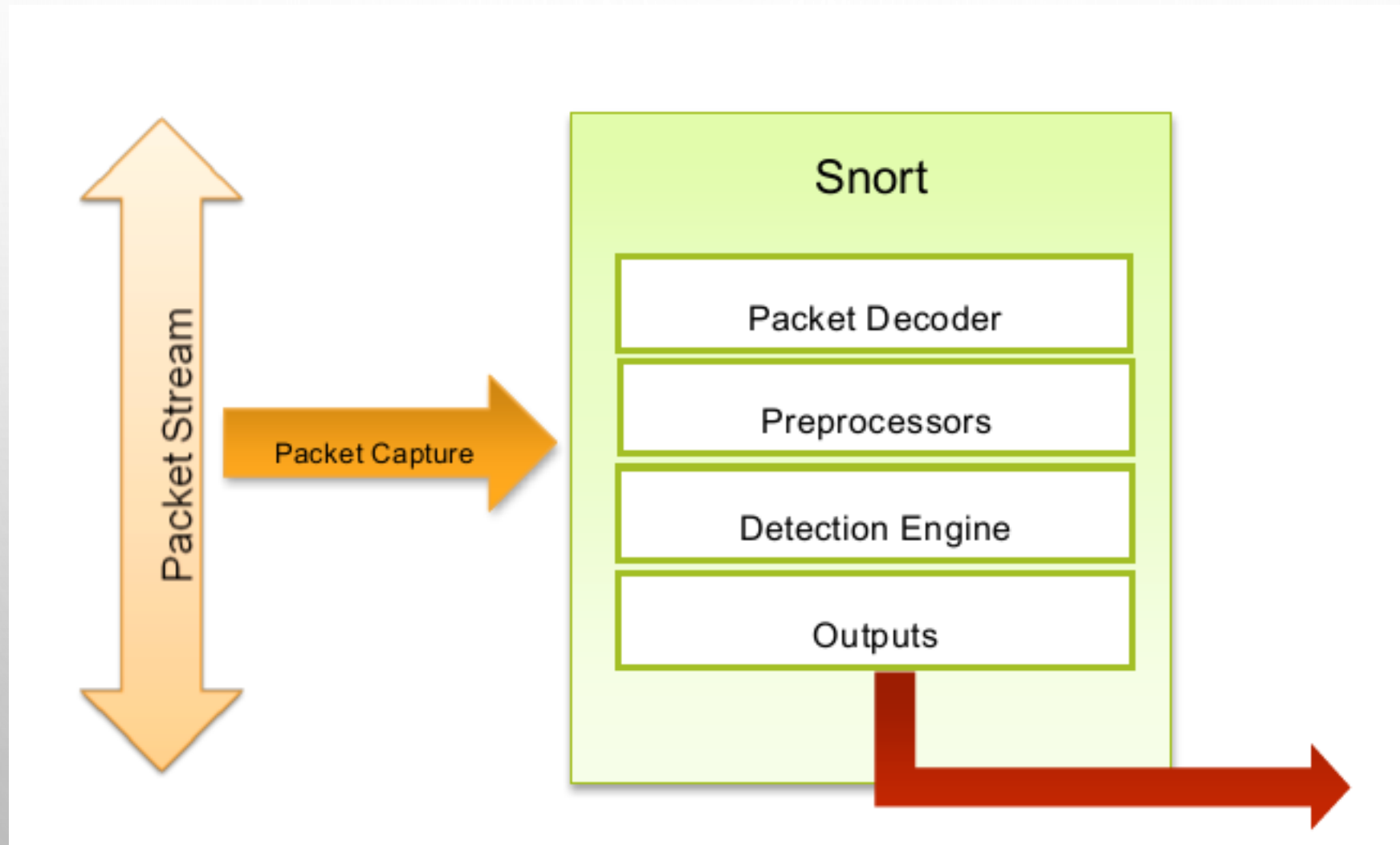


SNORT: AN INTRUSION-DETECTION SYSTEM

SNORT Introduction

- **Snort is a network monitoring tool that watches traffic for signs of malicious activity (e.g., buffer overflows being executed against a service, command and control traffic from malware), suspicious activity (e.g., port scans and service enumeration), and anything else that you wish to look out for.**
- **Snort is a robust IDS that runs on Unix-based and Windows systems. It is also completely free.**
- **It is an open source intrusion prevention system capable of real-time traffic analysis and packet logging.**

SNORT Basic Architecture



SNORT Modes

- Snort can be configured to run in three modes:
- **Sniffer mode:** which simply reads the packets off of the network and displays them for you in a continuous stream on the console (screen).
- **Packet Logger mode:** which logs the packets to disk.
- **Network Intrusion Detection System (NIDS) mode:** which performs detection and analysis on network traffic. This is the most complex and configurable mode.

Exploring Snort.conf

- In the Snort source directory, there are 2 subdirectories of interest: etc and rules. The actual snort.conf file lives in etc.
- The first part of the snort.conf file lets you set some important global variables, indicating such things as your home subnet, your web servers, and your rule locations.

- **Example of some global variable definition :**

```
ipvar HOME_NET [192.168.1.0/24]
ipvar EXTERNAL_NET any
ipvar DNS_SERVERS [192.168.1.150/32,192.168.1.151/32]
ipvar HTTP_SERVERS [192.168.1.42/32]
portvar HTTP_PORTS [80,81,311,591,593,901,1220,1414,1830,2301,2381,2809,
3128,3702,5250,7001,7777,7778028,8080,8088,8118,8123,8180,8181,8243,
8280,8888,9090,9091,9443,9999,11371]
var RULE_PATH ../rules
```

- **These variable definitions tell Snort that it's running on a 192.168.1.0 Class C network, with DNS servers at 192.168.1.150 and 192.168.1.151 and a web server on 192.168.1.42. The Snort ruleset will reference these variables to help cut down on the amount of work it has to do. Rules that check for web attacks will watch only hosts defined in HTTP_SERVERS, instead of watching every host—even those that aren't running web servers. Finally, the RULE_PATH points to the directory containing the actual rule files.**

CONTD..

- The second part of the file lets us configure preprocessors. The preprocessors handle such things as fragmented packets, port scan detection, and stream reassembly.

Snort Rules

- Snort has several types of rules that affect how it handles traffic:
- **Alert rules** Log packets whose characteristics match a predefined suspicious pattern (e.g., generated by a common hacking tool, or contain a string indicative of a buffer overflow or web attack) or custom rules that monitor packets you determine to be prohibited or undesirable on your network (e.g., file sharing, gaming, etc.).
- **Pass rules** Explicitly ignore packets. Traffic that matches these rules will not be logged.
- **Log rules** Record packets but do not generate rules. This would be useful for diagnosing network problems, storing traffic for audits, or monitoring sensitive systems so that traffic can be analyzed in case a compromise is detected.

CONTD..

- **Activate rules** Generate an alert for traffic that matches this rule's trigger, then activate a subsequent dynamic rule. (Until it is activated, a dynamic rule will not generate an alert even if traffic matches it.)
- **Dynamic rules** Triggered by activate rules. This enables you to chain rules together in a way that makes inspection more efficient (don't run rules needlessly) and more effective (create complex chains). These are great mechanisms for gathering more information during an attack.
- Snort comes with a standard ruleset that checks for such activity as Nmap stealth scans, vulnerability exploits, attempted buffer overflows, anonymous FTP access etc.
- By default, Snort checks the packet against alert rules first, followed by pass rules, and then log rules. But using the default setup may produce more false positives.
- So in this scenario, custom defined and tailor-made rule set is necessary .

CONTD..

- **Basic Snort rules consist of two parts: the header and the options. The first part of the header tells Snort what type of rule it is (such as alert, log, pass). The rest of the header indicates the protocol (ip, udp, icmp, or tcp), a directional operator (either -> to specify source to destination or <> to specify bidirectional), and the source and destination IP address and port.**

Preprocessors

- **Preprocessors are set up in the snort.conf file using the preprocessor command. They operate on packets after they've been received and decoded by Snort but before it starts trying to match rules.**

CONTD..

- **Basic Snort rules consist of two parts: the header and the options. The first part of the header tells Snort what type of rule it is (such as alert, log, pass). The rest of the header indicates the protocol (ip, udp, icmp, or tcp), a directional operator (either -> to specify source to destination or <> to specify bidirectional), and the source and destination IP address and port.**

Preprocessors

- **Preprocessors are set up in the snort.conf file using the preprocessor command. They operate on packets after they've been received and decoded by Snort but before it starts trying to match rules.**

Output Modules

- **Output modules are also set up in the snort.conf file using the output command, which controls how, where, and in what format Snort stores the data it receives. Any rule types we define can be specified to use a particular kind of output plug-in.**

- Examples of Preprocessors

Preprocessor	Options	Description
<code>http_inspect</code>	(Lots! Refer to Snort's documentation.)	Provides the capability to examine HTTP traffic. Many aspects of HTTP can be examined, including URLs, parameter strings, cookies, and character encodings. This preprocessor can identify scans from tools like Nikto or Nessus.
<code>sfPortscan</code>	<code>proto <protocol></code> <code>scan_type <scan_type></code> <code>sense_level <n></code> <code>watch_ip <IP [list]></code> <code>ignore_scanners <IP [list]></code> <code>ignore_scanned <IP [list]></code> <code>logfile <file></code>	Supersedes the older <code>Portscan</code> and <code>Flow-Portscan</code> preprocessors. It provides good heuristics for identifying different scan types and methods.
<code>Frag3</code>	<code>max_fragments <n></code> <code>memcap <n></code> <code>prealloc_flags <n></code>	Performs IP packet defragmentation on up to <code>max_fragments</code> fragments, using no more than <code>memcap</code> bytes of memory to prevent resource exhaustion. Defragmentation is often necessary to properly examine packets and TCP sessions.

- Examples of Output modules

Module	Options	Description
alert_fast	<i>file</i>	As with the fast alert mode that can be specified on the command line with <code>-A fast</code> , you can specify a separate file here. Useful if you're defining your own rules and you want some to use the <code>alert_fast</code> module to log to one file while other rules use the <code>alert_fast</code> module to log to another.
alert_full	<i>file</i>	Same as <code>alert_fast</code> , except it uses the default Snort full log mode for alerts.
alert_syslog	<i>facility</i> <i>priority</i>	Similar to the <code>-s</code> option, it allows you to send Snort alert messages directly to syslog using the facility and priority you specify.
alert_unixsock		Establishes a Unix socket from which alerts can be read.

- Examples of Output modules

Module	Options	Description
log_null		Useful when defining rule types when you want to output the alert but don't care about logging the packet data.
log_tcpdump	<i>file</i>	Identical to running Snort in binary logging format (-b) and specifying a different filename for the tcpdump logfile (-L).
alert_unified log_unified	<i>file</i>	An efficient logging method that saves data in binary format. Other programs, such as Barnyard, are required to parse and analyze these files.
Database	<i>rule_type</i> <i>database_type</i> <i>parameters</i>	Logs either Snort log rules or Snort alert rules (depending on <i>rule_type</i>) to an external database. The <i>database_type</i> indicates what kind of SQL database it is (mssql, mysql, postgresql, oracle, odbc) and the parameter list contains necessary information like database host, username and password, database name, and so on.
CSV	<i>file format</i>	Choose from available items to log in the <i>format</i> string and log Snort output into a comma-separated values file named <i>file</i> .



Module 4

WEB APPLICATION TOOLS

Scanning for web vulnerabilities tools

- ▶ Only a few kinds of web servers drive the Web's traffic. Apache HTTP Server is the most recognizable in the open source category, while Microsoft's Internet Information Server (IIS) is the most recognizable commercial one.
- ▶ The web server is the most obvious component of a web application platform; something has to deliver pages to web browsers. But the platform may also comprise data stores, load balancers, and the programming framework used to write pages.
- ▶ We can use a web vulnerability scanner to test the basic security of a web application.
- ▶ A vulnerability scanner contains a knowledge base of all vulns reported for different components of a web platform.

Nikto

- ▶ Nikto is web specific scanner.
- ▶ Developed by Chris Sullo and David Lodge.
- ▶ It is a Perl-based scanner that searches for known vulnerabilities in common web applications, looks for the presence of common files that have the potential to leak information about an application or its platform, and probes a site for indicators of common misconfigurations.
- ▶ The tool focuses on identifying vulns in commercial and open source web application frameworks.
- ▶ It won't be as helpful for assessing the security of a custom web application. For example, it may tell that a site uses an outdated (and insecure) version of WordPress, but it won't be able to tell if the blogging application we wrote from scratch is secure or not.

Implementation

- ▶ Nikto is written in Perl, so it will run on any platform that Perl runs on.

Scanning

- ▶ Nikto is uncomplicated, but not unsophisticated. Use the `-host` option to start scanning a single target for the presence of default files, pages that might expose sensitive information, or pages with known vulnerabilities.
- ▶ The tool requires a target for running.
- ▶ specify the target (`-host` or `-h`: Specifies the target. Use a dash (`-h -`) to take the target name from stdin on the command line. This is useful for typing multiple commands together, such as `nmap: nmap -p80 192.168.0.0/24 -oG - | nikto.pl -h-`):
,
- ▶ specify the port (`-p`: Specifies an arbitrary port. Take care: specifying port 443 does not imply HTTPS. We must remember to include `-ssl`),
- ▶ and record the output to a file (`-output`: Logs output to a file. For example: `-output nikto80_website.html -F htm`).

► Some of the basic options necessary to run Nikto

<code>-Display</code>	Controls the information Nikto reports. For example, <code>-D V</code> displays verbose output. Use <code>-D 2</code> to report the cookies set by the application. This may produce a lot of noise, depending on how the application handles cookies.
<code>-ssl</code>	Forces SSL for the connection, regardless of the port or scheme. Use this if the target expects an HTTPS connection on a nonstandard port. Use <code>-noSSL</code> if you need to disable HTTPS.
<code>-Tuning</code>	Adjusts the types of checks conducted by Nikto. There are currently 13 options, 0–9 and a–c. For example, <code>-T 4</code> restricts the scan to HTML injection checks, <code>-T 9</code> tests for SQL injection, and <code>-T b</code> runs application fingerprinting. See the <code>-Help</code> output for a complete description. The more checks you enable, the longer (and “louder” in terms of triggering monitoring) the scan will be. All checks are enabled by default.
<code>-Format</code>	Records output in a particular format; combine this with the <code>-output</code> option. This helps for consuming the output manually (e.g., as a web page) or passing it to another tool, even Metasploit. You can also imply an output format by assigning a file extension to the <code>-output</code> argument (e.g., <code>-output site.htm</code>). To format the output as a web page or for Metasploit, use the <code>-F</code> option with either the <code>htm</code> or <code>msf</code> value as follows: <code>-F htm</code> <code>-F msf</code>

- Some of the basic options necessary to run Nikto

-vhost	Uses for the target web server a virtual host (vhost) rather than the IP address. This affects the content of the HTTP Host: header. It is important to use this option in shared server environments.
-Cgidirs	Influences how Nikto runs its searches for vulnerable or important CGI directories and files. This disregards 404 errors received for the base directory that would shortcut checks for that “not found” directory. See the upcoming
-mutate	Nikto runs its catalog of files and directories through different permutations in order to discover their presence on a target. Mutated checks are described in more detail in

Nikto Components

- ▶ A line that starts with the # character is ignored. The following example shows some default settings:

```
#CLIOPTS=-g -a
SKIPPORTS=21 111
USERAGENT=Mozilla/5.00 (Nikto/@VERSION) (Evasions:@EVASIONS) (Test:@TESTID)
RFIURL=http://cirt.net/rfiinc.txt?
DEFAULTHTTPVER=1.1
#PROXYHOST=10.1.1.1
#PROXYPORT=8080
#STATIC-COOKIE=cookieName=cookievalue
@@MUTATE=dictionary;subdomains
@@DEFAULT=@@ALL;-@@MUTATE;tests(report:500)
```

- ▶ The CLIOPTS setting contains command-line options to include every time Nikto runs. This is useful for shortening the command line if we always wish to include certain options.

- ▶ The **SKIPPORTS** setting determines whether Nikto will ignore a target if given one of these ports.
- ▶ Modify the **USERAGENT** setting to spoof the header used by a particular browser. This only spoofs the header; it doesn't affect behavior and browser- fingerprinting that a server may attempt against the client.
- ▶ Nikto uses the **RFIURL** to determine if a web page is vulnerable to remote file inclusion. For example, a page might expect to load HTML from a template stored on its own server and use a URL like `http://web.site/index?page=contact.html`. Nikto (or a hacker) could try substituting a link for the `contact.html` page, as in a URL like `http://web.site/index?page=http://cirt.net/rfiinc.txt`. If the web application retrieves and executes the PHP code from the `cirt.net` server, then the application is one step away from being completely compromised.

- ▶ The catch is that every time we run a scan—and every time we find a web site that is vulnerable to an RFI attack—we're signaling its presence in the logs at cirt.net. If we change the link to point to our own page on our own web server, we can check our logs instead.
- ▶ Use the **PROXY*** settings to enable proxy support for Nikto.
- ▶ Although there is rarely a need to change the **DEFAULTHTTPVER** setting, we may find servers that support only version 1.0.
- ▶ The **@@MUTATE** and **@@DEFAULT** values affect which scan databases Nikto will use to search for vulns against the target. The **@@MUTATE** settings greatly increase the time it takes to scan a target because they create different combinations of files and directories in order to find vulnerable resources whose location has been slightly altered from its expected default location.
- ▶ Nikto uses the files in the database subdirectory to determine what kinds of test it performs and how it categorizes responses from a server. The most important file is the **db_dictionary** file that contains a manifest of common directories found on web servers.

w3af

- ▶ w3af (web application attack and audit framework) is an open-source web application security scanner. The project provides a vulnerability scanner and exploitation tool for Web applications. It provides information about security vulnerabilities for use in penetration testing engagements. The scanner offers a graphical user interface and a command-line interface.
- ▶ w3af is divided into two main parts, the core and the plug-ins. The core coordinates the process and provides features that are consumed by the plug-ins, which find the vulnerabilities and exploit them. The plug-ins are connected and share information with each other using a knowledge base.
- ▶ Plug-ins can be categorized as Discovery, Audit, Grep, Attack, Output, Mangle, Evasion or Bruteforce.
- ▶ w3af was started by Andres Riancho in March 2007
- ▶ It is a software that will identify vulnerabilities in web applications by sending specially crafted HTTP requests to it.
- ▶ The framework work on all Python supported platforms. It supports mainly LINUX Oses. But it can be installed on Windows OS also.

Main plugin types

- ▶ The framework has three main plugins types: crawl, audit and attack
- ▶ Crawl plugins
 - ▶ They have only one responsibility, finding new URLs, forms, and other injection points. A classic example of a discovery plugin is the web spider. This plugin takes a URL as input and returns one or more injection points. When a user enables more than one plugin of this type, they are run in a loop: If plugin A finds a new URL in the first run, the w3af core will send that URL to plugin B. If plugin B then finds a new URL, it will be sent to plugin A. This process will go on until all plugins have run and no more information about the application can be found.
- ▶ Audit plugins
 - ▶ Take the injection points found by crawl plugins and send specially crafted data to all in order to identify vulnerabilities. A classic example of an audit plugin is one that searches for SQL injection vulnerabilities by sending a'b"c to all injection points.
- ▶ Attack plugins
 - ▶ Their objective is to exploit vulnerabilities found by audit plugins. They usually return a shell on the remote server, or a dump of remote tables in the case of SQL injection exploits.

Other plugins

► Infrastructure

- Identify information about the target system such as installed WAF (web application firewalls), operating system and HTTP daemon.

► Grep

- Analyze HTTP requests and responses which are sent by other plugins and identify vulnerabilities. For example, a grep plugin will find a comment in the HTML body that has the word “password” and generate a vulnerability.

► Output

- The way the framework and plugins communicate with the user. Output plugins save the data to a text, xml or html file. Debugging information is also sent to the output plugins and can be saved for analysis

► Mangle

- Allow modification of requests and responses based on regular expressions, think “sed (stream editor) for the web”.

Other plugins

▶ Bruteforce

- ▶ Bruteforce logins found during the crawl phase.

▶ Evasion

- ▶ Evade simple intrusion detection rules by modifying the HTTP traffic generated by other plugins.

▶ Scan configuration

- ▶ After configuring the crawl and audit plugins, and setting the target URL the user starts the scan and waits for the vulnerabilities to appear in the user interface.
- ▶ Any vulnerabilities which are found during the scan phase are stored in a knowledge base; which is used as the input for the attack plugins. Once the scan finishes the user will be able to execute the attack plugins on the identified vulnerabilities.
- ▶ In most cases it is recommend to run w3af with the following configuration:
 - ▶ crawl: web_spider
 - ▶ audit: Enable all
 - ▶ grep: Enable all

HTTP Utilities

Stunnel

- ▶ There are situations in which the client sends out HTTPS connections and cannot be downgraded to HTTP. In these cases, you need a tool that can either decrypt SSL or sit between the client and server and watch traffic in clear text. Stunnel provides this functionality.
- ▶ Stunnel is a proxy designed to add TLS (Transport Layer Security) encryption functionality to existing clients and servers without any changes in the programs' code. Its architecture is optimized for security, portability, and scalability (including load-balancing), making it suitable for large deployments.
- ▶ Stunnel is a free software authored by Michał Trojnara.
- ▶ Stunnel uses the OpenSSL library for cryptography, so it supports whatever cryptographic algorithms are compiled into the library
- ▶ SSL communications rely on certificates. The first thing you need is a valid [PEM file](#) that contains encryption keys to use for the communications. Stunnel comes with a default file called stunnel.pem, which it lets you define at compile time.

- ▶ One use of stunnel is to intercept traffic by downgrading client connections from HTTPS to HTTP, inspect or manipulate the traffic, and then upgrade the connection back from HTTP to HTTPS for the server. The concept is similar to using an interactive proxy to be able to view the plaintext form of HTTPS traffic.
- ▶ Run stunnel in normal daemon mode (-d). This mode accepts SSL traffic and outputs traffic in clear text. The -f option forces stunnel to remain in the foreground. This is useful for watching connection information and making sure the program is working. Stunnel is not an end-point program.
- ▶ In other words, we need to specify a port on which the program listens (-d port) and a host and port to which traffic is forwarded (-r host:port)
- ▶ Run stunnel in client mode with the -c option to accept plaintext traffic and forward it over an SSL/TLS connection to a remote (-r) host.
- ▶ Stunnel is a robust way to wrap SSL/TLS protection around an otherwise unencrypted service. Use the -l option to specify the full path to a service daemon.

- ▶ Most services natively support SSL/TLS connections. This is more useful for setting up redirects in order to inspect traffic between a client and server.
- ▶ For example: Some clients either don't provide HTTP proxy settings (otherwise you could use a tool like the Zed Attack Proxy discussed a bit later) or run some protocol other than HTTP over the SSL/TLS connection. In these cases, it's necessary to use host spoofing tricks and redirection so that you can "downgrade" the client's connection from SSL/TLS in order to manipulate it, then "upgrade" the connection back to SSL/TLS when sending traffic on to the server.

Password Cracking and Brute-Force Tools

John the Ripper

- ▶ John the Ripper remains one of the fastest, most versatile, and most popular password crackers available. It supports password hashing schemes used by many systems, including most Unix-based systems (like OpenBSD and various Linux distributions) and the various Windows hashes, as well as proprietary password hashing functions used by several database and software packages for user account management. John's cracking modes include specialized wordlists, the ability to customize the generation of guesses based on character type and placement (useful when targeting a specific password policy), raw brute force, and statistically guided brute force that uses successfully cracked passwords to influence future guesses.
- ▶ John runs on just about any operating system.

Implementation

- ▶ Obtain and Compile John
- ▶ John has hard-coded many compilation flags and optimization settings for dozens of specific operating systems and CPU architectures.
- ▶ The following commands would compile John under OS X, Cygwin, and FreeBSD:
 - ▶ `$ make macosx-x86-64`
 - ▶ `$ make win32-cygwin-x86-sse2`
 - ▶ `$ make freebsd-x86-64`
- ▶ The make step configures and compiles John for our platform. When this step has finished, the binaries and configuration files will be placed in the `./run` directory relative to the `./src` directory in which you executed the make command.
- ▶ If it has installed correctly, then we can run John.
- ▶ Now verify that John works by generating a baseline cracking speed for our

Cracking Passwords

- ▶ John automatically recognizes common password formats extracted from operating system files like /etc/ shadow or dumped by tools like pwdump. In practice, John supports close to 150 different hashing algorithms
- ▶ The following example shows John's ability to guess the correct format for password entries.

- ▶ First, create a text file named windows.txt with the following two lines containing an entry for "Ged" and "Arha." They represent passwords taken from a Windows system.

```
Ged:1006:NO PASSWORD*****:FB9C381BD729E7A93C14EBAFBA9B78DE:::
```

```
Arha:1007:NO PASSWORD*****:2C5F5597333BD214B5BEA2C01C591BC9:::
```

- ▶ Next, run John against the windows.txt file:

```
$ ./john windows.txt
```

```
Warning: detected hash type "nt", but the string is also recognized as "nt2"
```

```
Use the "--format=nt2" option to force loading these as that type instead
```

```
Loaded 2 password hashes with no different salts (NT MD4 [128/128 X2 SSE2-16])
```

```
Tenar (Arha)
```

- ▶ The brute-force attack should very quickly discover that “Tenar” is the password for the Arha account. It will take much longer to guess the Ged account’s password unless we try some refinements to the brute-force approach.
- ▶ In the example, John recommended that we use the `--format=nt2` option to explicitly define which hash algorithm to target with the cracker. If the format isn’t evident, or John misinterprets the format of the target file, use that option to correct it. We can obtain all formats supported by John with the `--list` option, as follows:

```
$ ./john --list=formats
```

```
...
```

```
$ ./john --list=format-all-details
```

- ▶ To make effective and to reduce time consuming in password cracking using John, expend the resources effectively
- ▶ We can try to improve the power of the brute-force attack by optimizing the implementation of algorithms, using faster CPUs, using customized processors, distributing the work, etc. to attain higher cracks per second

- ▶ We can try to improve the efficiency of the attack by guiding the sequence of guesses or choosing dictionaries that are statistically more likely to match the kinds of passwords humans create.
- ▶ One password should have been cracked so far. We use the --show option to list it:

```
$ ./john --show windows.txt
```

```
Arha:Tenar:NO PASSWORD*****:2C5F5597333BD214B5BEA2C01C591BC9:::
```

```
1 password hash cracked, 1 left
```

- ▶ John keeps track of all passwords it has ever cracked in a john.pot file by default. For example, here's what ours currently looks like:

```
$ cat john.pot
```

```
$NT$2c5f5597333bd214b5bea2c01c591bc9:Tenar
```

- ▶ Use the --pot option to specify alternate files to store (or read) cracked passwords from.

- ▶ From an efficiency perspective, we can try different wordlists (aka dictionaries) of common passwords against our unknown hashes. The measure of “common” may be based on past successful cracks, actual dictionaries, or popular terms from media. Use the `--wordlist` option to try a (relatively) quick pass against the hashes. John provides a single dictionary, `password.lst`, with its distribution. We can find more, larger dictionaries on the John the Ripper web site.

Incremental Mode Cracking

- ▶ John’s incremental mode uses “charset” files and `john.conf` directives to control what kinds of guesses it performs and therefore how many guesses and how long the guesses will take to complete.
- ▶ John comes with several predefined incremental modes.
- ▶ John’s incremental mode tries all eventual permutations of a charset file
- ▶ Incremental mode is guaranteed to guess every combination at the expense of taking a very, very long time to complete.

- ▶ By default, the mode tries all combinations between one and eight characters long.
- ▶ If we want to target a specific length, we can edit the john.conf file to add a new incremental mode.
- ▶ John builds the charset file with statistical properties from an input file that contains the target characters.

Markov Mode Cracking

- ▶ One of John's improvements over time is its adoption of cracking techniques that rely on the statistical composition of cracked passwords to guide the generation of new guesses.
- ▶ Its Markov mode tries a limited set of permutations based on a "stats" file.
- ▶ Markov mode trades completeness for speed; it tries guesses that are very close to known passwords under the assumption that humans choose passwords based on habit or identifiable patterns.
- ▶ Use the --markov option to start this mode against a password file.

Contd..

- ▶ Markov mode is most useful when targeting long passwords. For example, trying to brute force a 19-character password composed from a pool of 96 characters is roughly equivalent to brute-forcing a 125-bit encryption algorithm.
- ▶ In order to use Markov mode against long passwords, you need to provide the `calc_stat` command with a source of words of the same size.

Pwdump-Grabbing Windows Password Hashes

- ▶ The original pwdump program was written by Jeremy Allison in 1997 to demonstrate how to extract password hashes from the Windows Registry.
- ▶ Till then there are a number of versions available. But they all rely on extracting hashes from the Registry, SAM file, or the lsass.exe process's memory space. The lsass.exe process handles the Local Security Subsystem Service; it's essentially responsible for authentication, which is why its memory contains the system's password hashes.

Pwdump6

- ▶ The pwdump tools are simple to use. They require Administrator privileges, so we will need to start the cmd.exe shell with Run As Administrator.
- ▶ The following example demonstrates pwdump6 on a 64-bit Windows system. The -x option is necessary to let pwdump6 know the target system is 64-bit. Otherwise, the process will hang without returning results. The -n option instructs pwdump6 to forego the search for password histories. The output may be passed to John the Ripper in order to start cracking hashes.

```
C:\pwdump6\PwDumpRelease> PwDump.exe -n -x localhost
Administrator:500:NO PASSWORD*****:NO
PASSWORD*****:..
Arha:1007:NO PASSWORD*****:2C5F5597333BD214B5BEA2C01C591BC9::
Ged:1006:NO PASSWORD*****:FB9C381BD729E7A93C14EBAFBA9B78DE::
Guest:501:NO PASSWORD*****:NO PASSWORD*****:..
Completed.
```

- ▶ Pwdump6 supports remote enumeration provided you have Administrator access to the target's network shares

Pwdump7

- ▶ Pwdump7 is hardly any different from pwdump6 in terms of execution. Its command- line options enable us to specify specific source files from which to extract hashes.
- ▶ It does not support remote access to a target.

THC-Hydra

- ▶ **THC-Hydra (aka simply Hydra) easily surpasses the majority of brute-force tools available on the Internet for two reasons: it is fast, and it targets authentication mechanisms for several dozen protocols.**
- ▶ **When we need to brute force crack a remote authentication service, Hydra is often the tool of choice. It can perform rapid dictionary attacks against more than 50 protocols, including telnet, ftp, http, https, smb, several databases, and much more**

Implementation

- ▶ **Hydra compiles on BSD and Linux systems without a problem; The software can be used under Windows through the Cygwin environment. Follow the usual ./configure, make, make install method for compiling source code.**

► The command-line arguments

Hydra Option	Description
-R	Restores a previous aborted / crashed session from the hydra .restore file (by default this file is created in the directory from which Hydra was executed).
-S	Connects via SSL.
-s <i>n</i>	Connects to port <i>n</i> instead of the service's default port.
-l <i>name</i>	Uses <i>name</i> from the command line or from each line of <i>file</i> as the username portion of the credential.
-L <i>file</i>	
-p <i>password</i>	Uses <i>password</i> from the command line or from each line of <i>file</i> as the password portion of the credential.
-P <i>file</i>	
-C <i>file</i>	Loads user:password combinations from <i>file</i> . Each line contains one combination separated by a colon.

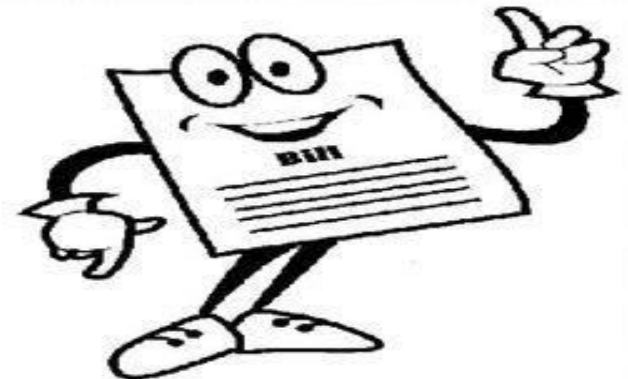
Hydra Option	Description
<code>-e nsr</code>	Also tests the login prompt for a null password (n), a password equal to the username (s), or a password of the login name reversed (r).
<code>-M file</code>	Targets the hosts listed in each line of <i>file</i> instead of a single host.
<code>-o file</code>	Writes a successful username and password combination to <i>file</i> instead of stdout.
<code>-f</code>	Exits after the first successful username and password combination is discovered for the host. If multiple hosts are targeted (-M), then Hydra will continue to run against other hosts until the first successful credentials are found.
<code>-t n</code>	Executes <i>n</i> parallel connects to the target service. The default is 16. The performance gain from this option is affected by both your system's resources and the target's resources.
<code>-w n</code>	Waits no more than <i>n</i> seconds for a response from the service before assuming no response will come.
<code>-v</code>	Reports verbose status information.
<code>-V</code>	
<code>-4</code>	Connects over IPv4 (-4) or IPv6 (-6).
<code>-6</code>	
<code>server</code>	Specifies the target's IP address or hostname. For multiple targets, use the -M option to load targets from a text file (with each target on a single line).
<code>service</code>	Specifies the target's service to brute force.

MODULE 5

INTRODUCTION TO CYBER CRIME AND LAW

1.1 INTRODUCTION

- ❖ The internet in India is growing rapidly. It has given rise to new opportunities in every field we can think of be it entertainment, business, sports or education.
- ❖ There're two sides to a coin. Internet also has it's own disadvantages is Cyber crime- illegal activity committed on the internet.



1.2 DEFINING CYBER CRIME

- Crime committed using a computer and the internet to steal data or information.
- Illegal imports.
- Malicious programs.



Alternative definitions for cybercrime



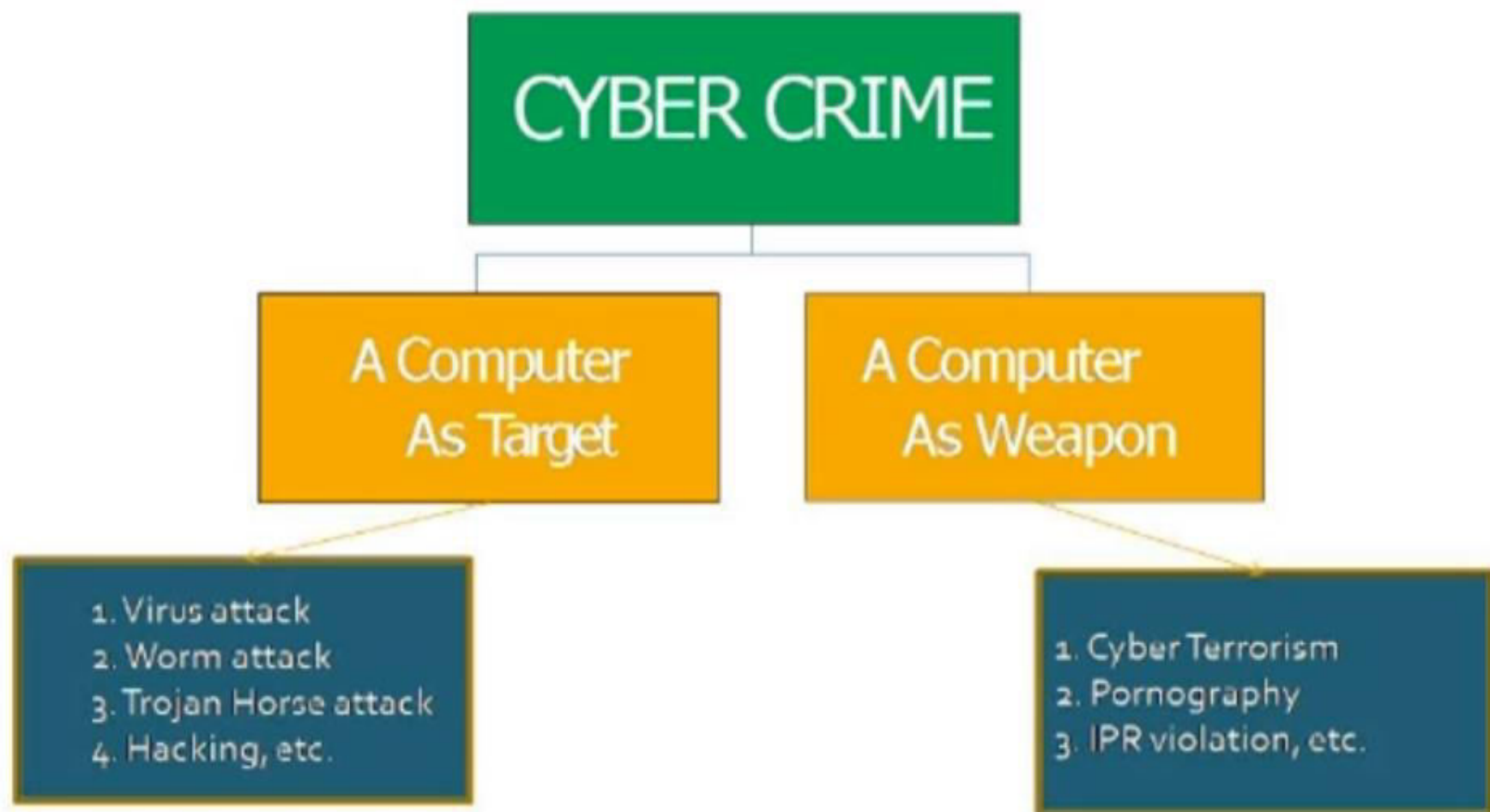
- Any illegal act where a special knowledge of computer technology is essential for its perpetration, investigation or prosecution
- Any traditional crime that has acquired a new dimension or order of magnitude through the aid of a computer, and abuses that have come into being because of computers
- Any financial dishonesty that takes place in a computer environment.
- Any threats to the computer itself, such as theft of hardware or software, sabotage and demands for ransom

CYBER CRIME

- Unlawful acts done by using the computer either as a tool or target.
- Criminal activities done by using computers, Internet etc.



CLASSIFICATION OF CYBER CRIME



Another definition



- “Cybercrime (computer crime) is any illegal behavior, directed by means of electronic operations, that target the security of computer systems and the data processed by them”.
- Hence cybercrime can sometimes be called as computer-related crime, computer crime, E-crime, Internet crime, High-tech crime....



Cybercrime specifically can be defined in number of ways...



- A crime committed using a computer and the internet to steal a person's identity(identity theft) or sell contraband or stalk victims or disrupt operations with malevolent programs.
- Crimes completed either on or with a computer
- Any illegal activity through the Internet or on the computer.
- All criminal activities done using the medium of computers, the Internet, cyberspace and the WWW.

further

- Cybercrime refers to the act of performing a criminal act using cyberspace as communication vehicle.
- Two types of attacks are common:
 - **Techno- crime : Active attack**
 - Techno Crime is the term used by law enforcement agencies to denote criminal activity which uses (computer) technology, not as a tool to commit the crime, but as the subject of the crime itself. Techno Crime is usually pre-meditated and results in the *deletion, corruption, alteration, theft or copying of data on an organization's systems*.
 - Techno Criminals will usually probe their prey system for weaknesses and will almost always leave an electronic 'calling card' to ensure that their pseudonym identity is known.
 - **Techno – vandalism: Passive attack**
 - Techno Vandalism is a term used to describe a *hacker or cracker* who breaks into a computer system with the *sole intent of defacing and or destroying its contents*.
 - Techno Vandals can deploy '*sniffers*' on the Internet to locate soft (insecure) targets and then execute a range of commands using a variety of protocols towards a range of ports. If this sounds complex - it is! The best weapon against such attacks is a firewall which will hide and disguise your organization's presence on the Internet.

1.3 Cybercrime and information security

- Lack of information security give rise to cybercrime
- **Cybersecurity:** means protecting information, equipment, devices, computer, computer resource, communication device and information stored therein from unauthorized access, use, disclosure, disruption, modification or destruction.



1.4 Who are Cybercriminals?

- Are those who conduct acts such as:
 - Child pornography
 - Credit card fraud
 - Cyberstalking
 - Defaming another online
 - Gaining unauthorized access to computer systems
 - Ignoring copyrights
 - Software licensing and trademark protection
 - Overriding encryption to make illegal copies
 - Software piracy
 - Stealing another's identity to perform criminal acts



Categorization of Cybercriminals

- **Type 1: Cybercriminals- hungry for recognition**
 - Hobby hackers
 - A person who enjoys exploring the limits of what is possible, in a spirit of playful cleverness. May modify hardware/ software
 - IT professional(social engineering):
 - Ethical hacker
 - Politically motivated hackers :
 - promotes the objectives of individuals, groups or nations supporting a variety of causes such as : Anti globalization, transnational conflicts and protest
 - Terrorist organizations
 - Cyberterrorism
 - Use the internet attacks in terrorist activity
 - Large scale disruption of computer networks , personal computers attached to internet via viruses



Type 2: Cybercriminals- not interested in recognition

- Psychological perverts
 - Express sexual desires, deviates from normal behavior
 - Poonam panday
- Financially motivated hackers
 - Make money from cyber attacks
 - Bots-for-hire : fraud through phishing, information theft, spam and extortion
- State-sponsored hacking
 - Hacktivists
 - Extremely professional groups working for governments
 - Have ability to worm into the networks of the media, major corporations, defense departments

Type 3: Cybercriminals- the insiders

- Disgruntled or former employees seeking revenge
- Competing companies using employees to gain economic advantage through damage and/ or theft.

Motives behind cybercrime

- Greed
- Desire to gain power
- Publicity
- Desire for revenge
- A sense of adventure
- Looking for thrill to access forbidden information
- Destructive mindset
- Desire to sell network security services



Hacking

Every act committed toward breaking into a computer and/ or network is hacking.

Purpose

- Greed
- Power
- Publicity
- Revenge
- Adventure
- Desire to access forbidden information
- Destructive mindset

HACKING

- Unauthorised access to private computers.
- Hacking is done by Hackers.

- കമ്പ്യൂട്ടറുകളിലേക്കുള്ള അനധികൃത ആക്സ്.



TYPES OF HACKERS

a. White Hat Hackers

- White hat hackers, also known as ethical hackers.
- Hacking done for good purpose.
- The terminology used by the White Hat Hackers to prevent cyber attacks is kill switch.
- നല്ല ലക്ഷ്യത്തിനായി ഹാക്ക്‌ക്വിംഗ് ചെയ്യുന്നു

TYPES OF HACKERS

b. BLACK HAT

- Hacking done for criminal purpose.

- കുറ്റകരമായ ഉദ്ദേശ്യലക്ഷ്യങ്ങൾക്കായി ഹാക്ക് ചെയ്യുന്നു

c. GREY HAT

- They do hacking, sometimes for good purpose sometimes for bad purpose.

- ചിലപ്പോൾ നല്ല ഉദ്ദേശ്യത്തിനായി ചിലപ്പോൾ മോശമായ ലക്ഷ്യത്തിനായി ഹാക്കിംഗ് ചെയ്യുന്നു.

TYPES OF HACKERS

d. Neophyte:

- someone who is new to **hacking**.
- ആദ്യമായി ഹാക്ക് ചെയ്യാൻ തുടങ്ങുന്നവർ.

e. Hacktivist

- Hacking done for spreading ideological messages. (social, religious & political, etc.)..
- ഒരു ആശയപ്രചാരണത്തിനായി ഹാക്ക് ചെയ്യുന്നു.

1.5 Classification of cybercrimes

1. Cybercrime against an individual
2. Cybercrime against property
3. Cybercrime against organization
4. Cybercrime against Society
5. Crimes emanating from Usenet newsgroup

1. Cybercrime against an individual

- Electronic mail spoofing and other online frauds
- Phishing, spear phishing
- spamming
- Cyberdefamation
- Cyberstalking and harassment
- Computer sabotage
- Pornographic offenses
- passwordsniffing

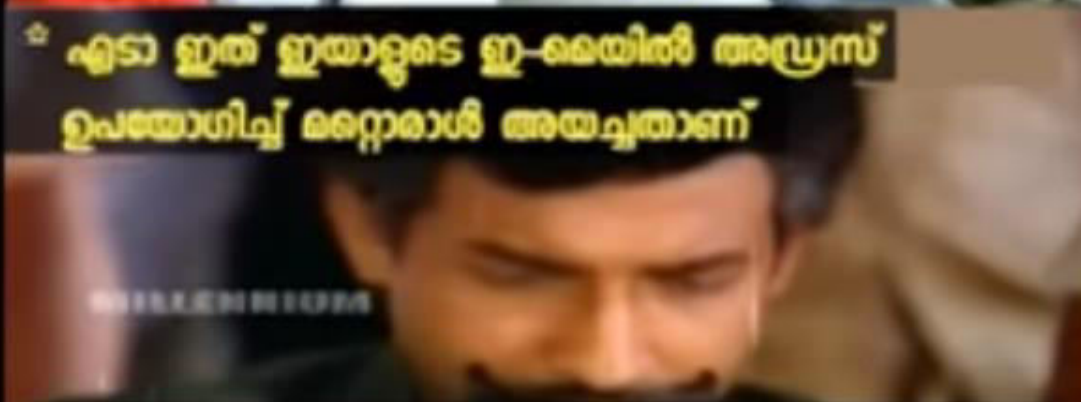
Online frauds



- Fraud that is committed using the internet is “online fraud.” Online fraud can involve financial fraud and identity theft.
- Online fraud comes in many forms.
 - viruses that attack computers with the goal of retrieving personal information, to email schemes that lure victims into wiring money to fraudulent sources,
 - “phishing” emails that purport to be from official entities (such as banks or the Internal Revenue Service) that solicit personal information from victims to be used to commit identity theft,
 - to fraud on online auction sites (such as Ebay) where perpetrators sell fictional goods.
 - E-Mail spoofing to make the user to enter the personal information : financial fraud
 - Illegal intrusion: log-in to a computer illegally by having previously obtained actual password. Creates a new identity fooling the computer that the hacker is the genuine operator. Hacker commits innumerable number of frauds.

E-Mail Spoofing

- E-mail spoofing is the forgery of an e-mail header so that the message appears to have originated from someone or somewhere other than the actual source.
- To send spoofed e-mail, senders insert commands in headers that will alter message information.
- It is possible to send a message that appears to be from anyone, anywhere, saying whatever the sender wants it to say.
- Thus, someone could send spoofed e-mail that appears to be from you with a message that you didn't write.
- Classic examples of senders who might prefer to disguise the source of the e-mail include a sender reporting mistreatment by a spouse to a welfare agency.



Email spoofing : ഇക്കെയിൽ ഔദ്യോഗിക അഡ്രസ്സിൽ നിന്ന് അയയ്ക്കുന്നതായി തോന്നുകയും അത് യഥാർത്ഥത്തിൽ ഒറ്റത്തവണയായി നിന്ന് അയച്ചതാണ്.

EMAIL SPOOFING

- The email appears to have been originated from official address but it is actually sent from another source.
- ഇമെയിൽ ഔദ്യോഗിക അഡ്രസ്സിൽ നിന്ന് അയയ്ക്കുന്നതായി തോന്നുമെങ്കിലും അത് യഥാർത്ഥത്തിൽ മറ്റൊരു സ്രോതസ്സിൽ നിന്ന് അയച്ചതാണ്.



Spamming

- People who create electronic spam : **spammers**
- **Spam** is abuse of electronic messaging systems to send unsolicited bulk messages indiscriminately
- Spamming may be
 - E-Mail Spam
 - Instant messaging spam
 - Usenet group spam
 - Web search engine spam
 - Spam in blogs, wiki spam
 - Online classified ads spam
 - Mobile phone messaging spam
 - Internet forum spam
 - Junk fax spam
 - Social networking spam
 -

☆ ചോട്ട എന്താ ഈ Email Spamming



☆ Spam Messages



☆ ഇന്റർനെറ്റ് വഴി Spam Messages നിങ്ങളി Inbox ലേക്ക് അയയ്ക്കുക

" Email Spamming "

EMAIL SPAMMING

- Sending **junk** emails and commercial messages over internet.

Spam

- Unwanted messages.
- Email **spam**, also known as **junk** email.

- ഇൻറർ നെറ്റിലൂടെ അനാവശ്യ മെസ്സേജസ് സ്പ്രെഡ് ചെയ്യുക.




CYBER PHISHING

- An attempt or accessing sensitive information like OTP number, username, password, account number, ATM number etc... by using electronic mean.

- കമ്പ്യൂട്ടർ പോലുള്ള ഇലക്ട്രോണിക് മാധ്യമം ഉപയോഗിച്ചുകൊണ്ട് sensitive information ആക്സ് ചെയ്യുക.





☆ Sensitive Information

User Name

Password

OTP number

Cyber Phishing : ഇലക്ട്രോണിക് മാധ്യമം ഉപയോഗിച്ചുകൊണ്ട് sensitive information Access ചെയ്യുക.

TYPES OF PHISHING

a. Vishing

- Phishing by using telephone calls.
- ഫോൺ കോളുകൾ ഉപയോഗിച്ച് നടത്തുന്ന Phishing.





* സർ ഞാൻ SBI യിൽ നിന്നാണ് വിളിക്കുന്നത് നിങ്ങളുടെ ATM Card നമ്പർ തന്നാൽ Card Renew ചെയ്യാം



* ആഹാ..ഫോൺ വിളിച്ച് Vishing ചെയ്യുന്നുടാ.

Vishing : ഫോൺ കോൾ ഉപയോഗിച്ചുള്ള Phishing

TYPES OF PHISHING

b. Smishing

- Phishing by using messages(SMS).

- Messages(SMS)
ഉപയോഗിച്ച് നടത്തുന്ന
Phishing.



* ഉഗ്രാണ്ടയിൽ ലോട്ടറി അടിച്ചിട്ടുണ്ട് കാശ് അയക്കാൻ അക്കൗണ്ട് നമ്പർ ചോദിച്ചു ഫെയ്സ്ബുക്ക് വന്നപ്പോൾ ഞാൻ.



* Fake ഫെയ്സ്ബുക്ക് ആണെന്ന് അറിഞ്ഞപ്പോൾ



Smishing : Messages ഉപയോഗിച്ച് നടത്തുന്ന Phishing.

TYPES OF PHISHING

c. Spear Phishing

- Phishing targets a specific organization or person.

- ഒരു വ്യക്തിയെയോ ഒരു കമ്പനിയെയോ മാത്രം ലക്ഷ്യമാക്കി നടത്തുന്ന

Phishing



TYPES OF PHISHING

d. Whaling

- Phishing that's targeting high-profile.

- ഉന്നതന്മാരെ മാത്രം
Target ചെയ്യുന്നു.



CYBER STALKING OR BULLYING

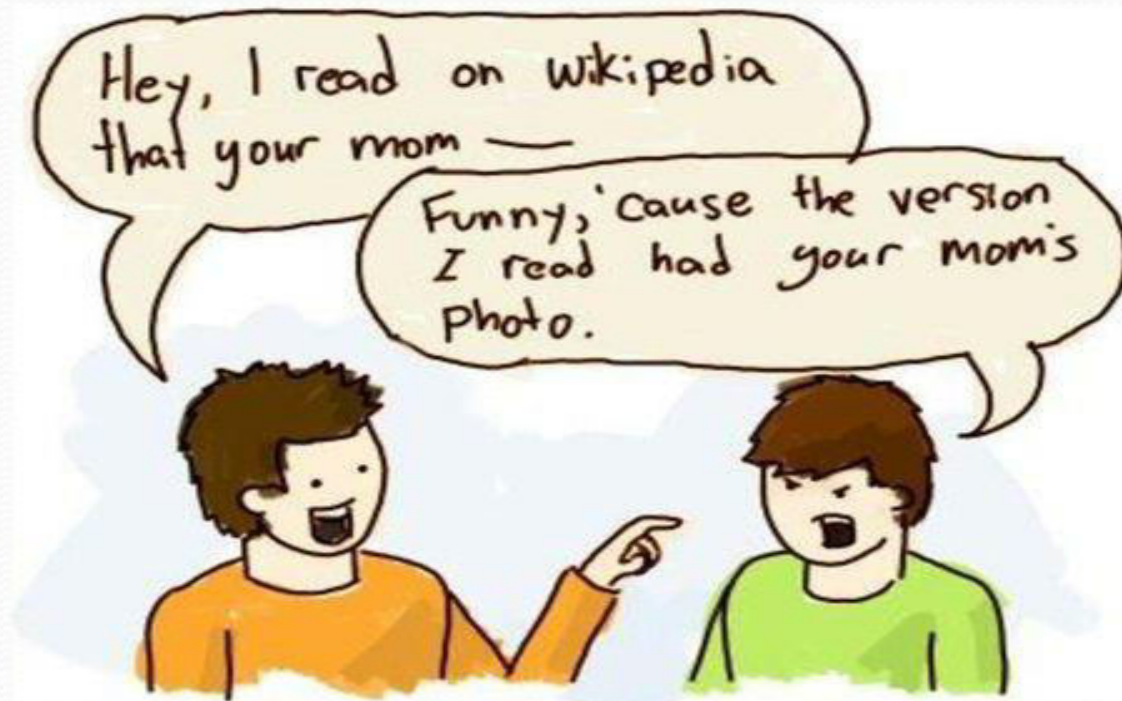
- Using Internet to harass or threaten a person, group, or organization.

• Internet ഉപയോഗിച്ച് ഭീഷണിപ്പെടുത്തുക.



Cyber defamation

- The tort of cyber defamation is considered to be the act of defaming, insulting, offending or otherwise causing harm through false statements pertaining to an individual in cyberspace.
- Example: someone publishes defamatory matter about someone on a website or sends an E-mail containing defamatory information to all friends of that person.



തൻ ആത്മവ്യാ .. ഞാൻ ഒരു കമ്പ്യൂട്ടറും ഇന്റർനെറ്റും കൊണ്ട് ഇറങ്ങാൻ
പോവുക തന്നെയും തന്റെ ഓരോയും നാറ്റിച്ച് ഇല്ലെങ്കിൽ തന്റെ പേര് പട്ടികയിട്ടോ



Cyber Defamation : ഇന്റർനെറ്റിലൂടെ വ്യാജപ്രചാരണം നടത്തി
ഒരാളെയോ ഒരു കമ്പനിയെയോ അപകീർത്തിപ്പെടുത്തുക

CYBER DEFAMATION

- Spreading false communication about a person, company, nation etc. through internet.

- .

- ഇന്റർനെറ്റിലൂടെ വ്യാജപ്രചാരണം നടത്തി ഒരാളെയോ ഒരു കമ്പനിയെയോ അപകീർത്തിപ്പെടുത്തുക..

-



Types of defamation

- Libel : written defamation
- Slander: oral defamation
- The plaintiff must have to show that the defamatory statements were unlawful and would indeed injure the person's or organization's reputation.
- When failed to prove, the person who made the allegations may still be held responsible for defamation.



CYBER SQUATTING

- Creating an identical website for official website with the similar domain name.

Eg : www.keralapsc.gov.in - Official website

www.psc.kerala.gov.in - Identical website.

- സമാനമായ domain name വരുന്ന identical website ഉണ്ടാക്കുക



www.keralapsc.gov.in

pssc.kerala.gov.in

Official Website

Identical Website



ഒഫീഷ്യൽ ഏതാണെന്ന് കൺഫ്യൂഷൻ ആയല്ലോ

Cybersquatting : ഒഫീഷ്യൽ വെബ് സൈറ്റിന് സമാനമായ
ഡൊമൈൻ നെയിം ഉള്ള വെബ്സൈറ്റ് ഉണ്ടാക്കുക

CYBER PHARMING

- Redirecting to a fake website.

• വ്യാജ വെബ്സൈറ്റിലേക്ക്
വഴിതിരിച്ചുവിടുക.



☆ ചോട്ട എനിക്ക് www.keralapsc.gov.in ലേക്കാണ്
ഫോക്കേണ്ടത്

☆ അങ്ങോട്ട് ദയക്കര ട്രാഫിക് നമുക്ക് psc.kerala.gov.in
ലേക്ക് ഫോക്കാം

☆ ഈ DUDE നോട് ആണോ നിനർനെ Cyber Pharming

Cyber Pharming : നഗ്ന വെബ്സൈറ്റിലേക്ക് നഴ്സിതിരിച്ചുവിടുക.

PORNOGRAPHY

- Spreading obscene(sexual) materials over internet.

• ഇൻറർനെറ്റ് ഉപയോഗിച്ച് അശ്ലീലം പ്രചരിപ്പിക്കുക.



Pornographic offenses: Child pornography

- Means any visual depiction, including but not limited to the following:
 1. Any photograph that can be considered obscene and/or unsuitable for the age of child viewer.
 2. Film ,video, picture;
 3. Obscene Computer generated image or picture

2.Cybercrime against property

- Credit card frauds
- Intellectual property(IP) crimes
- Internet time theft

Internet Time Theft

- Occurs when an unauthorized person uses the Internet hours paid for by another person
- Comes under hacking
- The person get access to someone else's ISP user ID and password, either by hacking or by gaining access to it by illegal means
- And uses the internet without the other person's knowledge
- This theft can be identified when Internet time is recharged often, despite infrequent usage.
- This comes under “identity theft”



3.Cybercrime against organization

- Unauthorized accessing of computer
- Password sniffing
- Denial-of-service attacks
- Virus attack/dissemination of viruses
- E-Mail bombing/mail bombs
- Salami attack/ Salami technique
- Logic bomb
- Trojan Horse
- Data diddling
- Industrial spying/ industrial espionage
- Computer network intrusions
- Software piracy

Password sniffing

- Password sniffers are programs that monitor and record the name and password of network users as they login, jeopardizing security at a site.
- through sniffers installed, anyone can impersonate an authorized user and login to access restricted documents.




DoS(denial-of-service attack)

- An attempt to make a computer resource unavailable to intended users.


Eg : Email Bombing, Network traffic, etc.

- മനഃപൂർവ്വം സെർവറിന്റെ അല്ലെങ്കിൽ നെറ്റ് വർക്കിന്റെ സർവീസിന്റെ ലഭ്യത തടസ്സപ്പെടുത്തുക.





* Service Access ചെയ്യാൻ ശ്രമിക്കുന്ന യൂസർ



മനപ്പൂർവ്വം തടസ്സപ്പെടുത്തുന്ന DoS Attacker

DoS(denial-of-service attack) : Service അല്ലെങ്കിൽ
Server നുകളുടെ ലഭ്യത മനപ്പൂർവ്വം തടസ്സപ്പെടുത്തുക


Salami attack/ salami technique

- Are used for committing financial crimes.
- The alterations made are so insignificant that in a single case it would go completely unnoticed.
- Example: a bank employee inserts a program, into the bank's serve, that deduces a small amount from the account of every customer every month,
- The unauthorised debit goes unnoticed by the customers, but the employee will make a sizable amount every month.

Data diddling



- Data diddling involves changing data input in a computer.
- In other words, information is changed from the way it should be entered by a person typing in the data.
- Usually, a virus that changes data or a programmer of the database or application has pre-programmed it to be changed.
- For example, a person entering accounting may change data to show their account, or that of a friend or family member, is paid in full. By changing or failing to enter the information, they are able to steal from the company.

- 
- To deal with this type of crime, a company must implement policies and internal controls.
 - This may include performing regular audits, using software with built-in features to combat such problems, and supervising employees.

DATA SNOOPING

- Snooping is the unauthorised access to another person's or companies data.
- മറ്റൊരു വ്യക്തിയുടെ അല്ലെങ്കിൽ കമ്പനികളുടെ ഡാറ്റയിലേക്കുള്ള അനധികൃത ആക്സ് ആണ് **SNOOPING** .





ഒരു വ്യക്തിയുടെയോ കമ്പനിയുടെയോ പ്രൈവറ്റ് ഡാറ്റാ അന്വേഷം ഇല്ലാതെ
ആക്സസ് ചെയ്യാൻ **Data Snooping**

Industrial spying/ Industrial Espionage

- Industrial espionage is the covert and sometimes illegal practice of investigating competitors to gain a business advantage.
- The target of investigation might be a trade secret such as a proprietary product specification or formula, or information about business plans.
- In many cases, industrial spies are simply seeking any data that their organization can exploit to its advantage.

Software piracy

- Theft of software through the illegal copying of genuine programs or the counterfeiting and distribution of products intended to pass for the original.
- End-user copying
- Hard disk loading with illicit means
- Counterfeiting
- Illegal downloads from internet



Buying Pirated software have a lot to lose:

- Getting untested software that may have been copied thousands of times.
- Potentially contain hard-ware infecting viruses
- No technical support in case of software failure
- No warranty protection
- No legal right to use the product

Computer sabotage



- Computer sabotage involves deliberate attacks intended to disable computers or networks for the purpose of disrupting commerce, education and recreation for personal gain, committing espionage, or facilitating criminal conspiracies.
- Through viruses, worms, logic bombs
- Chernobyl virus
 - **The Chernobyl virus is a computer virus with a potentially devastating payload that destroys all computer data when an infected file is executed.,**
- Y2K virus
 - **Y2K bug**, also called Year 2000 bug or Millennium Bug, a problem in the coding of computerized systems that was projected to create havoc in computers and computer networks around the world at the beginning of the year 2000



E-mail bombing/mail bombs

- In Internet usage, an *email bomb* is a form of net abuse consisting of sending huge volumes of *email* to an address in an attempt to overflow the mailbox or overwhelm the server where the *email* address is hosted in a denial-of-service attack.
- Construct a computer to repeatedly send E-mail to a specified person's E-mail address.
- Can overwhelm the recipient's personal account and potentially shut down the entire system.



Bulk Messages അമ്പൻറെ Inbox ലേക്ക് മാത്രം അയച്ചുകൊണ്ട്



*** Bulk Messages**

**Email Bombing : ഒരു പ്രത്യേക ഇമെയിൽ വിലാസത്തിലേക്ക്
വളരെയധികം ഇമെയിലുകൾ അയക്കുക**

EMAIL BOMBING

- Sending large number of email to a specific email address.

- ഒരു പ്രത്യേക ഇമെയിൽ addressലേക്ക് വളരെയധികം ഇമെയിലുകൾ അയക്കുക.





Computer network intrusions

- An intrusion to computer network from any where in the world and steal data, plant viruses, create backdoors, insert trojan horse or change passwords and user names.
- An intrusion detection system (IDS) inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system.
- The practice of strong password

Credit card frauds



- **Credit card fraud** is a wide-ranging term for theft and fraud committed using or involving a payment card, such as a credit card or debit card, as a fraudulent source of funds in a transaction.
- The purpose may be to obtain goods without paying, or to obtain unauthorized funds from an account.
- Credit card fraud is also an adjunct to identity theft.

Identity theft

- Identity theft is a fraud involving another person's identity for an illicit purpose.
- The criminal uses someone else's identity for his/ her own illegal purposes.
- Phishing and identity theft are related offenses
- Examples:
 - Fraudulently obtaining credit
 - Stealing money from victim's bank account
 - Using victim's credit card number
 - Establishing accounts with utility companies
 - Renting an apartment
 - Filing bankruptcy using the victim's name



4. Cybercrime against Society

- Forgery
- Cyberterrorism
- Web jacking

Forgery

- The act of forging something, especially the unlawful act of counterfeiting a document or object for the purposes of fraud or deception.
- Something that has been forged, especially a document that has been copied or remade to look like the original.
- Counterfeit currency notes, postage, revenue stamps, marksheets, etc., can be forged using sophisticated computers, printers and scanners.

CYBER TERRORISM

- Terrorist activity done by using computer, internet etc.

• കമ്പ്യൂട്ടർ ഇൻറർനെറ്റ്
തുടങ്ങിയവ ഉപയോഗിച്ച്
നടത്തുന്ന ഭീകര
പ്രവർത്തനങ്ങൾ



WEB JACKING

- Gaining access and control over the **web** site.

• വെബ്സൈറ്റിന്റെ നിയന്ത്രണം നേടിയെടുക്കുക.



Web jacking

- This term is derived from the term hi jacking.
- In these kinds of offences the hacker gains access and control over the web site of another.
- He may even change the information on the site.
- The first stage of this crime involves “password sniffing”.
- The actual owner of the website does not have any more control over what appears on that website
- This may be done for fulfilling political objectives or for money



5. Crimes emanating from Usenet newsgroup

- Usenet groups may carry very offensive, harmful, inaccurate material
- Postings that have been mislabeled or are deceptive in another way
- Hence service at your own risk

MODULE 6

CYBER CRIME INVESTIGATION

1

KEYLOGGERS

- *A Keylogger is malicious program which is used by the hackers to steal the personally identifiable data of the user.*

OR

- *A key logger is a program that runs in the background , recording all the keystrokes. Once keystrokes are logged, they are hidden in the machine for later retrieval, or shipped raw to the attacker.*
- It records the key strokes made by the users.
- It collects the needed information and sends it to the predetermined server using the internet.
- Attacker checks files carefully in the hopes of either finding passwords, or possibly other useful information.

KEYLOGGERS

- Key loggers, as a surveillance tool, are often used by employers to ensure employees use work computers for business purposes only.
- Such systems are also highly useful for law enforcement and espionage.
- Keystroke logging can be achieved by both hardware and software means. There are two types of keyloggers :
 - 1. Hardware Keylogger
 - 2. Software Keylogger

HARDWARE KEYLOGGERS

- Hardware keyloggers are used for keystroke logging, a method of capturing and recording computer users' keystrokes, including sensitive passwords.
- Generally, recorded data is retrieved by typing a special password into a computer text editor.
- The hardware keyloggers plugged in between the keyboard and computer detects that the password has been typed and then presents the computer with "typed" data to produce a menu.

HARDWARE KEYLOGGERS

- This device is able to monitor user's behaviors by physically hide the device in computer.
- Owing to the function of recording all keystrokes, it can also track sent emails, chat contents, instant messages, website address, etc

HARDWARE KEYLOGGERS

- The advantages of hardware keylogger can be divided into three points:
 - No need of installation; you just need to plug it to the computer.
 - Able to capture all keystrokes, including “Ctrl + C”, “Alt + F” and “Ctrl + Alt + Delete”.
 - Impossible to be detected by other software.

HARDWARE KEYLOGGERS

- Advantage of hardware keylogger is that it can begin logging keystrokes when a computer is turned on, which enables it to intercept password for the BIOS or disk encryption software.
- The disadvantage of this kind of keyloggers –
 - Person who installed the keylogger must retrieve and physically remove it in order to access the logs in the device.
 - This disadvantage make hardware keylogger a risky tool, for camera surveillance or the review of access card swipe records can helps other people determine who gained physical access to the area during the time period that the keylogger was removed.

HARDWARE KEYLOGGERS

- Come in three types:
 - Inline devices that are attached to the keyboard cable.
 - Devices which can be installed inside standard keyboards.
 - Replacement keyboards that contain the key logger already built-in.

SOFTWARE KEYLOGGERS

- Software keyloggers track system , collect keystroke data within the target operating system , store them on disk or in remote location , and send them to the attacker who installed the keyloggers.
- Anti malware, personal firewall, and Host-based Intrusion prevention(HIPS) solution detect and remove application keyloggers.
- It don't need physical access to target user's computer.
- It works on the target computer's operating system.

SOFTWARE KEYLOGGERS

- Almost all keylogger software has function of remote access, which means it allows. user to access the local records from a remote location.
- How does the remote access do that?
 - Some update the data to a website, database or FTP server.
 - Some automatically email the logs to pre-defined email address based on the preset time interval.
 - Some even allow remote login to the local machine via the Internet or the LAN.
- This may become one of keylogger software advantages, which helps user to capture target computer's information without depending on physical access to target user's computer.

SOFTWARE KEYLOGGERS

- The features of keylogger software are as follows:
 - 1. Keystroke logging: able to record all keystrokes typed in the keyboard after the computer turns on.
 - 2. Clipboard logging: all operations occurred in clipboard such as copy, paste, etc. can be captured.
 - 3. Screenshots logging: able to snap screenshots based on the time intervals.
 - 4. Applications logging: some keylogger software can record applications running on the computer with program name, time, etc.
 - 5. Some keylogger software has added features, such as visited websites record, or parental control functions. Compared to hardware keyloggers, one of the most significant advantages of keylogger software is it needs no direct physical access of target computer.

SOFTWARE KEYLOGGERS

- Scan local drive for log.txt or other log file names associate with known keyloggers.
 - Implement solution that detect unauthorized file transfer via FTP or other protocols.
- Scan content sent via email or other authorized means looking for sensitive information.
- Detect encrypted files transmitted to questionable destinations.

SPYWARE

- Applications that send information from your computer to the creator of the spyware.
- Sometimes consists of an apparent core functionality and a hidden functionality of information gathering (Trojan).
- Can be used by web sites for marketing information, to determine their stance with regard to competitors and market trends.
- Can also be used to log keystrokes and send those to whomever.

SPYWARE

- Software that is installed on a computer without the user's knowledge which monitors user activity and transmits it to another computer.
- Many spyware programs are set to monitor what web sites you visit them generally for advertising /marketing purposes.
- Software or hardware installed on a computer without the user's knowledge which gathers information about that user for later retrieval by whomever controls the spyware.
- Spyware can be broken down into two different categories: ☐
 - surveillance spyware.
 - advertising spyware.

CLASS OF SPYWARE

- **1 TRACKING COOKIES** - Cookies that can track your Web activities □ May include cookies that contain - user names ,passwords , other private information that you enter on web sites (SSN, banking info, credit cards).
- **2 BROWSER HIJACKING** –
 - Hosts File - Redefine the addresses of trusted sources, i.e. anti-virus tools, software patches and upgrades.
 - Home Page - Redefine the page that opens up when you start your browser.
 - Search Page - Redefine the page that opens up when you enter an undefined URL.
 - Redefine the page that opens up when you click your “Search” button .
 - Error Pages - Redefine the pages that open when an error occurs.

CLASS OF SPYWARE

- **3. SPYBOTS** - Spybots are the prototypical example of “spyware.” A spybot monitors a user’s behavior, collecting logs of activity & transmitting them to third parties.
 - A spybot may be installed as a browser helper object, it may exist as a DLL on the host computer, or it may run as a separate process launched whenever the host OS boots.
- **4. MALWARE** - Refers to a variety of malicious software, including viruses, worms, Trojan horses.
- **5. ADWARE** - Software that displays advertisements tuned to the user’s current activity, potentially reporting aggregate or anonymized browsing behavior to a third party.
- **6. KEYLOGGERS**

PREVENTION OF SPYWARE

- Do not installed free software available on internet.
- Do not click on email attachments or links of you don't know the sender or even if you send know the sender, but the content is unexpected.
- Do not installed unknown software.
- Do not click on links or buttons or popup windows.

VIRUS & WORMS



VIRUS

• Vital Information Resources Under Seize (Virus) .

- A computer virus is a malicious computer program (executable file) that can copy itself and infect a computer without permission or knowledge of the user.
- A virus can only spread from one computer to another by:
 - Sending it over a network as a file or as an email payload.
 - Carrying it on a removable medium.
- Viruses need USER INTERVENTION to spread ...
 - Some viruses are programmed to damage the computer **by damaging programs, deleting files, or reformatting the hard disk.**
 - Others are not designed to do any damage, **but simply replicate themselves** and perhaps make their presence known by presenting text, video, or audio messages.



VIRUS

- **Infection mechanism:** The means by which a virus spreads, enabling it to replicate. This mechanism is also referred to as the infection vector.
- **Trigger:** The event or condition that determines when the payload is activated or delivered.

VIRUS— PHASE OF VIRUS.

- **Dormant phase:** It's the state when virus is idle. The virus will eventually be activated by some event, such as a date, the presence of another program or file, or the capacity of the disk exceeding some limit. Not all viruses have this stage.
- **Propagation phase:** The virus places a copy of itself into other programs or into certain system areas on the disk. The copy may not be identical to the propagating version; viruses often morph to evade detection. Each infected program will now contain a clone of the virus, which will itself enter a propagation phase.

VIRUS — PHASE OF VIRUS.

- **Triggering phase:** The virus is activated to perform the function for which it was intended. As with the dormant phase, the triggering phase can be caused by a variety of system events, including a count of the number of times that this copy of the virus has made copies of itself.
- **Execution phase:** The function is performed. The function may be harmless, such as a message on the screen, or damaging, such as the destruction of programs and data files.

VIRUS – TYPES OF VIRUS.

- **BOOT SECTOR** - Virus that infects the boot sector of floppy disks or the Master Boot Record (MBR) of hard disks (some infect the boot sector of the hard disk instead of the MBR).
- **FILE INFECTOR** - Virus that usually infects memory and executable files, Once they are in system they remain for a long time.
- **MACRO VIRUS** - Virus that "infects" a Microsoft Word or similar application and causes a sequence of actions to be performed automatically when the application is started or something else triggers it. Macro viruses tend to be surprising but relatively harmless.

VIRUS – TYPES OF VIRUS.

- **ENCRYPTED VIRUS** - Virus using encryption to hide itself from virus scanners. That is, the encrypted virus jumbles up its program code to make it difficult to detect. An encrypted virus's code begins with a decryption algorithm and continues with scrambled or encrypted code for the remainder of the virus.
- **STEALTH VIRUS** - virus that uses various mechanisms to avoid detection by antivirus software.
- **POLYMORPHIC VIRUS** - virus which is able to modify itself and making clone of it.
- **METAMORPHIC VIRUS** - virus that can transform based on the ability to translate, edit and rewrite its own code. It is considered the most infectious computer virus, and it can do serious damage to a system if it isn't detected quickly.

WORMS

- A computer worm is a type of malware that spreads copies of itself from computer to computer.
- A worm can replicate itself **without any human interaction, and it does not need to attach itself to a software program in order to cause damage.**

HOW TO TELL IF YOUR COMPUTER HAS A WORM

- **Keep an eye on your hard drive space**. When worms repeatedly replicate themselves, they start to use up the free space on your computer.
- **Monitor speed and performance**. Has your computer seemed a little sluggish lately? Are some of your programs crashing or not running properly? That could be a red flag that a worm is eating up your processing power.
- **Be on the lookout for missing or new files**. One function of a computer worm is to delete and replace files on a computer.

WORMS

- Worms can be transmitted via software vulnerabilities.
- Computer worms could arrive as attachments in spam emails or instant messages (IMs). Once opened, these files could provide a link to a malicious website or automatically download the computer worm. Once it's installed, the worm silently goes to work and infects the machine without the user's knowledge.
- Worms can modify and delete files, and they can even inject additional malicious software onto a computer. Sometimes a computer worm's purpose is only to make copies of itself over and over — depleting system resources, such as hard drive space or bandwidth, by overloading a shared network. In addition to wreaking havoc on a computer's resources, worms can also steal data, install a backdoor, and allow a hacker to gain control over a computer and its system settings.

STEGANOGRAPHY

- It is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity.
- The word steganography is of Greek origin and means "concealed writing" from the Greek words steganos meaning "covered or protected", and graphein meaning "writing".
- “Steganography means hiding one piece of data within another”.

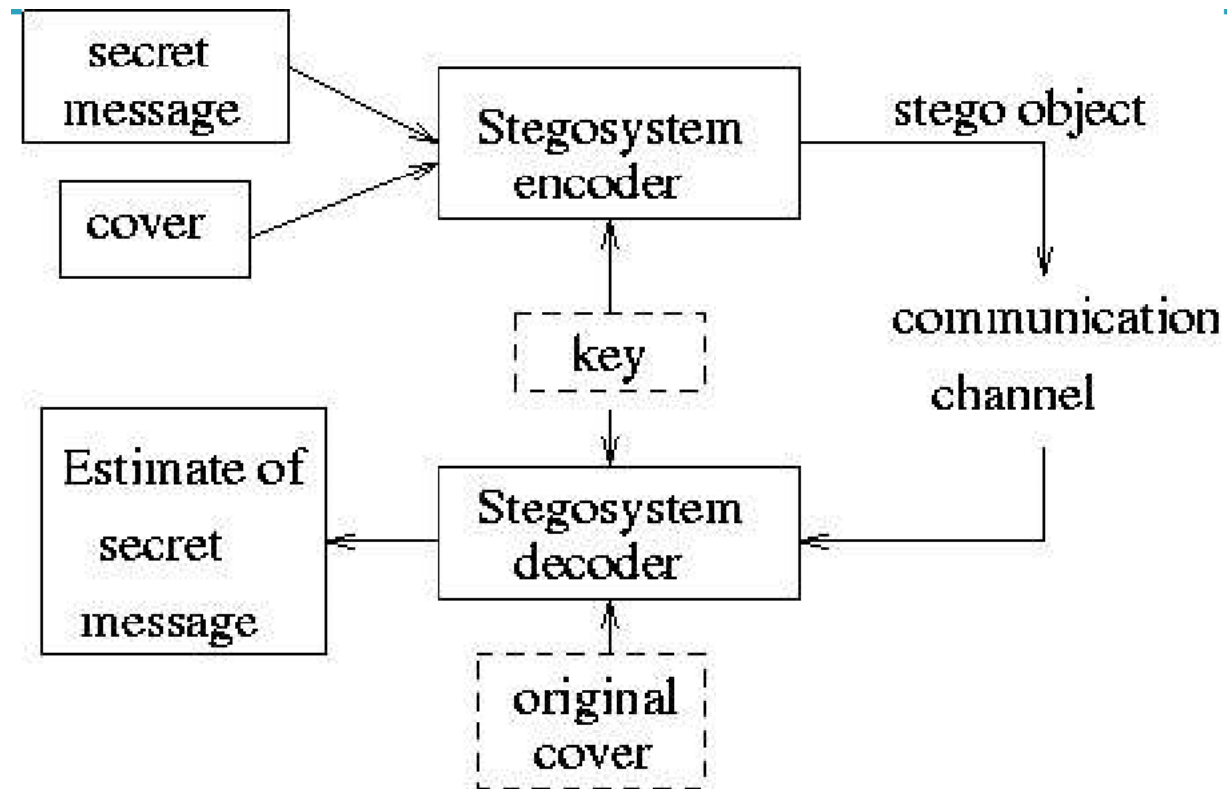
STEGANOGRAPHY

- Concealing messages within the lowest bits of noisy images or sound files. □ Chaffing and winnowing.
- Modifying the echo of a sound file (Echo Steganography).
- Including data in ignored sections of a file, such as after the logical end of the carrier file.

STEGANOGRAPHY

Steganography	Cryptography
Unknown message passing	Known message passing
Steganography prevents discovery of the very existence of communication	Encryption prevents an unauthorized party from discovering the contents of a communication
Little known technology	Common technology
Technology still being develop for certain formats	Most of algorithm known by all
Once detected message is known	Strong current algorithm are resistant to attacks ,larger expensive computing power is required for cracking

STEGANOGRAPHY



STEGANOGRAPHY

- **Carrier or Cover File** - A Original message or a file in which hidden information will be stored inside of it .
- **Stego-Medium** - The medium in which the information is hidden.
- **Embedded or Payload** - The information which is to be hidden or concealed.
- **Steganalysis** - The process of detecting hidden information inside a file.

STEGANOGRAPHY

- There are three basic types of stegosystems
- **Pure stegosystems** - no key is used.
- **Secret-key stegosystems** - secret key is used.
- **Public-key stegosystems** - public key is used

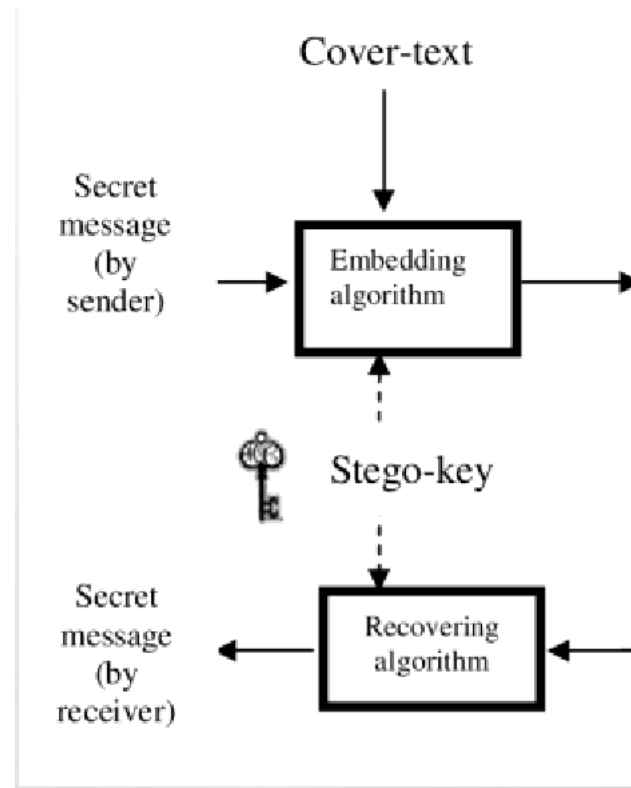
TEXT STEGANOGRAPHY

- Text steganography can be applied in the digital makeup format such as PDF, digital watermark or information hiding.
- It is more difficult to realize the information hiding based on text.
- The simplest method of information hiding is to select the cover first, adopt given rules to add the phraseological or spelling mistakes, or replace with synonymy words.

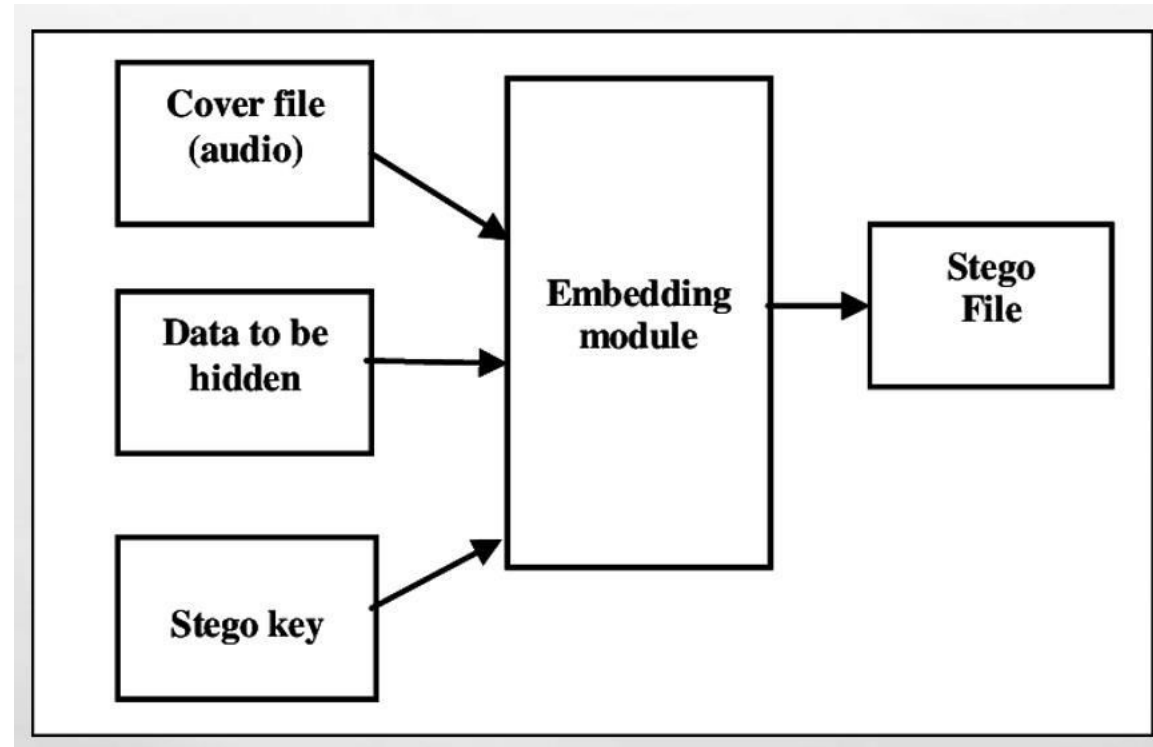
TEXT STEGANOGRAPHY METHODS

- Line shifting Method.
- Word shifting.
- Open spaces.
- Semantic methods.
- Character Encoding.
- An example of a message containing cipher text by German Spy in World War II:
 - *“Apparently neutral's protest is thoroughly discounted And ignored. Isman hard hit. Blockade issue affects Pretext for embargo on by products, ejecting suets and Vegetable oils. ”*
 - *Pershing sails from NY June 1.*

TEXT STEGANOGRAPHY



AUDIO STEGANOGRAPHY



VIDEO STEGANOGRAPHY

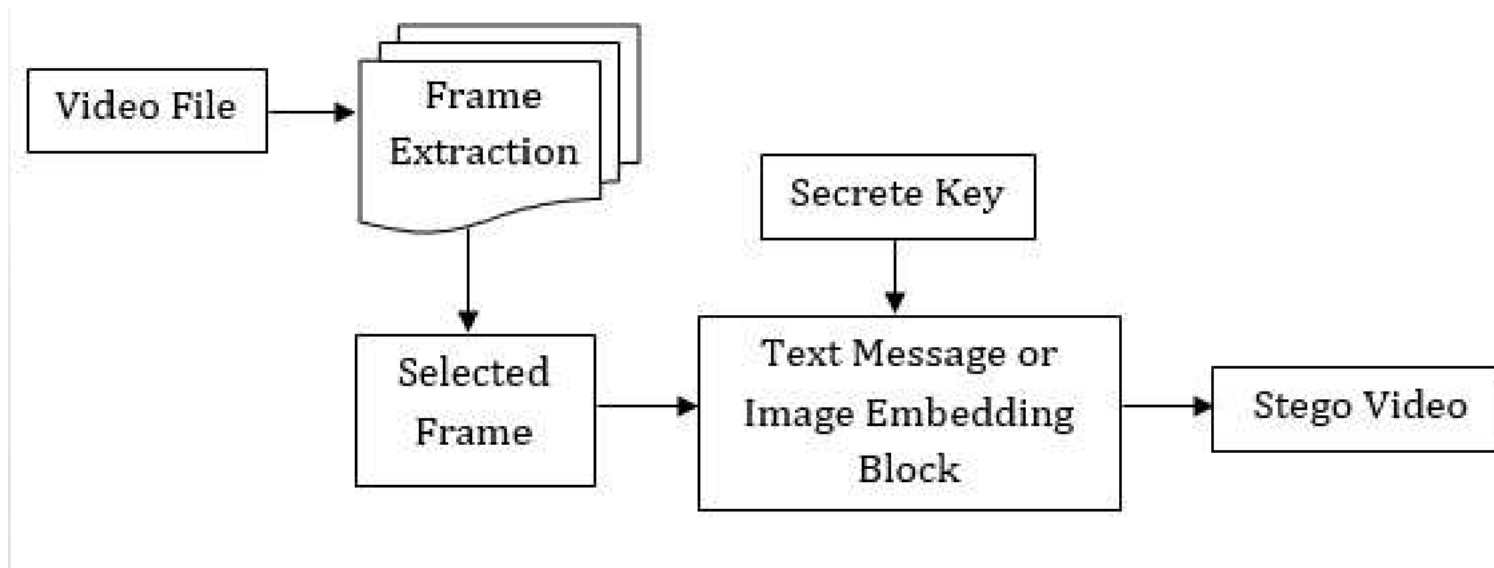


IMAGE STEGANOGRAPHY

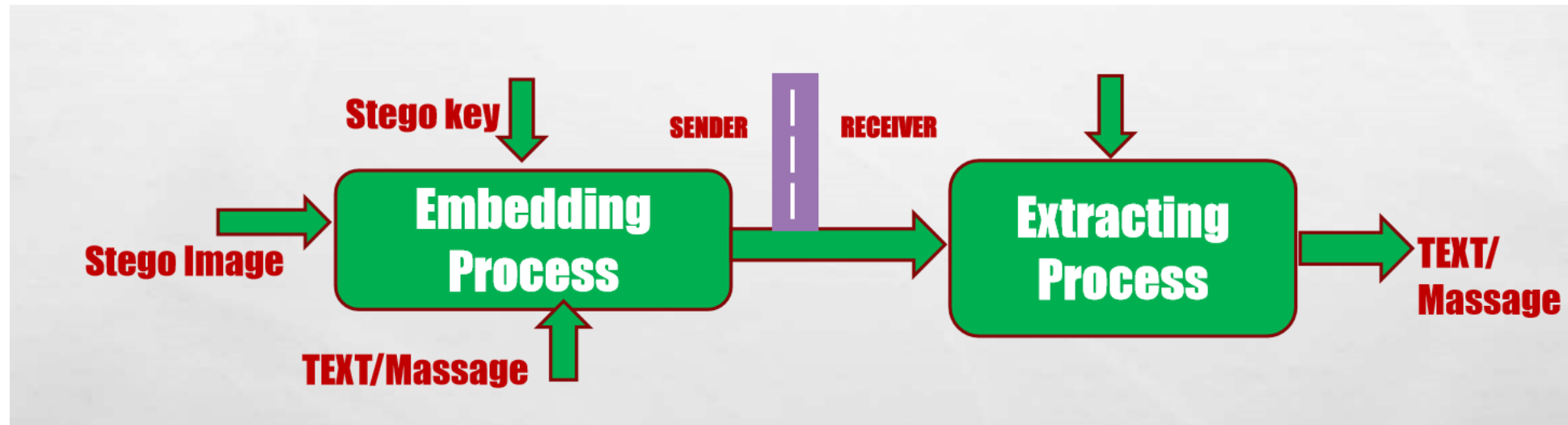
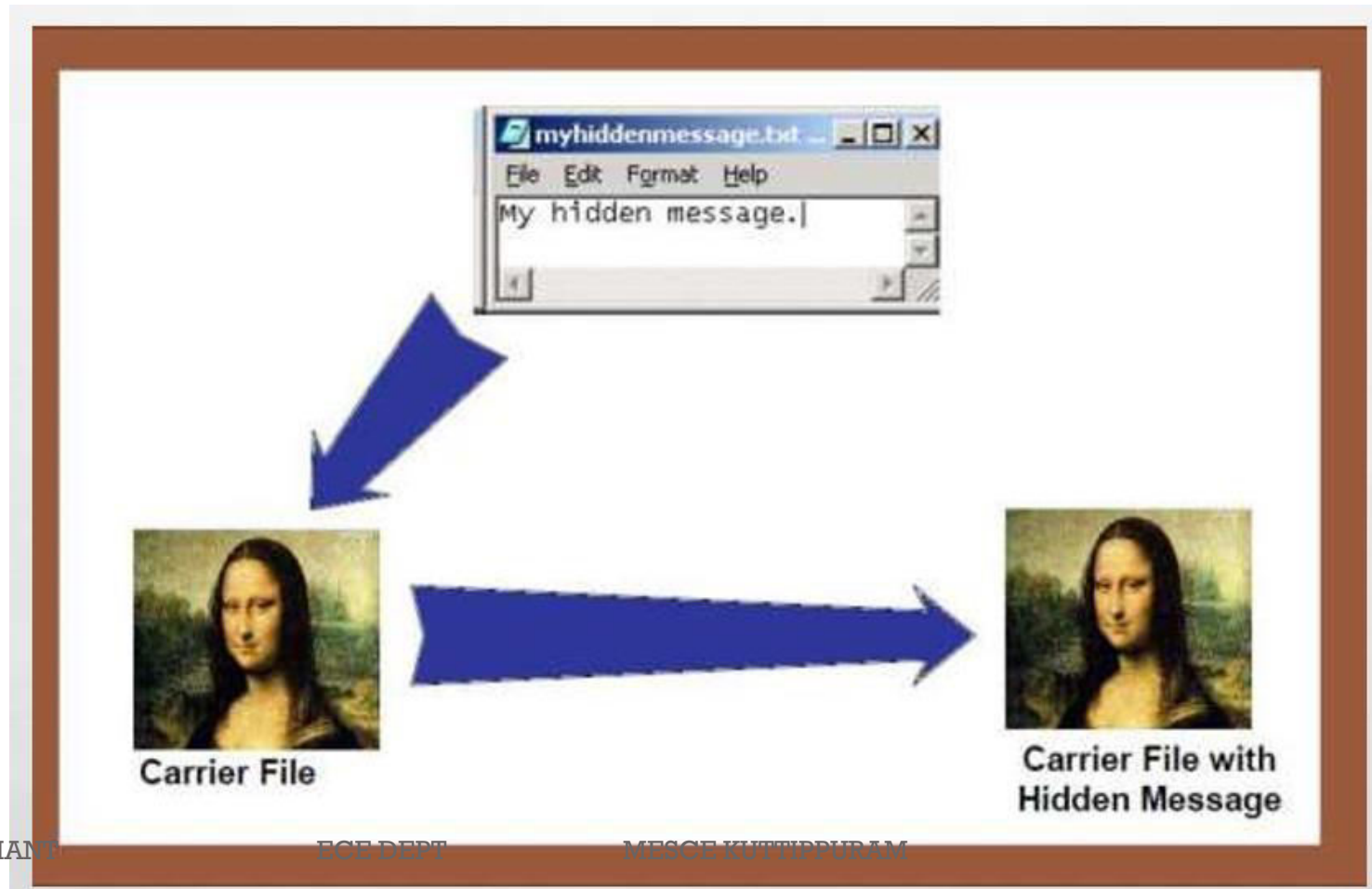
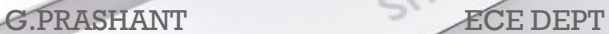


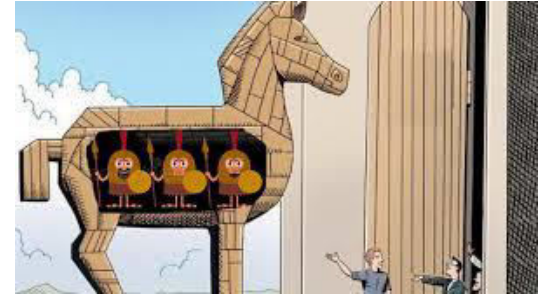
IMAGE STEGANOGRAPHY



TROJAN & BACKDOORS

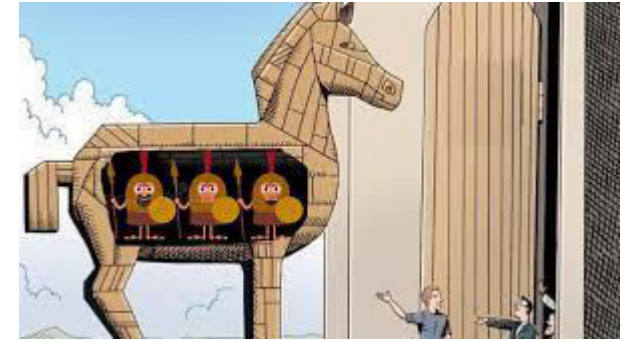


TROJAN HORSE

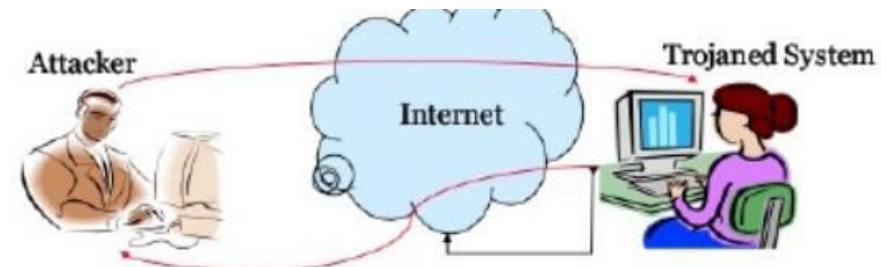


- Trojan horse is a malicious program that is designed as authentic, real and genuine software.
- Trojan Horses appear to be useful or interesting to an unsuspecting user, but are actually harmful.
- It can affect a system in following ways
 - Erase or overwrite data on a computer.
 - Spread other viruses or install a backdoor. In this case the Trojan horse is called a dropper.
 - Setting up networks of zombie computers in order to launch DDoS attacks or send Spam.
 - Logging keystrokes to steal information such as passwords and credit card numbers (known as a key logger).
 - Phish for bank or other account details, which can be used for criminal activities.
 - Or simply to destroy data.
 - Mail the password file.

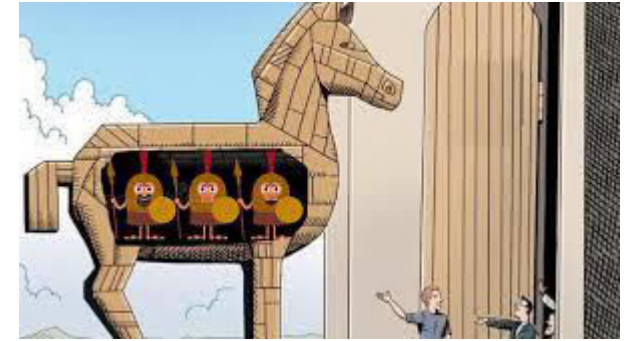
TROJAN HORSE



- Trojan horse is typically a Windows executable program file, and must have an executable file extension such as .exe, .com, .scr, .bat, or .pif.
- Since Windows is configured by default to hide extensions from a user, the Trojan horse's extension might be "masked" by giving it a name such as Readme.txt.exe.
- With file extensions hidden, the user would only see Readme.txt and could mistake it for a harmless text file.
- Unlike a computer virus, a Trojan horse is not able to replicate itself, nor can it propagate without an end user's assistance.

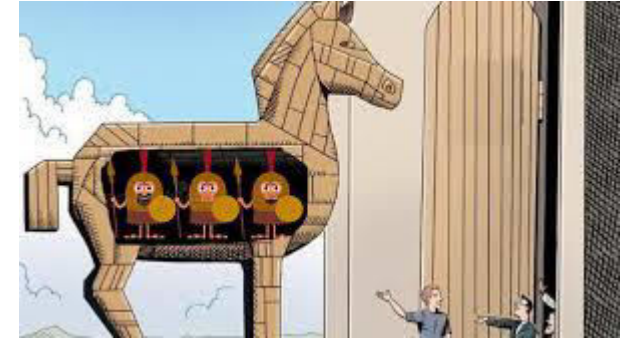


TROJAN HORSE



- The victim receives an official-looking email with an attachment. The attachment contains malicious code that is executed as soon as the victim clicks on the attachment.
- Because nothing bad happens and the computer continues to work as expected, the victim does not suspect that the attachment is actually a Trojan horse and his computing device is now infected.
- The malicious code resides undetected until a specific date or until the victim carries out a specific action, such as visiting a banking website. At that time, the trigger activates the malicious code and carries out its intended action.
- Depending upon how the Trojan has been created, it may delete itself after it has carried out its intended function, it may return to a dormant state or it may continue to be active.

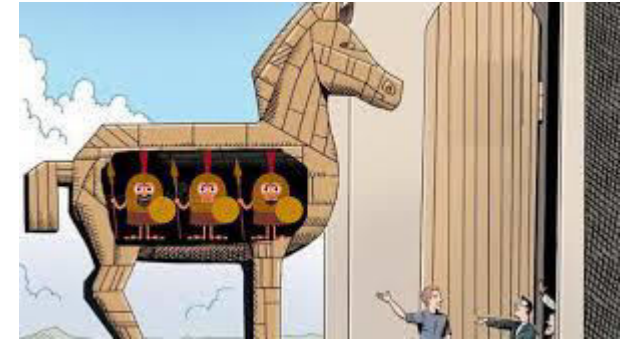
EXAMPLES OF TROJAN HORSE



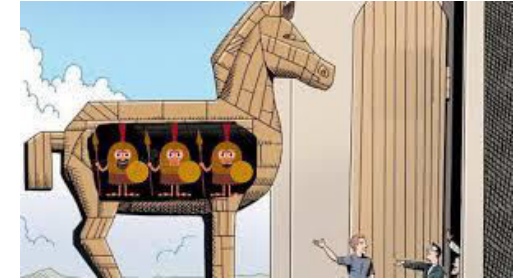
- **Bitfrost** -- remote access Trojan (RAT) that infected Windows clients by changing, creating and altering components.
- **Tiny Banker** -- allowed attackers to steal sensitive financial information. Researchers in the Center for Strategic and International Studies Security Group identified 'Tinba' in 2012 after two dozen major U.S. banks were infected.
- **FakeAV Trojan** -- embedded itself in the Windows system tray and continuously delivered an official-looking pop-up window, alerting the user to a problem with the computer. When users followed directions to fix the problem, they actually downloaded more malware.
- **Magic Lantern** -- a keystroke logging Trojan created by the FBI around the turn of the century to assist with criminal surveillance.
- **Zeus** -- a financial services crimeware toolkit that allows a hacker to build his own Trojan horse. First detected in 2007, the Trojans built with Zeus still remain the most dangerous banking Trojans in the world, using form grabbing, keylogging and polymorphic variants of the Trojan that use drive-by downloads to capture victim credentials.

TYPES OF TROJAN HORSE

- Remote access trojans
- Data sending trojans
- Destructive trojans
- Denial of service attack trojans
- Proxy trojans
- FTP trojans
- Security software disabler.

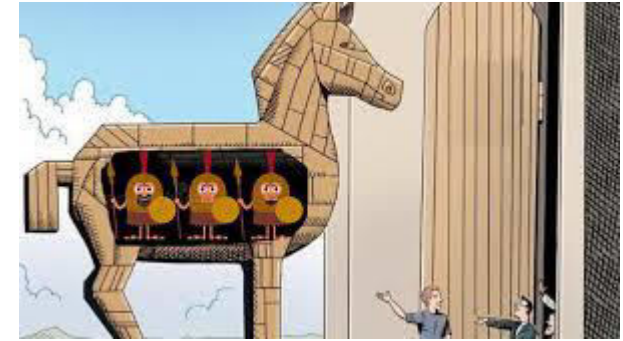


TYPES OF TROJAN HORSE



- **Trojan-Downloader:** is a type of virus that downloads and installs other malware.
- **Trojan-Droppers** are complex programs used by cyber criminals to install malware. Most antivirus programs do not detect droppers as malicious, and hence it is used to install viruses.
- **Ransomware** - It is a type of Trojan (Trojan - ransom) that can encrypt the data on your computer/device. The cyber criminals who control this ransomware would demand a ransom for providing the decryption key. It is very difficult to recover the data without the decryption key. The WannaCry and Petya were recent ransomware attacks. Cyber security experts recommend users to follow a robust and systematic backup and recovery policy
- **Trojan-Banker** malware programs steal account-related information related to card payments and online banking.

TYPES OF TROJAN HORSE



- **Trojan-Rootkits** prevent detection of malware and malicious activities on the computer. These are sophisticated malware that provides control of the victim's device. Rootkits are also used to enroll the victim's device as part of a botnet.
- **Trojan-Backdoor** is a popular type of Trojan. It creates a backdoor to allow cyber criminals to access the computer later on from remote using a remote access tool (RAT). As this Trojan provides complete control over the computer, it is a dangerous but commonly used Trojan.

BACKDOOR



- A backdoor is a malicious computer program used to provide the attacker with unauthorized remote access to a compromised PC **by exploiting security vulnerabilities**. This backdoor virus works in the background and hides from the user.
- It allows a malicious person to execute any possible actions on a compromised computer.
- Often a backdoor is known to have additional destructive capabilities, such as screenshot capture, keystroke logging, file infection, and encryption.
- This virus is a combination of different security and privacy threats, which works on its own and does not need to be controlled at all.



BACKDOOR

- Backdoors lack the capability of spreading themselves and infecting systems without a user's knowledge. These threats get into the system via four main ways:
 - A number of backdoors have been already integrated into specific applications. Even genuine programs may have undocumented remote access features. Here, the attacker will have to contact a computer with such software installed to immediately get complete unauthorized access to the system or take over control over the specific software.
 - There are a few backdoors that infect a computer **by exploiting specific software vulnerabilities. They work just like worms and automatically spread without the user's knowledge.**
 - PC users can accidentally install typical backdoors on their computers with being completely aware. **A backdoor virus can come attached to the file-sharing programs or e-mail messages.** By giving them unsuspecting names, their authors trick users into executing or opening such files.
 - Very often, backdoors are installed by other parasites like Trojans, viruses, or even spyware. They manage to enter into a system without a user's knowledge, and they then affect each of users who use a compromised computer.
 - It is possible for some threats to be manually installed by malicious users who have adequate privileges for the software installation. The small part of backdoors can spread by exploiting remote systems with specific security vulnerabilities.

BACKDOOR



- After entering your system, a backdoor virus **causes the following activities:**
 - Permits the intruder to create, delete, rename, edit or copy any file, execute different commands, change any system settings, adjust the Windows registry, run, control and terminate applications, and install other software and parasites.
 - Records keystrokes and captures screenshots.
 - Allows the attacker to control computer hardware devices, alter related settings, restart or shutdown a computer without asking for permission.
 - Steals sensitive personal data, passwords, login names, identity details, and valuable documents. Logs user activity and tracks web browsing habits.



BACKDOOR

- Infects files, damages the entire system, and corrupts installed applications.
- Prevents its removal by providing no uninstall feature.
- Reduces Internet connection speed and overall system performance.
- Distributes infected files to remote computers with specific security vulnerabilities and executes attacks against hacker defined remote hosts.
- Installs hidden FTP server that can be employed by malicious individuals for different illegal purposes.

DOS OR DDOS



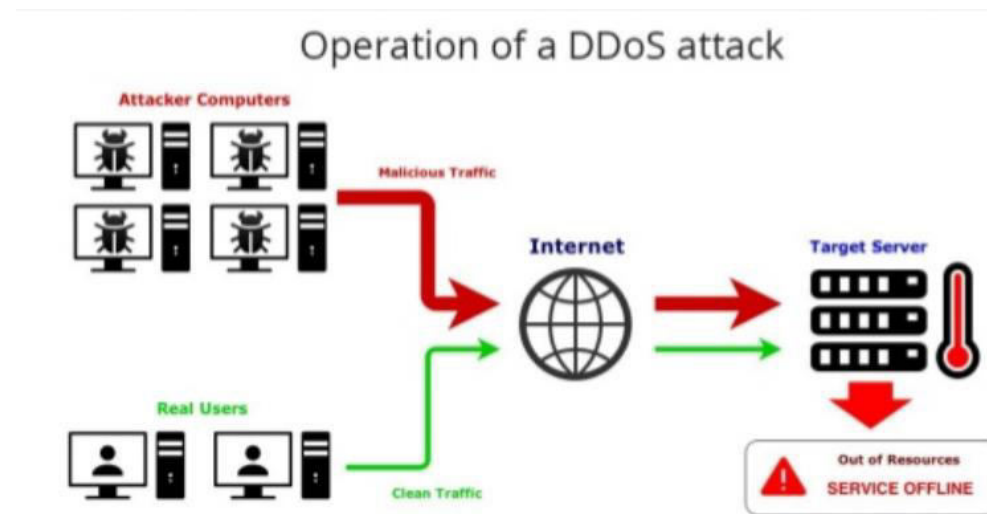
53

DOS OR DDoS

- A Distributed Denial of Service (DDoS) attack is an attempt to make an online service unavailable by overwhelming it with traffic from multiple sources.
- In a DoS attack, a hacker uses a single Internet connection to either exploit a software vulnerability or flood a target with fake requests — usually in an attempt to exhaust server resources .
- On the other hand, distributed denial of service (DDoS) attacks are launched from multiple connected devices that are distributed across the Internet.
- In a typical DDoS attack, the hacker begins by exploiting a computer system and making it the DDoS master.
- The attack master system identifies other vulnerable systems and gains control over them by either infecting the systems with malware or through bypassing the authentication controls .

DOS OR DDOS

- The attacker creates a command-and- control server to command the network of bots, also called a botnet.
- The person in control of a botnet is sometimes referred to as the botmaster.
- Their main aim is to prevent legitimate users from accessing a system or site.



DOS OR DDOS

- Training ground for other attacks.
 - Distraction from other malicious actions.
 - Anticompetitive business practices.
 - Means to extract money.
 - To disrupt operation of private or government enterprise.
- DDoS attacks are the single largest threat to our Internet and the Internet of Things.
 - The more our world becomes connected and dependent on the Internet, the more opportunities to get exploited by these types of attack

TYPES OF DDOS ATTACKS

- **Network or Volume centric attack (64%)** - These attacks use bots and botnets to flood the network layers with a substantial amount of seemingly legitimate traffic. This consumes an excessive amount of bandwidth within or outside of the network and causes network operations to become extremely slow or to not work at all.
- These kinds of attacks are more difficult to mitigate than attacks from a single source .Volumetric attacks come in a variety of forms, including ,User Datagram Protocol (UDP) Floods ,ICMP floods (Internet Control Message Protocol) Network or Volume Centric.

TYPES OF DDOS ATTACKS

- **Application layer attack - 16%** - The goal of an application layer attack is to exhaust resources by consuming too much. They target the layer that manages HTTP and SMTP communication. They target web application packets in order to disrupt the transmission of data between hosts. They attack on apache ,windows web server , as they are more vulnerable.
- These type of attacks are more sophisticated and are gaining in popularity than other types of DDoS attacks. For example:- an HTTP Flood – the most common application-layer attack – uses botnets to force a target to expend an excessive amount of resources when responding to a HTTP request.
- HTTP floods and other application-layer DDoS attacks mimic human-user behavior making them much more difficult to detect than other types of attacks.
- Web based email apps, WordPress, Joomla, and forum software are good examples of application specific targets.

TYPES OF DDOS ATTACKS

- **Protocol attacks (20%)** - Protocol attacks target the connection state tables in firewalls, web application servers, and other infrastructure components. One of the most common state-exhaustion attacks was the ping of death, in which a 65,536-byte ping packet is defragmented and sent to a target server as fast as possible.
- Once the target reassembles the large packet, a buffer overload typically occurs. In the likely scenario that the target attempts to respond to the pings, even more bandwidth is consumed, eventually causing the targeted system to crash.

SQL INJECTION

60

SQL INJECTION

- SQL injection is an attack technique that exploits a **security vulnerability occurring in the database layer of an application.**
- Hackers use injections to obtain unauthorized access to the underlying data, structure, and DBMS. **It is one of the most common web application vulnerabilities.**
- A Database is the heart of many web-applications and is used to store information needed by the application, such as, credit card information, customer demographics, customer orders, client preferences, etc.
- Consequently, databases have become attractive and very lucrative targets for hackers to hack into.
- SQL Injections happen when a developer accepts user input that is directly placed into a SQL Statement and doesn't properly validate and filter out dangerous characters.
- This can allow an attacker to alter SQL statements passed to the database as parameters and enable her to not only steal data from your database, but also modify and delete it.

SQL INJECTION

- The main categories of SQL injection attacks.
 - **Classic Attack** - Most attacks rely **on basic SQL manipulation** and are considered to be classic attacks. It includes WHERE clause modification, UNION operator injection and query stacking. Those are by far the most popular kinds of SQLIA .
 - **Inference attack** - Inference attacks involve a SQL manipulation that will provide the hacker the ability to verify a true/false condition. Depending on the database system reaction, it is possible to find out if the condition was realized or not.
 - **DBMS specific attack** - This type of SQLIA is used as an alternative to classic SQL injection. It is especially useful when trying to fingerprint the database system, but it can also provide the ability to achieve a complete attack when some particular conditions are met.

SQL INJECTION

- The risk of SQL injection exploits is on the rise because of automated tools. In the past, the danger was somewhat limited because an exploit had to be carried out manually: an attacker had to actually type their SQL statement into a text box.
- However, automated SQL injection programs are now available, and as a result, both the likelihood and the potential damage of an exploit has increased enormously.
- In an interview with Security Wire Perspectives, Caleb Sima, CTO of SPI Dynamics spoke of the potential danger: "This technology being publicly released by some black hat will give script-kiddies the ability to pick up a freeware tool, point it at a Web site and automatically download a database without any knowledge whatsoever.
- I think that makes things a lot more critical and severe. The automation of SQL injection gives rise to the possibility of a SQL injection worm, which is very possible. In fact, I am surprised this hasn't occurred yet." Sima estimates that about 60% of Web applications that use dynamic content are vulnerable to SQL injection.

SQL INJECTION

- It is the vulnerability that results when you give an attacker the ability to influence the Structured Query Language (SQL) queries that an application passes to a back-end database.
- By being able to influence what is passed to the database, the attacker can leverage the syntax and capabilities of SQL itself, as well as the power and flexibility of supporting database functionality and operating system functionality available to the database.
- SQL injection is not a vulnerability that exclusively affects Web applications; any code that accepts input from an untrusted source and then uses that input to form dynamic SQL statements could be vulnerable .

BUFFER OVERFLOW

65

BUFFER OVERFLOW

- A buffer overflow condition exists when a program attempts to put more data in a buffer than it can hold or when a program attempts to put data in a memory area past a buffer.
- In this case, a buffer is a sequential section of memory allocated to contain anything from a character string to an array of integers. Writing outside the bounds of a block of allocated memory can corrupt data, crash the program, or cause the execution of malicious code.
- Buffer overflow is probably the best known form of software security vulnerability. Most software developers know what a buffer overflow vulnerability is, but buffer overflow attacks against both legacy and newly developed applications are still quite common.
- Part of the problem is due to the wide variety of ways buffer overflows can occur, and part is due to the error prone techniques often used to prevent them.
- Exploiting a buffer overflow allows an attacker to control or crash the process or to modify its internal variables.

BUFFER OVERFLOW

- Buffer overflow always ranks high in the Common Weakness Enumeration/SANS Top 25 Most Dangerous Software Errors and is specified as CWE-120 under the Common Weakness Enumeration dictionary of weakness types.
- Despite being well understood, buffer overflows continue to plague software from vendors both large and small.
- A buffer overflow can occur inadvertently, but it can also be caused by a malicious actor sending carefully crafted input to a program that then attempts to store the input in a buffer that isn't large enough for that input.
- **If the excess data is written to the adjacent buffer, it overwrites any data held there. If the original data includes the exploited Function's return pointer -- the address to which the process should go next -- an attacker can set the new values to point to an address of his choosing.**
- The attacker usually sets the new values to point to a location where the exploit Payload has been positioned. This alters the execution path of the process and effectively transfers control to the attacker's malicious code.

BUFFER OVERFLOW

- The heap is a memory structure used to manage dynamic memory. Programmers often use the heap to allocate memory whose size is not known at compile time, where the amount of memory required is too large to fit on the stack or where the memory is intended to be used across function calls.
- Other buffer-related attacks include integer overflow, which is when a number is used in an operation, the result of which requires more memory to store. For example, 8 bits of memory are required to store the number 192. If the process adds 64 to this number, the answer 256 will not fit in the allocated memory, as it requires 9 bits.
- Format strings attacks alter the flow of an application by using string formatting library functions like printf and sprintf to access other memory space.

BUFFER OVERFLOW

- Finally, a Unicode overflow exploits the greater memory required to store a string in Unicode format than in ASCII characters.
- The most common reason why buffer overflow attacks work is because applications fail to manage memory allocations and validate input from the client or other processes.
- Applications developed in C or C++ should avoid dangerous standard Library functions that are not bounds checked, such as gets, scanf and strcpy, and instead use libraries or classes explicitly created to perform string and other memory operations securely.
- User input and data from untrusted sources should always be validated to ensure that they are within the bounds of what's expected and to prevent overly long input values.
- Vendors issue patches and updates for their software to fix buffer overflow vulnerabilities that have been discovered, but there is still a period of risk between the vulnerability being discovered and the patch being created and deployed.

BUFFER OVERFLOW

- Most operating systems have introduced runtime protections to make it harder for overflow buffer attacks to succeed.
- Address Space layout randomization randomly arranges the address space positions of key data areas of a process, including the base of the executable and the positions of the stack, heap and libraries. This makes it difficult for an attacker to reliably jump to a particular function in memory.
- Data Execution Prevention marks areas of memory as either executable or nonexecutable. This prevents an attacker from being able to execute instructions written to a data area via a buffer overflow.

ATTACK ON WIRELESS NETWORK

71

WIRELESS NETWORK ATTACK

■ 1) Access Control Attacks.

- **War driving** - In a wardriving attack, wireless LANS are detected either by sending probe requests over a connection or by listening to web beacons. Once a penetration point is discovered, further attacks can be launched on the LAN. Some of the tools that can be used to perform wardriving are KisMAC, NetStumbler, and WaveStumber.
- **Rogue Access Points** - In order to create a backdoor into a trusted network, an unsecured access point or fake access point is installed inside a firewall. Any software or hardware access points can be used to perform this kind of attack.
- **MAC Spoofing** - Using the MAC spoofing technique, the attacker can reconfigure the MAC address to appear as an authorized access point to a host on a trusted network. The tools for carrying out this kind of attack are: changemac.sh, SMAC, and Wicontrol.

WIRELESS NETWORK ATTACK

- **Ad Hoc Associations** - This kind of attack can be carried out by using any USB adapter or wireless card. In this method, the host is connected to an unsecured station to attack a particular station or to avoid access point security.
- **AP Misconfiguration** - If any of the critical security settings is improperly configured at any of the access points, the entire network could be open to vulnerabilities and attacks. The AP can't trigger alerts in most intrusion-detection systems, as it is authorized as a legitimate device on the network.
- **Client Misassociation** - The client may connect or associate with an AP outside the legitimate network either intentionally or accidentally. This is because the WLAN signals travel through walls in the air. This kind of client misassociation thus can be lead to access control attacks.

WIRELESS NETWORK ATTACK

- **Unauthorized Association** - Unauthorized association is the major threat to wireless network. Prevention of this kind of attack depends on the method or technique that the attacker uses in order to get associated with the network.
- **Promiscuous Client** - The promiscuous client offers an irresistibly strong signal intentionally for malicious purposes. Wireless cards often look for a stronger signal to connect to a network. In this way the promiscuous client grabs the attention of the users towards it by sending strong signal.

WIRELESS NETWORK ATTACK

■ 2) Integrity Attacks

- In integrity attacks, attackers send forged control, management, or data frames over a wireless network to misdirect the wireless devices in order to perform another type of attack (e.g., DoS).

Type of Attack	Description	Method and Tools
Data Frame Injection	Crafting and sending forged 802.11 frames.	Airpwn, File2air, libradiate, voidll, WEPWedgie, wnet dinject/reinject
WEP Injection	Crafting and sending forged WEP encryption keys	WEP cracking + injection tools
Data Replay	Capturing 802.11 data frames for later (modified) replay.	Capture + injection tools

WIRELESS NETWORK ATTACK

■ Integrity Attacks

Initialization Vector Replay Attacks	The key stream is derived by sending the plain-text message.	
Bit-Flipping Attacks	Captures the frame and flips random bits in the data payload, modifies ICV, and sends to the user	
Extensible AP Replay	Capturing 802.IX Extensible Authentication Protocols (e.g., EAP Identity, Success, Failure) for later replay.	Wireless capture + injection tools between station and AP
RADIUS Replay	Capturing RADIUS Access-Accept or Reject messages for later replay	Ethernet capture + injection tools between AP and authentication server
Wireless Network Viruses	Viruses have their impact on the wireless network to a great extent. It allows the attacker with simplest ways for attacking on APs.	

WIRELESS NETWORK ATTACK

- **3) Confidentiality Attacks** - These attacks attempt to intercept confidential information sent over wireless associations, whether sent in the clear text or encrypted by Wi-Fi protocols.

Type of Attack	Description	Method and Tools
Eavesdropping	Capturing and decoding unprotected application traffic to obtain potentially sensitive information.	bsd-airtools, Ethereal, Ettercap, Kismet, commercial analyzers
Traffic Analysis	Implication of information from the observation of external traffic characteristics.	
Cracking WEP Key	Capturing data to recover a WEP key using brute force or Fluhrer-Mantin-Shamir (FMS) cryptanalysis.	Aircrack, AirSnort, chopchop, dwepcrack, WepAttack, WepDecrypt, WepLab

WIRELESS NETWORK ATTACK

▪ Confidentiality Attacks

Evil Twin AP	Masquerading as an authorized AP by beaconing the WLAN's service set identifier (SSID) to lure users.	pureAP, HermesAP, HostAP, OpenAP, Quetec, WifiBSD
Man-in-the- Middle Attack	Running traditional man-in-the- middle attack tools on an evil twin AP to intercept TCP sessions or SSL/SSH tunnels.	dsniff, Ettercap
Masquerading	Pretends to be an authorized user of a system in order to gain access to it.	Stealing login IDs and passwords, bypassing authentication mechanisms
Session Hijacking	Manipulating the network so the attacker's host appears to be the desired destination.	Manipulating
Honeypot Access Point	Setting its service identifier (SSID) to be the same as an access point at the local hotspot assumes the attacker as the legitimate hotspot.	Manipulating SSID

WIRELESS NETWORK ATTACK

- **4) Availability Attack** - These attacks aim at obstructing the delivery of wireless services to legitimate users, either by crippling those resources or by denying them access to WLAN resources. There are many attacks using which an attacker can obstruct the availability of wireless networks.

Type of Attack	Description	Method and Tools
Access Point Theft	Physically removing an AP from a public space.	Five finger discount
Denial of Service	Exploiting the CSMA/CA Clear Channel Assessment (CCA) mechanism to make a channel appear busy	An adapter that supports CW Tx mode, with a low-level utility to invoke continuous transmit
Beacon Flood	Generating thousands of counterfeit 802.11 beacons to make it hard for stations to find a legitimate AP.	FakeAP

WIRELESS NETWORK ATTACK

- **5)Authentication Attacks** - The objective of authentication attacks is to steal the identity of Wi-Fi clients, their personal information, login credentials, etc. to gain unauthorized access to network resources.

Type of Attack	Description	Method and Tools
Application Login Theft	Capturing user credentials (e.g., email address and password) from clear text application protocols.	Ace Password Sniffer, Dsniff, PHoss, WinSniffer
PSK Cracking	Recovering a WPA PSK from captured key handshake frames using a dictionary attack tool.	coWPAtty, KisMAC, wpa_crack, wpa-psk-bf
Shared Key Guessing	Attempting 802.11 Shared Key Authentication with guessed vendor default or cracked WEP keys.	WEP cracking tools