Sudan University of Sciences and Technology

College of Post Graduate Studies

Department of Electronics Engineering

Penetration Testing:

An Ethical Hacking Plan for Sudan University of Sciences and Technology Network

اختبار الإختراق: خطة القرصنة الشرعية لشبكة
جامعة السودان للعلوم والتكنولوجيا

November 201١

Research to be submitted in partial fulfillment for requirement of the degree of M.Sc. In Computer Engineering

By: *Sideeg Abdelbage Elsideeg*

Supervisor: *Dr. Mamoun M.A. Suliman*

Quran

أعوذ بالله من الشيطان الرجيم

بسم الله الرحمن الرحيم

(وقل رب زدني علما)

سورة طه (١١٤)

(وما أوتيتم من العلم إلا قليلا)

سورة الإسراء (٨٥)

صدق الله العظيم

DEDICATION

To the:

Precious thing in my life….

Parents

Torches of the light…..

Companions

Beautiful thing in my life

My Wife

ACKNOWLEDMENT

I am grateful to my supervisor

Dr. Mamoun M. A. Suliman

I value his insights, careful critiques and support through the project.

I would like to thank the Computer Center family represented by the director ***Dr. Mohammed Awad.*** *Also I would like to thank* ***Eng. Ahmed Ali Abdelraheem, Eng. Abdel Aziz Attahir, Eng. Musab Abdelmoniem, and Mohammed Mahajoop*** *for their assistance and support.*

I would like to express my warmest gratitude to my

family for their Continuous support.

ABSTRACT

Computer systems are usually vulnerable to attacks of malicious hackers and crackers. Penetration testing is a form of stress testing, since it provides a way to assess the computer system, and points out any vulnerabilities that can be exploited by hackers. It can find flaws in the security system. It is a valued assurance assessment tool that can be of great help for the system administrators, as it will help them tighten up their system security.

It is an area worthy of research because it crosses a lot of IT domains, technologies, specializations and disciplines. Despite its critical importance, we find a dearth of research in this area. This research is a small attempt to explore the depth of this area.

We began by discussing and differencing between ethical hacking and malicious hacking, and then we reviewed the methodologies used in work of penetration test as well as a detailed description of the tools used in it.

As performed this research (penetration testing) on SUST network. We thought, behaved and used the same tools and techniques that malicious hackers used. We followed Certified Ethical Hacker (Five stages) methodology using black box testing with remote network ethical hacking type. This application has been represented by two phases of the overall methodology that describes the five stages.

The result of the test is rich with information that describes the current security situation of the SUST network. The simplicity of the information gathering process and the amount of information and the number of findings with high level risk, showed how much SUST network is vulnerable to attacks of malicious hackers.

المستخلص

عادة ماتكون انظمة المعلومات وشبكاتها عرضة لعمليات القرصنة والمهددات الامنية الاخرى، لذا يعتبر اختبار محاكاة الاختراق من الوسائل المفيدة لتقييم مستوى أمن هذه الانظمة وذلك باكتشاف مناطق الضعف والثغرات وتحديد العيوب التي يمكن ان يستغلها قراصنة الحاسوب. تساعد نتائج هذا الاختبار مديري هذه الانظمة في تشديد أمن هذه الأنظمة.

يعتبر علم تقييم واختبار النظم بطريقة محاكاة الاختراق من العلوم المفيدة والتي تحتاج الى كثير بحث وتدقيق خصوصا من مدراء الانظمة ولانه عملية شامله لكل النظام الذي يحتوي على كثير من المجالات التقنية والتخصصات المعلوماتية المختلفة، ومع هذا نجد ندرة في البحوث والتطبيقات في هذا المجال. هذا البحث محاولة صغيرة لسبر أغوار هذا المجال بدراسة وتطبيق بعض منهجياته.

يبدأ هذا البحث بمناقشة وتبين الفرق بين الاختراق اوالقرصنة الشرعية والاختراق او القرصنة غير الشرعية، بعدها يستعرض الطرق والاساليب والادوات والبرامج المستخدمة في عمل هذه الاختراقات والقرصنة.

ولقد قمنا بعمل هذا الاختبار على شبكة جامعة السودان للعلوم التكنولوجيا، وذلك بمحاكاة لتفكير القراصنة واتباع اساليبهم واستخدام ادواتهم وبرامجهم، وذلك على خلفية نموذج شهادة القرصنة الشرعية (Certified Ethical Hacker CEH) ذو الخمسة مراحل وبإستخدام طريقة (اختبار الصندوق الأسود) وتقييم الانظمة عن بعد، وكل ذلك تم بطريقة شرعية. ولقد أجرينا في هذا البحث مرحلتين من المراحل الخمسة.

وكانت نتيجة هذا الاختبار غنية بالمعلومات التي تصف الوضع الأمني الراهن لشبكة جامعة السودان للعلوم والتكنولوجيا. كشفت عملية الاختبار عن سهولة عملية جمع البيانات والمعلومات وعن الكثير من الثغرات ذات مستوى التهديد العالي والتي تجعل شبكة جامعة السودان عرضة بشكل كبير لخطر المهددات والثغرات المكتشفة من قبل المخترقين.

TABLE of CONTENT

Contents

CHAPTER THREE

CHPTER FOUR

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ACK	Acknowledge.
AES	Advanced Encryption Standard
AfriNIC	African Network Information Center
ARP	Address Resolution Protocol
BOT	Web robot
BOTNETS	Group of BOTs
CEH	EC-Council's Certified Ethical Hacker
CESG	Communications and Electronics Security Group
CIFS	Common Internet File System
CREST	Council of Registered Ethical Security Testers
CSS	Cascading Style Sheets
CSW	Core switch
DMZ	Demilitarized Zone
DNS	Domain Name System
DoS	Denial Of Service
DDoS	Distributed Denial Of Service
FIN	Finish. No more transmissions.
FW	Firewall
GCHQ	The Government Communications Head quarters
GUI	Graphical User Interface
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
InfoSec	Information Security
IPC$	Inter Process Communication share
IPS	intrusion prevention system
IRC	Internet Relay Chat
ISECOM	the Institute for Security and Open Methodologies
ISN	initial Sequence Number
MAC address	Media Access Control address
MIB	Management Information Base
NetBIOS	Network Basic Input-Output System
NSA	United States National Security Agency
IAM	INFOSEC Assessment Methodology
OSSTMM	Open Source Security Testing Methodology Manual
OWASP	Open Source Web Application Security Project
POP3	Post Office Protocol 3
PSH	Push
R	Router

RAS	Remote Access Server
RC4	Rivest Cipher 4
RC5	Rivest Cipher 5
RFC	Request For Comment
RST	Reset
SAM	Security Accounts Manager
SMB	Server Message Block
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
SSH	Secure Shell
SUST	Sudan University of Science and Technology
SYN	Synchronize
TCP/IP	Transmission Control Protocol/Internet Protocol
TTL	time to live
URG	Urgent
URL	Uniform Resource Locator
VPN	Virtual Private Network
Webapp	Web Application
WVS	Web Vulnerability Scanner
XSS	Cross Site Scripting

CHAPTER ONE

INTRODUCTION

1. INTRODUCTION

1.1. GENERAL

Over the past few years, information security has evolved from a technology issue to a boardroom issue. Organizations are affected every day by security-related incidents such as network intrusions, viruses, or denial-of-service attacks. Some of these incidents are reported but many probably are not. Business is becoming increasingly dependent on technology and the Internet [1]. This is particularly true in larger Organizations, where the ability to communicate and access information is the lifeblood of the business.

Therefore in today’s networking there is an ongoing fight between system administrators and hackers. System administrators crawl under a lot of pressure in order to keep their networks and systems up and running. While they have a finite number of good allies fighting with them, their enemies have growing number of allies in their hands (and downloadable from the Internet). Administrators need to defend themselves by knowing their faults before the attackers do.

Penetration test and ethical hacking can help them to get this knowledge. Penetration test and ethical hacking enable administrators to know hackers tactics, motivations and strategies and use hackers techniques, utilities, and tools.

This research discuses ethical hacking and penetration testing and applies some of its techniques over SUST network.

1.2. TERMINOLOGIES

In this section a number of terms are explained as follows [2].

i. **Threat**: A threat is an environment or situation that could lead to a potential breach of security. Ethical hackers look for and prioritize threats when performing a security analysis.

ii. **Exploit:** In computer security, an exploit is a piece of software that takes advantage of a bug, glitch, or vulnerability, leading to unauthorized access, privilege escalation, or denial of service on a computer system. An exploit is a defined way to breach the security of an IT system through vulnerability. There are two methods of classifying exploits:

   a. A remote exploit works over a network and exploits security vulnerabilities without any prior access to the vulnerable system.

   b. A local exploit requires prior access to the vulnerable system to increase privileges.

iii. **Vulnerability:** is an existence of a software flaw, logic design, or implementation error that can lead to an unexpected and undesirable event executing bad or damaging instructions to the system.

iv. **Target of Evaluation:** A target of evaluation is a system, program, or network that is the subject of a security analysis or attack.

v. **Attack:** An attack occurs when a system is compromised based on vulnerability. Many attacks are perpetuated via an exploit.

vi. **Hackers**: a hacker who uses their hacking skills and toolset for destructive or offensive purposes such as disseminating viruses or performing DoS attacks to compromise or bring down systems and networks. No longer just looking for fun, these hackers are sometimes paid to damage corporate reputations or steal or reveal credit card information, while slowing business processes and compromising the integrity of the organization. Another name for a hacker is a malicious hacker.

vii. **Ethical Hackers:** Ethical hackers are security professionals or network penetration testers who use their hacking skills and toolsets for defensive and protective purposes. Ethical hackers test their network and systems security for vulnerabilities using the same tools that a hacker might use to compromise the network.

viii. **System Hardening:** minimizes the risk of a security breach to the system. Hardening the applications on a system minimizes the chance of a security breach using an application.

1.3. Need for Protection

"Penetration testing" can bring up many images and thoughts. However, in today's digital data era, where organization is being built with people's personal information, penetration testing is a necessary function for every security department. Penetration testing needs to be thought of and discussed. The organization's data needs to be protected. No longer can we rely on firewall implementations as the answer to privacy. Misconfigurations, exploits, updates, patches, backdoors, disgruntled employees - all are driving reasons for the need for penetration testing. There is no best way to find out where our weaknesses lie than by regularly performing penetration testing [3].

1.4. What is Penetration Testing

A penetration test or ethical hacking as The International Council of E-Commerce consultants says, is a method of evaluating the security of a computer system or network by simulating an attack from a malicious source, known as a Black Hat Hacker, or Cracker. The process involves an active analysis of the system for any potential vulnerabilities that could result from poor or improper system configuration, both known and unknown hardware or software flaws, and operational weaknesses in process or technical countermeasures. This analysis is carried out from the position of a potential attacker and can involve active exploitation of security vulnerabilities. Any security issues that are found will be presented to the system owner, together with an assessment of their impact, and often with a proposal for mitigation or a technical solution [2].

The intent of a penetration test is to determine the feasibility of an attack and the amount of business impact of a successful exploit, if discovered. It is a component of a full security audit [4].

1.5. Why We Do Penetration Testing:

From a business perspective, penetration testing helps safeguard organizations against failure, through:

1. Preventing financial loss through fraud (hackers, extortionists and disgruntled employees) or through lost revenue due to unreliable business systems and processes.
2. Proving due diligence and compliance to organization industry regulators, customers and shareholders. Non-compliance can result in organization losing business, receiving heavy fines, gathering bad Public relations or ultimately failing. At a personal level it can also mean the loss of the job, prosecution and sometimes even imprisonment.
3. Protecting organization brand by avoiding loss of consumer confidence and business reputation [5].

From an operational perspective, penetration testing helps shape information security strategy through:

Identifying vulnerabilities and quantifying their impact and likelihood so that they can be managed proactively; budget can be allocated and corrective measures implemented [5].

1.6. Risks

Penetration testing can be an invaluable technique to any organization's information security program. Basic white box penetration testing is often done as a fully automated inexpensive process. However, black box penetration testing is a labor intensive activity and requires expertise to minimize the risk to targeted systems. At a minimum, it may slow the organization's networks response time due to network scanning and vulnerability scanning. Furthermore, the possibility exists that systems may be damaged in the course of penetration testing and may be rendered inoperable, even though the organization benefits in knowing that the system could have been rendered inoperable by an intruder. Although this risk is mitigated by the use of experienced penetration testers, it can never be fully eliminated [4].

A penetration test should be carried out on any computer system that is to be deployed in a hostile environment, in particular any Internet facing site, before it is deployed. This provides a level of practical assurance that any malicious user will not be able to penetrate the system [4].

Black box penetration testing is useful in the cases where the tester assumes the role of an outside hacker and tries to intrude into the system without adequate knowledge of it.

1.7. Scope

Penetration test for SUST network including the web server and web application is done here. The methodology described later in chapter four to do this testing ethically is followed.

A remote network ethical hacking type (see 2.11 Types of Ethical Hacks) with black box testing (see 2.12 Testing Types) is performed. The ethical hacking test will be done over the Internet i.e. outside SUST defenses, assuming that no information about SUST network except its URL (www.sustech.edu) is available. The aim of the test is to break and find vulnerability in the outside defenses of the network, such as firewalls and web servers.

This research describes most of the related issues about hacking, hacking techniques, ethical hacking and penetration testing methodologies and the tools that we use to perform such as “Nslookup”, “WHOIS”, “Traceroute”, “VisualRoute” and “Acunetix”.

1.8. **Problem statement:**

Penetration testing for SUST network including the web server and web application to help SUST network administrators take preemptive measures against malicious attacks by pointing out any vulnerabilities and flaws in the security system that can be exploited by hackers.

1.9. **Research objectives:**

The major objective of the research performed in this thesis is to assess, study, and analyze vulnerabilities and flaws in SUST network security and recommend appropriate security controls.

1.10. **Research Methodology:**

- Depth Study and Analysis of hacking, hacking techniques, ethical hacking and penetration testing.
- Select suitable penetration testing methodology.
- Using one of the selected hacking tools (“Nslookup”, “WHOIS”, “Traceroute”, “VisualRoute” and “Acunetix” etc...)
- Study the finding.
- Propose and recommend the appropriate countermeasures.

1.11. **Expected Results:**

Provide SUST network administrators the SUST penetration testing reports and appropriate recommendations.

1.12. **Thesis Outlines:**

Chapter One: Introduction, we discuss need for ethical hacking and penetration testing and its benefits, we define the problem, scope this research and the methodology.

Chapter Tow: brief study and background about hacking concepts and ethical hacking.

Chapter Three: detailed study and analysis about hacking techniques that can be used against SUST network.

Chapter four: describes common penetration testing methodologies and the one we choose, and then we present the utilities and tools we will operate to do our test, finally take a brief description about SUST network.

Chapter Five: shows the testing findings and the results of the operation of the penetration testing tools against SUST network.

Chapter Six: Discuss testing results, conclusion and recommendations for future work.

CHAPTER TWO

HACKING CONCEPTS and BACKGROUND

2. HACKING CONCEPTS and BACKGROUND

2.1. Introduction

Hackers break into computer systems. Contrary to widespread myth, doing this doesn’t usually involve a mysterious leap of hacker brilliance, but rather persistence and the dogged repetition of a handful of fairly well-known tricks that exploit common weaknesses in the security of target systems. Accordingly, most crackers are only mediocre hackers [2].

Many ethical hackers detect malicious hacker activity as part of the security team of an organization tasked with defending against malicious hacking activity. In this chapter, we’ll discuss a number of concepts and of terms, hacking, Hacktivism and Ethical hacking, because we need them in this research.

2.2. Hacking

Computer hacking is broadly defined as intentionally accessing a computer without authorization or exceeding authorized access [6].

2.3. What Is Hacktivism?

Hacktivism refers to hacking for a cause. These hackers usually have a social or political agenda. Their intent is to send a message through their hacking activity while gaining visibility for their cause and themselves.

Hacktivism commonly targets government agencies, political groups, and any other entities [2].

2.4. Cracker (Hacker)

The term cracker describes a hacker who uses their hacking skills and toolset for destructive or offensive purposes such as disseminating viruses or performing DoS attacks to compromise or bring down systems and networks.

No longer just looking for fun, these hackers are sometimes paid to damage corporate reputations or steal or reveal credit-card information, while slowing business processes and compromising the integrity of the organization. Another name for a cracker is a malicious hacker [2].

2.5. Listing Different Types of Hacker Classes

Hackers can be divided into three groups: white hats, black hats, and grey hats. Ethical hackers usually fall into the white-hat category, but sometimes they're former grey hats who have become security professionals and who use their skills in an ethical manner [2].

2.5.1. **White hats:** are the good men, the ethical hackers who use their hacking skills for defensive purposes. White-hat hackers are usually security professionals with knowledge of hacking and the hacker toolset and who use this knowledge to locate weaknesses and implement countermeasures.

2.5.2. **Black hats:** are the bad men: the malicious hackers or crackers who use their skills for illegal or malicious purposes. They break into or otherwise violate the system integrity of remote machines, with malicious intent. Having gained unauthorized access, black-hat hackers destroy vital data, deny legitimate users service, and basically cause problems for their targets.

2.5.3. **Grey hats:** are hackers who may work offensively or defensively, depending on the situation. This is the dividing line between hacker and cracker.

2.6. Types of Hacking Technologies

Many methods and tools exist for locating vulnerabilities, running exploits, and compromising systems. Trojans, backdoors, sniffers, rootkits, exploits, buffer overflows, SQL injection and etc., are all technologies that can be used to hack a system or network. Most hacking tools exploit weaknesses in one of the following four areas [2]:

2.6.1. **Operating systems:** Many system's administrators install operating systems with the default settings, resulting in potential vulnerabilities that remain un-patched.

2.6.2. **Applications:** Applications usually aren't tested for vulnerabilities when developers are writing the code, which can leave many programming flaws that a hacker can exploit.

2.6.3. **Shrink-wrap code:** Many off-the-shelf programs come with extra features the common user isn't aware of, which can be used to exploit the system. One example is macros in Microsoft Word, which can allow a hacker to execute programs from within the application.

2.6.4. **Misconfigurations:** Systems can also be misconfigured or left at the lowest common security settings to increase ease of use for the user, which may result in vulnerability and an attack.

2.7. Ethical Hacking

Ethical Hacking is the practice of breaking into computers without malicious intent, simply to find security hazards and report them to the people responsible [4].

2.8. Ethical Hackers and Penetration Testers

Ethical hackers are usually security professionals or network penetration testers who use their hacking skills and toolsets for defensive and protective purposes. Ethical hackers test their network and systems security for vulnerabilities using the same tools that a hacker might use to compromise the network [2].

2.9. Penetration Test

Ethical hackers are motivated by different reasons, but their purpose is usually the same as that of crackers: They are trying to determine what an intruder can see on a targeted network or system, and what the hacker can do with that information. This process of testing the security of a system or network is known as a penetration test.

2.10. Vulnerability Research

Vulnerability research is the process of discovering vulnerabilities and design weaknesses that could lead to an attack on a system. Several websites and tools exist to aid the ethical hacker in maintaining a current list of vulnerabilities and possible exploits for their systems or networks. It's essential that a systems administrator keep current on the latest viruses, Trojans, and other common exploits in order to adequately protect their systems and network. Also, by becoming familiar with the newest threats, an administrator can learn how to detect, prevent, and recover from an attack [2].

2.11. Types of Ethical Hacks

Ethical hackers can use many different methods to breach an organization's security during a simulated attack or penetration test. The most common methods follow:

2.11.1.**Remote Network:** A remote network hack attempts to simulate an intruder launching an attack over the Internet. The ethical hacker tries to break or find vulnerability in the outside defenses of the network, such as firewall, proxy, or router vulnerabilities.

2.11.2.**Remote Dial-up Network:** A remote dial-up network hack tries to simulate an intruder launching an attack against the client's modem pools. War dialing is the process of repetitive dialing to find an open system and is an example of such an attack.

2.11.3.**Local Network:** A local network hack simulates someone with physical access gaining additional unauthorized access using the local network. The ethical hacker must gain direct access to the local network in order to launch this type of attack.

2.11.4.**Stolen Equipment:** A stolen equipment hack simulates theft of a critical information resource such as a laptop owned by an employee.

Information such as usernames, passwords, security settings, and encryption types can be gained by stealing a laptop.

2.11.5. **Social Engineering:** A social engineering attack checks the integrity of the organization's employees by using the telephone or face-to-face communication to gather information for use in an attack. Social engineering attacks can be used to acquire usernames, passwords, or other organizational security measures.

2.11.6. **Physical Entry:** A physical entry attack attempts to compromise the organization's physical premises. An ethical hacker who gains physical access can plant viruses, Trojans, rootkits, or hardware key loggers (physical device used to record keystrokes) directly on systems in the target network [1].

2.12. **Testing Types**

When performing a security test or penetration test, an ethical hacker utilizes one or more types of testing on the system. Each type simulates an attacker with different levels of knowledge about the target organization. These types are as follows:

2.12.1. **Black Box:** Involves performing a security evaluation and testing with no prior knowledge of the network infrastructure or system to be tested. Testing simulates an attack by a malicious hacker outside the organization's security perimeter.

2.12.2. **White Box:** Involves performing a security evaluation and testing with complete knowledge of the network infrastructure such as a network administrator would have.

2.12.3. **Grey Box:** Involves performing a security evaluation and testing internally. Testing examines the extent of access by insiders within the network [1].

2.13. **Penetration Test Report**

This report details the results of the hacking activity, the types of tests performed, and the hacking methods used. These results are compared against the work scheduled prior to the conduction of the test.

In the security evaluation phase any vulnerability identified is detailed, and countermeasures are suggested. The document of the report is usually delivered to the organization in hard copy format, for security reasons. In this research Chapter Six includes the report.

The details of the ethical hacking report must be kept confidential, because they highlight the organization's security risks and vulnerabilities. If this document falls into the wrong hands, the results could be disastrous for the organization [7].

2.14. **Understanding the Legal Implications of Hacking**

An ethical hacker should know the penalties of unauthorized hacking into a system. No ethical hacking activities associated with a network penetration test or security audit should begin until a signed legal document giving the ethical hacker express permission to perform the hacking activities is received from the target organization. Ethical hackers need to be judicious with their hacking skills and recognize the consequences of misusing those skills [2].

CHAPTER THREE

HACKING and ATTACK TECHNIQUES

3. HACKING and ATTACK TECHNIQUES

3.1. Introduction

Hackers use a number of techniques to attack and disrupt computer systems and networks. Attacks on computing systems can be of different types depending on the motives of the attacker, the attacker knowledge and the information about the targeted organization.

Hackers are continually developing new attack tools and hacking strategies to gain malicious access to systems and attack organization network, making it difficult for organizations to develop and implement the proper policies and procedures necessary to prevent hacker attacks.

3.2. Hacking and Attack Techniques

This chapter discusses the major kinds of hacking and attacks techniques; these techniques include and are related to [2]:

(i) Gathering information techniques (Footprinting, scanning and enumeration)
(ii) Cracking Password techniques
(iii) Trojans, Backdoors, Virus, and Worms
(iv) Sniffers
(v) Denial of Service
(vi) Session Hijacking
(vii) Social Engineering
(viii) Hacking Web Servers and Web Application Vulnerabilities
(ix) Web-Based Password-Cracking Techniques
(x) SQL Injection
(xi) Buffer Overflows
(xii) Wireless Hacking
(xiii) Physical Security

(xiv) Operating system Hacking

This section discuss these techniques in a general detail focusing on the research scope as mentioned earlier, discussion start with gathering information:

3.3. Gathering Information

Generally, a hacker spends 90 percent of the time profiling and gathering information on a target and 10 percent of the time launching the attack [1].

3.4. Describe the Information Gathering Methodology

Reconnaissance or Footprinting an organization involves gathering information regarding a potential target without the targeted individual's or organization's knowledge. This process is generally called information gathering. Information gathering can be broken into seven logical steps, see Figure 3.1, The Footprinting process is performed during the first two steps of unearthing initial information and locating the network range. The other information-gathering steps are called "Scanning and Enumeration."

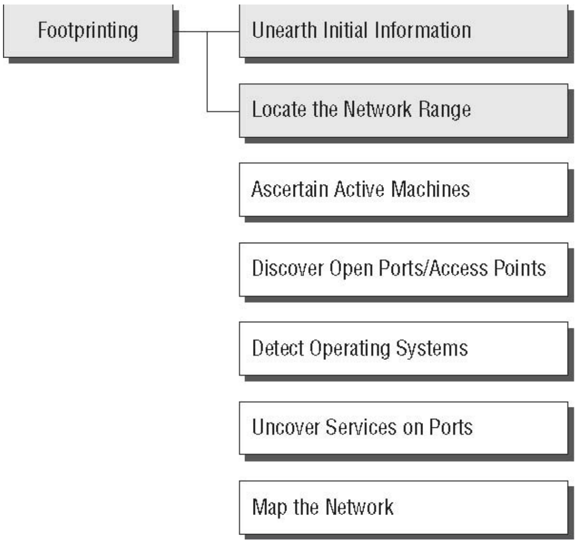

Footprinting
Unearth Initial Information
Locate the Network Range
Ascertain Active Machines
Discover Open Ports/Access Points
Detect Operating Systems
Uncover Services on Ports
Map the Network

Figure 3-1 Seven steps of information gathering [2]

Several techniques exist for the purpose of Information gathering and can be used by hackers to gather information about a potential target.

3.5. Footprinting

Footprinting is part of the preparatory pre-attack phase and involves accumulating data regarding a target's environment and architecture, usually for the purpose of finding ways to intrude into that environment. The purpose of this preparatory phase is to learn as much as we can about a system, its remote access capabilities, its ports and services, and any specific aspects of its security.

Footprinting begins by determining the target system, application, or physical location of the target. For example, the organization’s own web page may provide a personnel directory or a list of employee bios, which may prove useful if the hacker needs to use a social engineering attack to reach the objective. A hacker may also do a Google search or a Yahoo! People search to locate information about employees.

Some of the common techniques used for information gathering in Footprinting phase include the following:

(i) DNS Enumeration and Identify Types of DNS Records

(ii) Nslookup and DNSstuff

(iii) Whois and AfriNIC Lookups and Analyzing Whois Output

(iv) Finding the Address Range of the Network

(v) Using Traceroute

(vi) E-Mail Tracking

(vii) Web Spiders

3.6. DNS Enumeration

DNS enumeration is the process of locating all the DNS servers and their corresponding records for an organization. The organization may have both internal and external DNS servers that can give information such as usernames, computer names, and IP addresses of potential target systems.

3.6.1. Finding the Address Range of the Network

Every ethical hacker needs to understand how to find the network range and subnet mask of the target system. IP addresses are used to locate, scan, and connect to target systems. We can find IP addresses in Internet registries AfriNIC (African Network Information Center) in the Internet Assigned Numbers Authority (IANA) by doing (Nslookup, Whois).

An ethical hacker may also need to find the geographic location of the target system or network. This task can be accomplished by tracing the route a message takes as it's sent to the destination IP address. We can use tools like Traceroute, and VisualRoute, to identify the route to the target (we used VisualRoute .in this research).

Additionally, as we trace our target network, other useful information becomes available. For example, we can obtain internal IP addresses of host machines; even the Internet IP gateway of the organization may be listed. These addresses can then be used later in an attack or further scanning processes.

3.7. Using Traceroute in Footprinting

Traceroute is a packet-tracking tool that is available for most operating systems. It operates by sending an Internet Control Message Protocol (ICMP) echo to each hop (router or gateway) along the path, until the destination address is reached. When ICMP messages are sent back from the router, the time to live (TTL) is decremented by one for each router along the path. This allows a hacker to determine how many hops a router is from the sender.

One problem with using the Traceroute tool is that it times out (indicated by an asterisk) when it encounters a firewall or a packet-filtering router.

Although a firewall stops the Traceroute tool from discovering internal hosts on the network, it can alert an ethical hacker to the presence of a firewall; then, techniques for bypassing the firewall can be used.

Many hacking tools include a version of Traceroute. The Windows operating systems use the syntax tracert hostname to perform a Traceroute. Figure 5-4 .is an example of Traceroute output for a trace of www.sustech.edu

Because routers are generally named according to their physical location, the tracert results help us locate these devices.

The Hacking Tools available to do Traceroute include; “NeoTrace”, “VisualRoute”, and “VisualLookout” are all packet-tracking tools with a GUI or visual interface. They plot the path the packets travel on a map and can visually identify the locations of routers and other internetworking devices. These tools operate similarly to Traceroute and perform the same information gathering; however, they provide a visual representation of the results.

3.7.1. E-Mail Tracking

E-mail tracking programs allow the sender of an e-mail to know whether the recipient reads, forwards, modifies, or deletes an e-mail. Most e-mail–tracking programs work by appending a domain name to the e-mail address, such as http://www.sustech.edu/, a single-pixel graphic file that isn’t noticeable to the recipient is attached to the e-mail. Then, when an action is performed on the e-mail, this graphic file connects back to the server and notifies the sender of the action.

Available Hacking Tools include; Email Tracking Pro and MailTracking.com are tools that allow an ethical hacker to track e-mail messages. When using these tools to send an e-mail, forward an e-mail, reply to an e-mail, or modify an e-mail, the resulting actions and tracks of the original e-mail are logged. The sender is notified of all actions performed on the tracked e-mail by an automatically generated e-mail.

3.7.2. Web Spiders

Spammers and anyone else interested in collecting e-mail addresses from the Internet can use web spiders. A web spider combs websites collecting certain information such as email addresses. The web spider uses syntax such as the "@" symbol to locate email addresses then copies them into a list. These addresses are then added to a database and may be used later to send unsolicited e-mails. Web spiders can be used to locate all kinds of information on the Internet. A hacker can use a web spider to automate the information gathering process.

3.7.3. Social Engineering

Social engineering is a nontechnical method of breaking into a system or network. It's the process of deceiving users of a system and convincing them to do or give out information that can be used to defeat or bypass security mechanisms. Social engineering is important to understand because hackers can use it to attack the human element of a system and circumvent technical security measures. This method can be used to gather information before or during an attack. A social engineer commonly uses the telephone or Internet to trick people into revealing sensitive information or to get them to do something that is against the security policies of the organization. By this method, social engineers exploit the natural tendency of a person to trust their word, rather than exploiting computer security holes. It's generally agreed that users are the weak link in security; this principle is what makes social engineering possible.

The most dangerous part of social engineering is that organizations with authentication processes, firewalls, virtual private networks, and network-monitoring software are still wide open to attacks, because social engineering doesn't assault the security measures directly. Instead, a social-engineering attack bypasses the security measures and goes after the human element in an organization.

3.8. Scanning and Enumeration

Scanning and enumeration are the rest of the phases of gathering information and involve the hacker locating target systems or networks. Enumeration is the follow-on step once scanning is complete and is used to identify computer names, usernames, and shares. Scanning and enumeration are discussed together because many attacking tools perform both.

3.8.1. Scanning

Scanning involves taking the information discovered during reconnaissance and Footprinting and using it to examine the network. Data such as IP addresses, operating system, services, and installed applications can help the hacker decide which type of exploit to use in hacking a system. Tools that a hacker may employ during the scanning phase can include dialers, port scanners, network mappers, sweepers, and vulnerability scanners. Hackers are seeking any information that can help them perpetrate attack such as computer names, IP addresses, and user accounts.

3.8.2. Ethical hack Scanning Methodology

This methodology is the process by which a hacker scans the network. It ensures that no system or vulnerability is overlooked and that the hacker gathers all necessary information to perform an attack.

We'll look at the various stages of this scanning methodology throughout this research lightly, starting with the first three steps—checking for systems that are live and for open ports and service identification in the following section.

3.8.3. Network Scanning

Network scanning is a procedure for identifying active hosts on a network, either to attack them or as a network security assessment. Hosts are identified by their individual IP addresses. Network-scanning tools attempt to identify all the live or responding hosts on the network and their corresponding IP addresses.

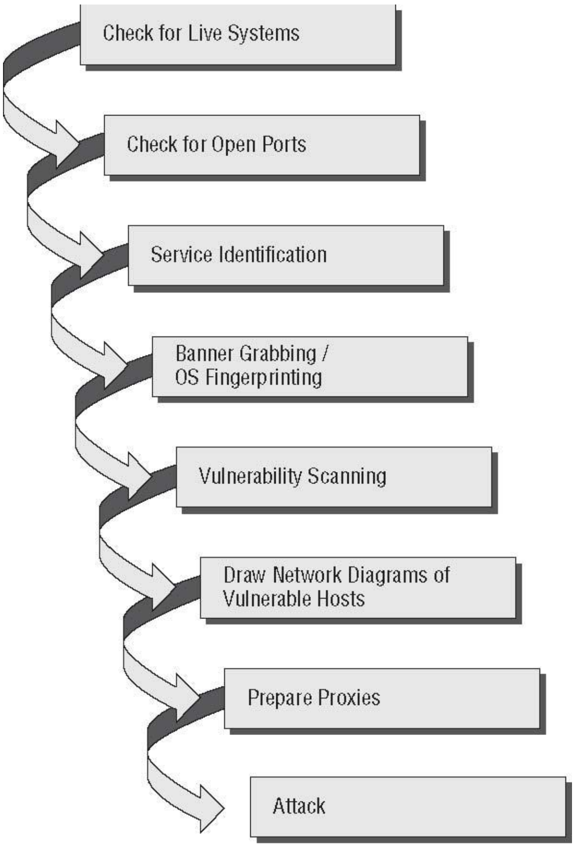


Figure 3-2 Scanning methodology [2]

The scanning methodology starts with checking for systems that are live on the network, meaning that they respond to probes or connection requests. The simplest, although not necessarily the most accurate, way to determine whether systems are live is to perform a ping sweep of the IP address range. All systems that respond with a ping reply are considered live on the network. Internet Control Message Protocol (ICMP) scanning is the process of sending an ICMP request or ping to all hosts on the network to determine which ones are up and responding to pings. One considerable problem with this method is that most firewall and proxy servers block ping responses so a hacker can’t accurately determine whether systems are available using a ping sweep alone, Almost any IDS or intrusion prevention system (IPS) will detect and alert the security administrator to a ping sweep occurring on the network. Another problem is that the computer must be powered on in order to be scanned.

3.8.4. Scanning Open ports and Service Identification

Port scanning is the process of identifying open and available TCP/IP ports on a system. Port-scanning tools enable a hacker to learn about the services available on a given system. Each service or application on a machine is associated with a well-known port number (C:\windows\ system32\drivers\etc\services).

Checking for open ports is the second step in the scanning methodology. Port scanning is the method used to check for open ports. The process of port scanning involves probing each port on a host to determine which ports are open. Port scanning generally yields more valuable information than a ping sweep about the host and vulnerabilities on the system.

Service identification is the third step in the scanning methodology; it’s usually performed using the same tools as port scanning. By identifying open ports, a hacker can usually also identify the services associated with that port number.

3.8.5. List TCP Communication Flag Types

TCP scan types are built on the TCP three-way handshake. TCP connections require a three-way handshake before a connection can be made and data transferred between the sender and receiver. Figure 3-3 details the steps of the TCP three-way handshake.

In order to complete the three-way handshake and make a successful connection between two hosts, the sender must send a TCP packet with the Synchronize (SYN) bit set. Then, the receiving system responds with a TCP packet with the synchronize (SYN) and acknowledge (ACK) bit set to indicate the host is ready to receive data. The source system sends a final packet with the Acknowledge (ACK) bit set to indicate the connection is complete and data is ready to be sent. Because TCP is a connection-oriented protocol, a process for establishing a connection (three-way handshake), restarting a failed connection, and finishing a connection is part of the protocol. These protocol notifications are called flags. TCP contains ACK, RST, SYN, URG, PSH, and FIN flags. The following list identifies the function of the TCP flags:

(i) SYN—Synchronize. Initiates a connection between hosts.

(ii) ACK—Acknowledge. Established connection between hosts.

(iii) PSH—Push. System is forwarding buffered data.

(iv) URG—Urgent. Data in packets must be processed quickly.

(v) FIN—Finish. No more transmissions.

(vi) RST—Reset. Resets the connection.

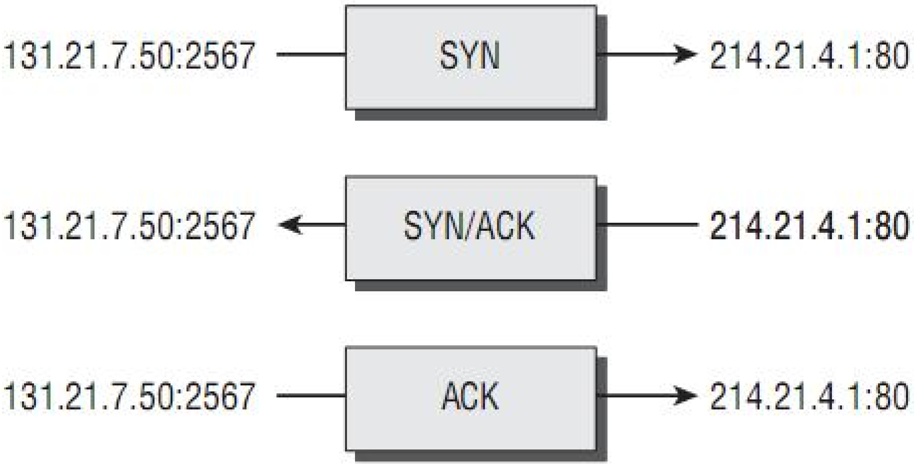


Figure 3-3 TCP three-way handshake [2]

A hacker can attempt to bypass detection by using flags instead of completing a normal TCP connection. The TCP scan types in Table 3.4 are used by some scanning tools to elicit a response from a system by setting one or more flags.

Table 3-1 TCP Scan Types

Flags sent by hacker	**XMAS Scan**
All flags set (ACK, RST, SYN, URG, PSH, FIN)	XMAS scan
FIN	FIN scan
No flags set	NULL Scan
SYN, then ACK	TCP connect / full-open scan
SYN, then RST	SYN scan / half-open scan

3.8.6. SYN, Stealth, XMAS, NULL, IDLE, and FIN Scans

SYN or stealth scan is also called a half-open scan because it doesn’t complete the TCP three-way handshake. A hacker sends a SYN packet to the target; if a SYN/ACK frame is received back, then it’s assumed that the target would complete the connect and the port is listening. If a RST is received back from the target, then it’s assumed that the port isn’t active or is closed. The advantage of the SYN stealth scan is that fewer IDS systems log this as an attack or connection attempt.

XMAS scans send a packet with the FIN, URG, and PSH flags set. If the port is open, there is no response; but if the port is closed, the target responds with a RST/ACK packet. XMAS scans work only on target systems that follow the RFC 793 implementation of TCP/IP and don’t work against any version of Windows.

FIN scan is similar to an XMAS scan but sends a packet with just the FIN flag set. FIN scans receive the same response and have the same limitations as XMAS scans.

NULL scan is also similar to XMAS and FIN in its limitations and response, but it just sends a packet with no flags set.

IDLE scan uses a spoofed IP address to send a SYN packet to a target. Depending on the response, the port can be determined to be open or closed. IDLE scans determine port scan response by monitoring IP header sequence numbers.

3.8.7. Banner Grabbing and OS Fingerprinting Techniques

Banner grabbing and operating system identification- which can also be defined as fingerprinting the TCP/IP stack- is the fourth step in the scanning methodology. The process of fingerprinting allows the hacker to identify particularly vulnerable or high value targets on the network. Banner grabbing is the process of opening a connection and reading the banner or response sent by the application. Many e-mail, FTP, and web servers will respond to a telnet connection with the name and version of the software. They aid a hacker in fingerprinting the OS and application software. For example, a Microsoft Exchange e-mail server would only be installed on Windows OS.

Active stack fingerprinting is the most common form of fingerprinting. It involves sending data to a system to see how the system responds. It’s based on the fact that various operating system vendors implement the TCP stack differently, and responses will differ based on the operating system. The responses are then compared to a database to determine the operating system. Active stack fingerprinting is detectable because it repeatedly attempts to connect with the same target system.

Passive stack fingerprinting is stealthier and involves examining traffic on the network to determine the operating system. It uses sniffing techniques instead of scanning techniques. Passive stack fingerprinting usually goes undetected by an IDS or other security system but is less accurate than active fingerprinting.

3.8.8. Vulnerability Scanning

Vulnerability scanning is the process of proactively identifying the vulnerabilities of computer systems on a network. Generally, a vulnerability scanner first identifies the operating system and version number, including patches and service packs that may be installed. Then, the vulnerability scanner identifies weaknesses or vulnerabilities in the operating system. During the later attack phase, a hacker can exploit those weaknesses in order to gain access to the system. Network and vulnerability scanning can usually be detected as well, because the scanner must interact with the target system over the network.

3.8.9. Drawing Network Diagrams of Vulnerable Hosts

A number of network-management tools can assist us with this step. Such tools are generally used to manage network devices but can be turned against security administrators by enterprising hackers.

3.8.10.Using Proxy Servers Launching an Attack

Preparing proxy servers is the last step in the scanning methodology. A proxy server is a computer that acts as an intermediary between the hacker and the target computer.

Using a proxy server can allow a hacker to become anonymous on the network. The hacker first makes a connection to the proxy server and then requests a connection to the target computer via the existing connection to the proxy. Essentially, the proxy requests access to the target computer not the hacker’s computer. This lets hacker surf the web anonymously or otherwise hides their attack.

3.8.11.Anonymizers

Anonymizers are services that attempt to make web surfing anonymous by utilizing a website that acts as a proxy server for the web client. The Anonymizers removes all the identifying information from a user's computers while the user surfs the Internet, thereby ensuring the privacy of the user.

To visit a website anonymously, the hacker enters the website address into the Anonymizers software, and the Anonymizers software makes the request to the selected site. All requests and web pages are relayed through the Anonymizers site, making it difficult to track the actual requester of the webpage.

3.8.12.HTTP Tunneling Techniques

A popular method of bypassing a firewall or IDS is to tunnel a blocked protocol (such as SMTP) through an allowed protocol (such as HTTP). Almost all IDS and firewalls act as a proxy between a client's PC and the Internet and pass only the traffic defined as being allowed.

Most organizations allow HTTP traffic because it's usually benign web access. However, a hacker using a HTTP tunneling tool can subvert the proxy by hiding potentially destructive protocols, such as IM or chat, within an innocent-looking protocol packet.

3.8.13.IP Spoofing Techniques

A hacker can spoof an IP address when scanning target systems to minimize the chance of detection. One drawback of spoofing an IP address is that a TCP session can't be successfully completed.

Source routing lets an attacker specify the route that a packet takes through the Internet. This can also minimize the chance of detection by bypassing IDS and firewalls that may block or detect the attack. Source routing uses a reply address in the IP header to return the packet to a spoofed address instead of the attacker's real address.

To detect IP address spoofing, we can compare the time to live (TTL) values: The attacker's TTL will be different from the spoofed address's real TTL.

3.9. Enumeration

Enumeration occurs after scanning and is the process of gathering and compiling usernames, machine names, network resources, shares, and services. It also refers to actively querying or connecting to a target system to acquire this information.

The objective of enumeration is to identify a user account or system account for potential use in hacking the target system. It isn’t necessary to find a system administrator account, because most account privileges can be escalated to allow the account more access than was previously granted.

Many hacking tools are designed for scanning IP networks to locate NetBIOS name information. For each responding host, the tools list IP address, NetBIOS computer name, logged-in username, and MAC address information.

3.9.1. Null Sessions

A null session occurs when we log in to a system with no username or password. NetBIOS null sessions are a vulnerability found in the Common Internet File System (CIFS) or SMB, depending on the operating system -Microsoft Windows uses SMB, and Unix/Linux systems use CIFS- Once a hacker has made a NetBIOS connection using a null session to a system, they can easily get a full dump of all usernames, groups, shares, permissions, policies, services and more using the Null user account. The SMB and NetBIOS standards in Windows include APIs that return information about a system via TCP port 139.

One method of connecting a NetBIOS null session to a Windows system is to use the hidden Inter Process Communication share (IPC$). This hidden share is accessible using the net use command. The net use command is a built-in Windows command that connects to a share on another computer. The empty quotation marks ("") indicate that we want to connect with no username and no password.

Once the net use command has been successfully completed, the hacker has a channel over which to use other hacking tools and techniques.

3.9.2. SNMP Enumeration

SNMP enumeration is the process of using SNMP to enumerate user accounts on a target system. SNMP employs two major types of software components for communication: the SNMP agent, which is located on the networking device; and the SNMP management station, which communicates with the agent.

Almost all network infrastructure devices, such as routers and switches and including Windows systems, contain an SNMP agent to manage the system or device. The SNMP management station sends requests to agents, and the agents send back replies. The requests and replies refer to configuration variables accessible by agent software. Management stations can also send requests to set values for certain variables. Traps let the management station know that something significant has happened in the agent software such as a reboot or an interface failure. Management Information Base (MIB) is the database of configuration variables, which resides on the networking device.

SNMP has two passwords we can use to access and configure the SNMP agent from the management station. The first is called a read community string. This password lets us view the configuration of the device or system. The second is called the read/write community string; it’s for changing or editing the configuration on the device. Generally, the default read community string is public and the default read/write community string is private. A common security loophole occurs when the community strings are left at the default settings: A hacker can use these default passwords to view or change the device configuration.

3.9.3. The Steps Involved in Performing Enumeration

Hackers need to be methodical in their approach to hacking. The following steps are an example of those a hacker might perform in preparation for hacking a target system:

1 Extract usernames using enumeration.

2 Gather information about the host using null sessions.

3 Perform OS (Windows) enumeration using the Superscan tool.

4 Acquire the user accounts using the tool GetAcct.

5 Perform SNMP port scanning.

3.10. Password-Cracking Techniques

Many hacking attempts start with attempting to crack passwords. Passwords are the key piece of information needed to access a system. Users, when creating passwords, often select passwords that are easy to being cracked. Many reuse passwords or choose one that's simple to help them remember it. Because of this human factor, most password cracking is successful; it can be the launching point for escalating privileges, executing applications, hiding files, and covering tracks. Passwords may be cracked manually or with automated tools such as a dictionary or brute-force method.

Manual password cracking involves attempting to log on with different passwords. The hacker follows these steps:

(i) Find a valid user account (such as Administrator or Guest).

(ii) Create a list of possible passwords.

(iii) Rank the passwords from high to low probability.

(iv) Key in each password.

(v) Try again until a successful password is found.

A hacker can also create a script file that tries each password in a list. This is still considered manual cracking, but it's time consuming and not usually effective.

A more efficient way of cracking a password is to gain access to the password file on a system. Most systems hash (one-way encrypt) a password for storage on a system. During the logon process, the password entered by the user is hashed using the same algorithm and then compared to the hashed passwords stored in the file. A hacker can attempt to gain access to the hashing algorithm stored on the server instead of trying to guess or otherwise identify the password. If the hacker is successful, they can decrypt the passwords stored on the server.

A hacker may use different types of attacks in order to identify a password and gain further access to a system. The types of password attacks are as follows: **Passive online** Eavesdropping on network password exchanges. Passive online attacks include sniffing, man-in-the-middle, and replay attacks. **Active online** Guessing the Administrator password. Active online attacks include automated password guessing. **Offline** Dictionary, hybrid, and brute-force attacks. **Nonelectronic** Shoulder surfing, keyboard sniffing, and social engineering. We would not look at these attacks in this research.

3.10.1.Key loggers and Other Spyware Technologies

Keystroke loggers (keyloggers) can be implemented either using hardware or software. Hardware keyloggers are small hardware devices that connect the keyboard to the PC and save every keystroke into a file or in the memory of the hardware device. In order to install a hardware keylogger, a hacker must have physical access to the system.

Software keyloggers are pieces of stealth software that sit between the keyboard hardware and the operating system, so that they can record every keystroke. Software keyloggers can be deployed on a system by Trojans or viruses.

3.11. **Sniffers**

A sniffer can be a packet-capturing or frame-capturing tool. It intercepts traffic on the network and displays it in either a command-line or GUI format for a hacker to view. Some sophisticated sniffers interpret the packets and can reassemble the packet stream into the original data, such as an e-mail or a document.

Sniffers are used to capture traffic sent between two systems. Depending on how the sniffer is used and the security measures in place, a hacker can use a sniffer to discover user-names, passwords, and other confidential information transmitted on the network. Several hacking attacks and various hacking tools require the use of a sniffer to obtain important information sent from the target system.

3.11.1.Protocols Susceptible to Sniffing

Sniffer software works by capturing packets not destined for the system's MAC address but rather for a target's destination MAC address. This is known as promiscuous mode. Normally, a system on the network reads and responds only to traffic sent directly to its MAC address. In promiscuous mode, the system reads all traffic and sends it to the sniffer for processing. Promiscuous mode is enabled on a network card with the installation of special driver software. Many of the hacking tools for sniffing include a promiscuous-mode driver to facilitate this process.

Any protocols that don't encrypt data are susceptible to sniffing. Protocols such as HTTP, POP3, Simple Network Management Protocol (SNMP), and FTP are most commonly captured using a sniffer and viewed by a hacker to gather valuable information such as usernames and passwords.

3.11.2.Active and Passive Sniffing

There are two different types of sniffing: passive and active. Passive sniffing involves listening and capturing traffic, and is useful in a network connected by hubs; active sniffing involves launching an Address Resolution Protocol (ARP) spoofing or traffic-flooding attack against a switch in order to capture traffic. As the names indicate, active sniffing is detectable but passive sniffing isn’t.

In networks that use hubs or wireless media to connect systems, all hosts on the network can see all traffic; therefore a passive packet sniffer can capture traffic going to and from all hosts connected via the hub. A switched network operates differently. The switch looks at the data sent to it and tries to forward packets to their intended recipients based on MAC address. The switch maintains a MAC table of all the systems and the port numbers to which they’re connected. This enables the switch to segment the network traffic and send traffic only to the correct destination MAC addresses. A switch network has greatly improved throughput and is more secure than a shared network connected via hubs.

3.11.3.ARP Poisoning

ARP allows the network to translate IP addresses into MAC addresses. When one host using TCP/IP on a LAN tries to contact another, it needs the MAC address or hardware address of the host it’s trying to reach. It first looks in its ARP cache to see if it already has the MAC address; if it doesn’t, it broadcasts an ARP request asking, “Who has the IP address I’m looking for?” If the host that has that IP address hears the ARP query, it responds with its own MAC address, and a conversation can begin using TCP/IP.

ARP poisoning is a technique that’s used to attack an Ethernet network and that may let an attacker sniff data frames on a switched LAN or stop the traffic altogether. ARP poisoning utilizes ARP spoofing where the purpose is to send fake, or spoofed, ARP messages to an Ethernet LAN. These frames contain false MAC addresses that confuse network devices such as network switches. As a result, frames intended for one machine can be mistakenly sent to another (allowing the packets to be sniffed) or to an unreachable host (a Denial of Service [DoS] attack). ARP spoofing can also be used in a man-in-the-middle attack in which all traffic is forwarded through a host by means of ARP spoofing and analyzed for passwords and other information.

To prevent ARP spoofing, permanently add the MAC address of the gateway to the ARP cache on a system. You can do this on a Windows system by using the ARP s command at the command line and appending the gateway's IP and MAC addresses. Doing so prevents a hacker from overwriting the ARP cache to perform ARP spoofing on the system but can be difficult to manage in a large environment because of the number of systems. In an enterprise environment, port based security can be enabled on a switch to allow only one MAC address per switch port.

3.11.4.Understand MAC Flooding

A packet sniffer on a switched network can't capture all traffic as it can on a hub network; instead, it captures either traffic coming from or traffic going to the system. It's necessary to use an additional tool to capture all traffic on a switched network. There are essentially two ways to perform active sniffing and make the switch send traffic to the system running the sniffer: ARP spoofing and flooding.

As mentioned earlier, ARP spoofing involves taking on the MAC address of the network gateway and consequently receiving all traffic intended for the gateway on the sniffer system. A hacker can also flood a switch with so much traffic that it stops operating as a switch and instead reverts to acting as a hub, sending all traffic to all ports. This active sniffing attack allows the system with the sniffer to capture all traffic on the network.

3.11.5.DNS Spoofing Techniques

DNS spoofing (or DNS poisoning) is a technique that tricks a DNS server into believing it has received authentic information when in reality it hasn't. Once the DNS server has been poisoned, the information is generally cached for a while, spreading the effect of the attack to the users of the server. When a user requests a certain website URL, the address is looked up on a DNS server to find the corresponding IP address. If the DNS server has been compromised, the user is redirected to a website other than the one that was requested, such as a fake website.

To perform a DNS attack, the attacker exploits a flaw in the DNS server software that can make it accept incorrect information. If the server doesn’t correctly validate DNS responses to ensure that they come from an authoritative source, the server ends up caching the incorrect entries locally and serving them to users that make subsequent requests.

This technique can be used to replace arbitrary content for a set of victims with content of an attacker’s choosing. For example, an attacker poisons the IP addresses DNS entries for a target website on a given DNS server, replacing them with the IP address of a server the hacker controls. The hacker then creates fake entries for files on this server with names matching those on the target server. These files may contain malicious content, such as a worm or a virus. A user whose computer has referenced the poisoned DNS server is tricked into thinking the content comes from the target server and unknowingly downloads malicious content.

The types of DNS spoofing techniques are as follows:

(i) Intranet spoofing—acting as a device on the same internal network

(ii) Internet spoofing—acting as a device on the Internet

(iii) Proxy server DNS poisoning—modifying the DNS entries on a proxy server so the user is redirected to a different host system

(iv) DNS cache poisoning—modifying the DNS entries on any system so the user is redirected to a different host

3.11.6.Describe Sniffing Countermeasures

The best security defense against a sniffer on the network is encryption. Although encryption won’t prevent sniffing, it renders any data captured during the sniffing attack useless because hacker can’t interpret the information. Encryption such as AES and RC4 or RC5 can be utilized in VPN technologies and is a common method to prevent sniffing on a network.

3.12. **Denial of Service**

During a Denial of Service (DoS) attack, a hacker renders a system unusable or significantly slows the system by over loading resources or preventing legitimate users from accessing the system. These attacks can be perpetrated against an individual system or an entire network and are usually successful in their attempts.

Session hijacking is a hacking method that creates a temporary DoS for an end user when an attacker takes over the session. Session Hijacking is used by hackers to take over a current session after the user has established an authenticated session. Session hijacking can also be used to perpetrate a man-in-the-middle attack when the hacker steps between the server and legitimate client and intercepts all traffic.

3.12.1.Types of DoS Attacks

There are two main categories of DoS attacks. DoS attacks can be either sent by a single system to a single target (simple DoS) or sent by many systems to a single target (DDoS).

The goal of DoS isn’t to gain unauthorized access to machines or data, but to prevent legitimate users of a service from using it. A DoS attack may do the following:

(i) Flood a network with traffic, thereby preventing legitimate network traffic.

(ii) Disrupt connections between two machines, thereby preventing access to a service.

(iii) Prevent a particular individual from accessing a service.

(iv) Disrupt service to a specific system or person.

Different tools use different types of traffic to flood a victim, but the result is the same: A service on the system or the entire system is unavailable to a user because it’s kept busy trying to respond to an exorbitant number of requests.

A DoS attack is usually an attack of last resort. It’s considered an unsophisticated attack because it doesn’t gain the hacker access to any information but rather annoys the target and interrupts their service. DoS attacks can be destructive and have a substantial impact when sent from multiple systems at the same time (DDoS attacks).

DDoS attacks can be perpetrated by BOTs and BOTNETS, which are compromised systems that an attacker uses to launch the attack against the end victim. The system or network that has been compromised is a secondary victim, whereas the DoS and DDoS attacks flood the primary victim or target.

3.12.2.DDoS Attacks

Distributed Denial of Service (DDoS) is an advanced version of the DoS attack. Like DoS, DDoS also tries to deny access to services running on a system by sending packets to the destination system in a way that the destination system can’t handle. The key of a DDoS attack is that it relays attacks from many different hosts (which must first be compromised), rather than from a single host like DoS. DDoS is a large-scale, coordinated attack on a victim system.

The services under attack are those of the primary victim; the compromised systems used to launch the attack are secondary victims. These compromised systems, which send the DDoS to the primary victim, are sometimes called zombies or BOTs. They’re usually compromised through another attack and then used to launch an attack on the primary victim at a certain time or under certain conditions. It can be difficult to track the source of the attacks because they originate from several IP addresses. Normally, DDoS consists of three parts:

(i) Master/Handler

(ii) Slave/secondary victim/zombie/agent/BOT/BOTNET

(iii) Victim /primary victim

The master is the attack launcher. A slave is a host that is compromised by and controlled by the master. The victim is the target system. The master directs the slaves to launch the attack on the victim system.

DDoS is done in two phases. In the intrusion phase, the hacker compromises weak systems in different networks around the world and installs DDoS tools on those compromised slave systems. In the DDoS attack phase, the slave systems are triggered to cause them to attack the primary victim.

3.12.3.BOTs/BOTNETs

A BOT is short for web robot and is an automated software program that behaves intelligently. Spammers often use BOTs to automate the posting of spam messages on news-groups or the sending of emails. BOTs can also be used as remote attack tools. Most often, BOTs are web software agents that interface with web pages. For example, web crawlers (spiders) are web robots that gather web-page information.

The most dangerous BOTs are those that covertly install themselves on users’ computers for malicious purposes.

Some BOTs communicate with other users of Internet-based services via instant messaging, Internet Relay Chat (IRC) or another web interface. These BOTs allow IRC users to ask questions in plain English and then formulate a proper response. Such BOTs can often handle many tasks, including reporting weather, providing zip-code information, listing sports scores, converting units of measure, such as currency, and so on.

A BOTNET is a group of BOT systems. BOTNETs serve various purposes, including DDoS attacks, creation or misuse of Simple Mail Transfer Protocol (SMTP) mail relays for spam, Internet Marketing fraud, the theft of application serial numbers, login IDs, and financial information such as credit card numbers. Generally a BOTNET refers to a group of compromised systems running a BOT for the purpose of launching a coordinated DDOS attack.

3.12.4.Smurf Attack

A smurf attack sends a large amount of ICMP echo (ping) traffic to a broadcast IP address with the spoofed source address of a victim. Each secondary victim's host on that IP network replies to the ICMP echo request with an echo reply, multiplying the traffic by the number of hosts responding. On a multi-access broadcast network, hundreds of machines might reply to each packet. This creates a magnified DoS attack of ping replies, flooding the primary victim. IRC servers are the primary victim of smurf attacks on the Internet.

3.12.5.SYN Flooding

A SYN flood attack sends TCP connection requests faster than a machine can process them. The attacker creates a random source address for each packet and sets the SYN flag to request a new connection to the server from the spoofed IP address. The victim responds to the spoofed IP address and then waits for the TCP confirmation that never arrives. Consequently, the victim's connection table fills up waiting for replies; after the table is full, all new connections are ignored. Legitimate users are ignored, as well, and can't access the server. Some of the methods to prevent SYN Flood attacks are SYN cookies, RST cookies, Micro Blocks, and Stack Tweaking.

Ping of Death is an attack that can cause a system to lock up by sending multiple IP packets, which will be too large for the receiving system when reassembled. Ping of Death can cause a DoS to clients trying to access the server that has been a victim of the attack.

3.13. **Session Hijacking**

Session hijacking is when a hacker takes control of a user session after the user has successfully authenticated with a server. Session hijacking involves an attack identifying the current session IDs of a client/server communication and taking over the client’s session. Session hijacking is made possible by tools that perform sequence-number prediction.

3.13.1.Spoofing vs. Hijacking

Spoofing attacks are different from hijacking attacks. In a spoofing attack, the hacker performs sniffing and listens to traffic as it’s passed along the network from sender to receiver. The hacker then uses the information gathered to spoof or uses an address of a legitimate system. Hijacking involves actively taking another user offline to perform the attack. The attacker relies on the legitimate user to make a connection and authenticate. After that, the attacker takes over the session, and the valid user’s session is disconnected.

Session hijacking involves the following three steps to perpetuate an attack:

Tracking the session the hacker identifies an open session and predicts the sequence number of the next packet.

Desynchronizing the connection the hacker sends the valid user’s system a TCP reset (RST) or finish (FIN) packet to cause them to close their session.

Injecting the attacker's packet the hacker sends the server a TCP packet with the predicted sequence number, and the server accepts it as the valid user's next packet.

3.13.2.List the Types of Session Hijacking

Hackers can use two types of session hijacking: active and passive. The primary difference between active and passive hijacking is the hacker's level of involvement in the session. In an active attack, an attacker finds an active session and takes over the session by using tools that predict the next sequence number used in the TCP session.

In a passive attack, an attacker hijacks a session and then watches and records all the traffic that is being sent by the legitimate user. Passive session hijacking is really no more than sniffing. It's used to gather information such as passwords and then to use that information later to authenticate as a separate session.

3.13.3.TCP Concepts: Three-Way Handshake

One of the key features of TCP is reliability and ordered delivery of packets. To accomplish this, TCP uses acknowledgment (ACK) packets and sequence numbers. Manipulating these numbers is the basis for TCP session hijacking. To understand session hijacking, let's review the TCP three-way handshake described in earlier chapters:

1 The valid user initiates a connection with the server. This is accomplished by the valid user sending a packet to the server with the SYN bit set and the user's initial Sequence Number (ISN).

2 The server receives this packet and sends back a packet with the SYN bit set and an ISN for the server, plus the ACK bit set identifying the user's ISN incremented by a value of one.

3 The valid user acknowledges the server by returning a packet with the ACK bit set and incrementing the servers ISN by one.

This connection can be closed from either side due to a timeout, or upon receipt of a package with the FIN or RST flag set.

Upon receipt of a packet with the RST flag set, the receiving system closes the connection, and any incoming packets for the session are discarded. If the FIN flag is set in a packet, then the receiving system goes through the process of closing the connection, and any packets received while closing the connection are still processed. Sending a packet with the FIN or RST flag set is the most common method hijackers use to close the client's session with the server and take over the session by acting as the client.

3.13.4.Sequence Prediction

TCP is a connection-oriented protocol, responsible for reassembling streams of packets into their original intended order. Therefore, each packet must have a unique number known as a sequence number (SN). Every packet has to be assigned a unique session number that enables the receiving machine to reassemble the stream of packets into their original and intended order; this unique number is known as a sequence number. If the packets arrive out of order, as happens regularly over the Internet, then the sequence number is used to stream the packets correctly. As just illustrated, the system initiating a TCP session transmits a packet with the SYN bit set. This is called a synchronize packet and includes the client's Initial Sequence Number (ISN). The ISN is a pseudo-randomly generated number with over 4 billion possible combinations, yet it is statistically possible for it to repeat.

When the acknowledgment (ACK) packet is sent, each machine uses the SN from the packet being acknowledged, plus an increment. This not only properly confirms receipt of a specific packet, but also tells the sender the next expected TCP packet sequence number. Within the three-way handshake, the increment value is 1. In normal data communications, the increment value equals the size of the data in bytes (for example, if you transmit 45 bytes of data, the ACK responds using the incoming packet's SN plus 45.

Figure (3.4) illustrates the sequence numbers and acknowledgments used during the TCP three-way handshake.

Hacking tools used to perform session hijacking do sequence number prediction. In order to successfully perform a TCP sequence prediction attack, the hacker must sniff the traffic between two systems. Next, the hacker or the hacking tool must successfully guess the sequence number or locate an ISN to calculate the next sequence number. This can be more difficult than it sounds, because packets travel very fast.

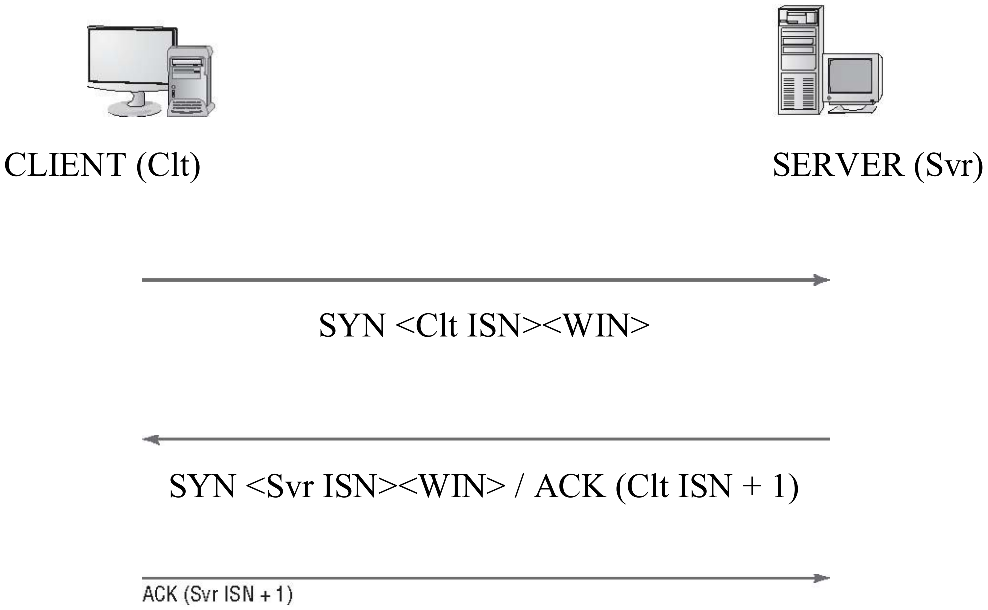


Figure 3-4 Sequence Numbers and Acknowledgment During the TCP Three-Way Handshake [2]

When the hacker is unable to sniff the connection, it becomes much more difficult to guess the next sequence number. For this reason, most session-hijacking tools include features to permit sniffing the packets to determine the sequence numbers.

Hackers generate packets using a spoofed IP address of the system that had a session with the target system. The hacking tools issue packets with the sequence numbers that the target system is expecting. But the hacker’s packets must arrive before the packets from the trusted system whose connection is being hijacked. This is accomplished by flooding the trusted system with packets or sending a RST packet to the trusted system so that it is unavailable to send packets to the target system.

What Are the Steps in Performing Session Hijacking?

In summary, session hijacking involves the following three steps to perpetuate the attack:

Tracking the session the hacker identifies an open session and predicts the sequence number of the next packet.

Desynchronizing the connection the hacker sends the valid user’s system a TCP reset (RST) or finish (FIN) packet to cause them to close their session. Alternately the hacker can use a DoS tool to disconnect the user from the server.

Injecting the attacker’s packet the hacker sends the server a TCP packet with the predicted sequence number, and the server accepts it as the valid user’s next packet.

3.13.5.Dangers Posed by Session Hijacking

TCP session hijacking is a dangerous attack: Most systems are vulnerable to it, because they use TCP/IP as their primary communication protocol. Newer operating systems have attempted to secure themselves from session hijacking by using pseudorandom number generators to calculate the ISN, making the sequence number harder to guess. However, this security measure is ineffective if the attacker is able to sniff packets, which gives all the information required to perform this attack.

The following are reasons why it’s important to be aware of session hijacking:

- Most computers are vulnerable.

- Few countermeasures are available to adequately protect against it.
- Session hijacking attacks are simple to launch.
- Hijacking is dangerous because of the information that can be gathered during the attack.

3.14. **Hacking Web Servers**

Web servers and web applications have a very high potential to be compromised. The primary reason for this is that the systems that run web server software must be publicly available on the Internet. Once a web server has been compromised, the system can provide hackers another door into the network. Not only the web server software, but also applications that run on the web server, are open to attack and can be exploited. Due to their function, web servers are more accessible than other systems and less protected, so they're easier to exploit.

3.14.1.Hacking Web Servers

Web servers, like other systems, can be compromised by a hacker. The following vulnerabilities are most commonly exploited in web servers:

- Misconfigurations of the web server software
- Operating system or application bugs, or flaws in programming code
- Vulnerable default installation of operating system and web server software, and/or lack of patch management to update operating system or web server software
- Lack of or not following proper security policies and procedures

Hackers exploit these vulnerabilities to gain access to the web server. Because web servers are located in a Demilitarized Zone (DMZ), which is a publicly accessible area between two packet filtering devices, and can be more easily accessed by the organization's client systems, an exploit of a web server offers a hacker easier access to internal systems or databases.

3.14.2.Attacks against Web Servers

The most visible type of attack against web servers is defacement. Hackers deface websites for sheer joy and an opportunity to enhance their reputations. Defacing a website means the hacker exploits vulnerability in the operating system or web server software and then alters the website files to show that the site has been hacked. Often the hacker displays their hacker name on the website's home page.

Common website attacks that enable a hacker to deface a website include the following:

(i) Capturing administrator credentials through man-in-the-middle attacks
(ii) Revealing an administrator password through a brute-force attack
(iii) Using a DNS attack to redirect users to a different web server
(iv) Compromising an FTP or e-mail server
(v) Exploiting web application bugs that result in a vulnerability
(vi) Misconfigurings web shares
(vii) Taking advantages of weak permissions
(viii) Rerouting a client after a firewall or router attack
(ix) Using SQL injection attacks (if the SQL server and web server are the same system)
(x) Using Telnet or Secure Shell (SSH) intrusion
(xi) Carrying out URL poisoning, which redirects the user to a different URL
(xii) Using web server extension or remote service intrusion
(xiii) For cookie-enabled security—Intercept the communication between the client and the server and change the cookie to

make the server believe that there is a user with higher privileges

3.15. **Web Application Vulnerabilities**

3.15.1.How Web Applications Work

Web applications are programs that reside on a web server to give the user functionality beyond just a website. Database queries, webmail, discussion groups, and blogs are all examples of web applications.

A web application uses a client/server architecture, with a web browser as the client and the web server acting as the application server. JavaScript is a popular way to implement web applications. Since web applications are widely implemented, any user with a web browser can interact with most site utilities.

3.15.2.Objectives of Web Application Hacking

The purpose of hacking a web application is to gain confidential data. Web applications are critical to the security of a system because they usually connect to a database that contains information such as identities with credit card numbers and passwords. Web application vulnerabilities increase the threat that hackers will exploit the operating system and web server or web application software. Web applications are essentially another door into a system and can be exploited to compromise the system.

3.15.3.Web Application Threats

Many web application threats exist on a web server. The following are the most common threats:

Cross-site scripting a parameter entered into a web form is processed by the web application. The correct combination of variables can result in arbitrary command execution.

SQL injection Inserting SQL commands into the URL gets the database server to dump, alter, delete, or create information in the database.

Command injection The hacker inserts programming commands into a web form.

Cookie poisoning and snooping The hacker corrupts or steals cookies.

Buffer overflow Huge amounts of data are sent to a web application through a web form to execute commands.

Authentication hijacking The hacker steals a session once a user has authenticated.

Directory traversal / Unicode The hacker browses through the folders on a system via a web browser or Windows explorer.

3.15.4.Web-Based Password Cracking Techniques

Web servers and web applications support multiple authentication types. The most common is HTTP authentication. There are two types of HTTP authentication: basic and digest. HTTP authentication sends the username and password in cleartext, whereas digest authentication hashes the credentials and uses a challenge-response model for authentication.

3.15.5.Password Cracker

A password cracker is a program designed to decrypt passwords or disable password protection. Password crackers rely on dictionary searches (attacks) or brute-force methods to crack passwords.

The first step in a dictionary attack is to generate a list of potential passwords that can be found in a dictionary. The hacker usually creates this list with a dictionary generator program or dictionaries that can be downloaded from the Internet. Next, the list of dictionary words is hashed or encrypted. This hash list is compared against the hashed password the hacker is trying to crack. The hacker can get the hashed password by sniffing it from a wired or wireless network or directly from the Security Accounts Manager (SAM) or shadow password files on the hard drive of a system. Finally, the program displays the unencrypted version of the password. Dictionary password crackers can only discover passwords that are dictionary words.

If the user has implemented a strong password, then brute-force password cracking can be implemented. Brute-force password crackers try every possible combination of letters, numbers, and special characters, which takes much longer than a dictionary attack because of the number of permutations.

3.16. **Cross Site Scripting**

Websites today are more complex than ever, containing a lot of dynamic content making the experience for the user more enjoyable. Dynamic content is achieved through the use of web applications which can deliver different output to a user depending on their settings and needs. Dynamic websites suffer from a threat that static websites don't, called "Cross Site Scripting" (or XSS dubbed by other security professionals). Currently small informational tidbits about Cross Site Scripting holes exist but none really explain them to an average person or administrator.

3.16.1.What is Cross Site Scripting?

Cross site scripting (also known as XSS) occurs when a web application gathers malicious data from a user. The data is usually gathered in the form of a hyperlink which contains malicious content within it. The user will most likely click on this link from another website, instant message, or simply just reading a web board or email message. Usually the attacker will encode the malicious portion of the link to the site in HEX (or other encoding methods) so the request is less suspicious looking to the user when clicked on. After the data is collected by the web application, it creates an output page for the user containing the malicious data that was originally sent to it, but in a manner to make it appear as valid content from the website. Many popular guestbook and forum programs allow users to submit posts with html and javascript embedded in them. If for example I was logged in as "john" and read a message by "joe" that contained malicious javascript in it, then it may be possible for "joe" to hijack my session just by reading his bulletin board post. Further details on how attacks like this are accomplished via "cookie theft" are explained in detail below.

3.16.2.XSS and CSS

Often people refer to Cross Site Scripting as CSS. There has been a lot of confusion with Cascading Style Sheets (CSS) and cross site scripting. Some security people refer to Cross Site Scripting as XSS. If you hear someone say (I found a XSS hole), they are talking about Cross Site Scripting for certain.

3.16.3.The Threats of Cross Site Scripting

Often attackers will inject JavaScript, VBScript, ActiveX, HTML, or Flash into a vulnerable application to fool a user (Read below for further details) in order to gather data from them. Everything from account hijacking, changing of user settings, cookie theft/poisoning, or false advertising is possible. New malicious uses are being found every day for XSS attacks. The post below by Brett Moore brings up a good point with regard to (Denial of Service) and potential (auto-attacking) of hosts if a user simply reads a post on a message board.

3.16.4.Vendor Protection

This is a simple answer. Never trust user input and always filter metacharacters. This will eliminate the majority of XSS attacks. Converting < and > to &lt; and &gt; is also suggested when it comes to script output. Remember XSS holes can be damaging and costly to organization business if abused. Often attackers will disclose these holes to the public, which can erode customer and public confidence in the security and privacy of organization organization's site. Filtering < and > alone will not solve all cross site scripting attacks and it is suggested you also attempt to filter out (and) by translating them to &#40; and &#41;, and also # and & by translating them to &#35 (#) and &#38 (&).

3.16.5.User Protection

The easiest way to protect ourselves as a user is to only follow links from the main website you wish to view. If you visit one website and it links to CNN for example, instead of clicking on it visit CNN's main site and use its search engine to find the content. This will probably eliminate ninety percent of the problem. Sometimes XSS can be executed automatically when you open an email, email attachment, read a guestbook, or bulletin board post. If you plan on opening an email, or reading a post on a public board from a person you don't know BE CAREFUL. One of the best ways to protect ourself is to turn off Javascript in your browser settings. In IE turn security settings to high. This can prevent cookie theft, and in general is a safer thing to do.

3.16.6.Common XSS Holes

Cross site scripting holes are gaining popularity among hackers as easy holes to find in large websites. Websites from FBI.gov, CNN.com, Time.com, Ebay, Yahoo, Apple computer, Microsoft, Zdnet, Wired, and Newsbytes have all had one form or another of XSS bugs. Every month roughly 10-25 XSS holes are found in commercial products and advisories are published explaining the threat [8].

3.17. SQL Injection and Buffer Overflows

SQL injection and buffer overflows are similar exploits in that they’re both usually delivered via a user input field. The input field is where a user may enter a username and password on a website, add data to a URL, or perform a search for a keyword in another application.

Both SQL server injection and buffer overflow vulnerabilities are caused by the same issue: invalid parameters. If programmers don’t take the time to validate the variables a user can enter into a variable field, the results can be serious and unpredictable. Sophisticated hackers can exploit this vulnerability, causing an execution fault and shutdown of the system or application, or a command shell to be executed for the hacker.

3.17.1.SQL Injection

During a SQL injection attack, malicious code is inserted into a web form field or the website’s code to make a system execute a command shell or other arbitrary commands. Just as a legitimate user enters queries and additions to the SQL database via a web form, the hacker can insert commands to the SQL server through the same web form field. For example, an arbitrary command from a hacker might open a command prompt or display a table from the database. A database table may contain secret information, or passwords. SQL servers are very common database servers and used by many organizations to store confidential data. This makes a SQL server a high value target and therefore a system that is very attractive to hackers.

3.17.2.Understand the Steps to Conduct SQL Injection

Before launching a SQL injection attack, the hacker determines whether the configuration of the database and related tables and variables is vulnerable. The steps to determine the SQL server’s vulnerability are as follows:

1. Using web browser, search for a website that uses a login page or other database input or query fields (such as an “I forgot my password” form). Look for web pages that display the POST or GET HTML commands by checking the site’s source code.

2. Test the SQL server using single quotes (‘‘). Doing so indicates whether the user input variable is sanitized or interpreted literally by the server. If the server responds with an error message that says use ‘a’=‘a’ (or something similar), then it’s most likely susceptible to a SQL injection attack.
3. Use the SELECTcommand to retrieve data from the database or the INSERT command to add information to the database.

3.17.3.SQL Server Vulnerabilities

Some examples of variable field text we can use on a web form to test for SQL vulnerabilities:

- Blah’ or 1=1--
- Login:blah’ or 1=1--
- Password::blah’ or 1=1--
- http://search/index.asp?id=blah’ or 1=1—

These commands and similar variations may allow the bypassing of a login depending on the structure of the database. When entered in a form field the commands may return many rows in a table or even an entire database table because the SQL server is interpreting the terms literally. The double dashes near the end of the command tell SQL to ignore the rest of the command as a comment. Here are some examples of how to use SQL commands to take control. To get a directory listing, type the following in a form field:

```
Blah’;exec master..xp_cmdshell “dir c:\*.* /s >c:\directory.txt”-
```

To create a file, type the following in a form field:

```
Blah’;exec master..xp_cmdshell “echo hacker-was-here > c:\hacker.txt”-
```

To ping an IP address, type the following in a form field:

Blah’;exec master..xp_cmdshell “ping 192.168.1.1”- [9]

3.18. **Buffer Overflows**

Buffer overflows are exploits that hackers use against an operating system or application; like SQL injection attacks, they're usually targeted at user input fields. A buffer overflow exploit causes a system to fail by overloading memory or executing a command shell or arbitrary code on the target system. A buffer-overflow vulnerability is caused by a lack of bounds checking or a lack of input-validation sanitization in a variable field (such as on a web form). If the application doesn't check or validate the size or format of a variable before sending it to be stored in memory an overflow vulnerability exits.

The two types of buffer overflows are stack-based and heap-based. The stack and the heap are storage locations for user-supplied variables within a running program. Variables are stored in the stack or heap until the program needs them. Stacks are static locations of memory address space, whereas heaps are dynamic memory address spaces that occur while a program is running. A heap-based buffer overflow occurs in the lower part of the memory and over-writes other dynamic variables. As a consequence, a program can open a shell or command prompt or stop the execution of a program.

To detect program buffer overflow vulnerabilities that result from poorly written source code, a hacker sends large amounts of data to the application via a form field and sees what the program does as a result.

3.18.1.Stack-Based Buffer Overflows

The following are the steps a hacker uses to execute a stack-based buffer overflow:

1 Enter a variable into the buffer to exhaust the amount of memory in the stack.

2 Enter more data than the buffer has allocated in memory for that variable, which causes the memory to overflow or run into the memory space for the next process. Then, add another

variable, and overwrite the return pointer that tells the program where to return to after executing the variable.

3 A program executes this malicious code variable and then uses the return pointer to get back to the next line of executable code. If the hacker successfully overwrites the pointer, then the program executes the hacker's code instead of the program code.

Most hackers don't need to be this familiar with the details of buffer overflows. Prewritten exploits can be found on the Internet and are exchanged between hacker groups.

3.18.2.Buffer Overflow Mutation Techniques

As you see, hackers can graduate from standard buffer overflows to redirecting the return pointer to the code of their choosing. A hacker must know the exact memory address and the size of the stack in order to make the return pointer execute their code. A hacker can use a No Operation (NOP) instruction, which is just padding to move the instruction pointer and does not execute any code. The NOP is added to a string before the malicious code to be executed.

If an intrusion detection system (IDS) is present on the network, it can thwart a hacker who sends a series of NOPs to forward the instruction pointer. To bypass the IDS, the hacker can randomly replace some of the NOPs with equivalent pieces of code, such as x++,x-;?NOPNOP. This example of a mutated buffer overflow attack can bypass detection by IDS [2].

CHPTER FOUR

ETHICAL HACKING and PENTERATION TEST METHODOLOGIES and TOOLS

4. ETHICAL HACKING and PENTERATION TEST METHODOLOGIES and TOOLS

4.1. Introduction

In this chapter, we discuss the ethical hacking and penetration test methodologies then we present the utilities and tools we will use to do our test, finally we take a little description about SUST network.

We follow the five stages for ethical hacking and penetration test (discussed earlier). And we use appropriate tools for each stage.

4.2. Methodologies

There are many different penetration testing standards and methodologies used in the world such as:

- EC-Council's Certified Ethical Hacker (**CEH)**
- The United States National Security Agency (NSA) INFOSEC Assessment Methodology (IAM) **NSA IAM,**
- The Government Communications Head quarters(GCHQ) in the United Kingdom, Communications and Electronics Security Group **CESG CHECK (UK),**
- **ISECOM**'s Open Source Security Testing Methodology Manual (OSSTMM)(http://www.osstmm.org)
- **Master Card SDP,**
- **CREST**,
- Council of Registered Ethical Security Testers (**CREST**) (http://www.crestapproved.com)
- **TIGER** Scheme (http://www.tigerscheme.org)

- Open Source Web Application Security Project (**OWASP**) (http://www.owasp.org) [1]

4.2.1. EC-Council's Certified Ethical Hacker (CEH)

The Certified Ethical Hacker (CEH) is a professional certification provided by the International Council of E-Commerce Consultants (EC-Council.)

An Ethical Hacker is one name given to a Penetration Tester. An ethical hacker is usually employed by an organization that trusts him to attempt to penetrate networks and/or computer systems, using the same methods as a hacker, for the purpose of finding and fixing computer security vulnerabilities. Illegal hacking (i.e.; gaining unauthorized access to computer systems) is a crime in most countries, but penetration testing done by request of the owner of the targeted system or network is not [2].

4.2.2. NSA IAM

The United States National Security Agency (NSA) has provided an INFOSEC Assessment Methodology (IAM) frame work to help consultants and security professionals outside the NSA provide assessment services to clients in line with a recognized standard. The NSA IAM home page is http://www.iatrp.com.

In this research we concentrate in IAM frame work levels

The IAM frame work defines three levels of assessment:

Level1 (Assessment)

Level 1 involves discovering a cooperative high-level over view of the organization being assessed, including access to policies, procedures, and information flow. No hands-on network or system testing is undertaken at this level.

Level2 (Evaluation)

Level 2 is a hands-on cooperative process that involves testing with network scanning, penetration tools, and the use of specific technical expertise.

Level3 (Red Team)

Level 3 non-cooperative and external to the target network, involving penetration testing to simulate the appropriate adversary. IAM assessment is non-intrusive, so within this frame work, aLevel3assessment involves full qualification of vulnerabilities [1].

4.2.3. CESG CHECK

The Government Communications Head quarters(GCHQ) in the United Kingdom Has an information assurance arm known as the Communications and Electronics Security Group (CESG). In the same way that the NSAI AM frame work allows security consultants outside the NSA to provide assessment services, CESG operates a program known as CHECK to evaluate and accredit security testing teams with in the U.K. to undertake government assessment work. The CESG CHECK home page is accessible at http://www.cesg.gov.uk/site/check/index.cfm.

Unlike the NSA IAM, which covers many aspects of information security (including review of security policy, antivirus, backups, and disaster recovery), CHECK squarely tackles the area of network security assessment. Asecond program is the CESG Listed Adviser Scheme (CLAS), which covers information security in a broader sense and tackles areas such as ISO/IEC 27002, security policy creation, and auditing [1].

Many of these methodologies have same techniques in some, in Sudan and middle east we one popular method CEH we chose this standard as our penetration test methodology because its available and common in Sudan also it cover and high light many of the others methodologies (Network Security Assessment2).

4.3. The Certified Ethical Hacker (CEH)

An ethical hacker based on CEH methodology follows processes similar to those of a malicious hacker. The steps to gain and maintain entry into a computer system are similar no matter what the hacker's intentions are. The Certified Ethical Hacker (CEH) ethical hacking methodology consists of five stages as follow [2]:

Stage one: Passive and Active Reconnaissance.
Stage two: Scanning.
Stage three: Gaining Access
Stage four: Maintaining Access
Stage five: Covering Tracks

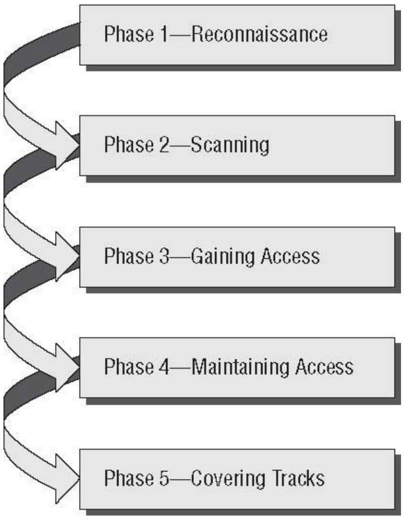


Figure 4-1 The CEH **five stages**

4.3.1. Stage One Passive and Active Reconnaissance

Passive reconnaissance involves gathering information regarding a potential target without the targeted individual's or organization's knowledge. Passive reconnaissance can be as simple as watching a building to identify what time employees enter the building and when they leave. However, it's usually done using Internet searches or by Googling an individual or company to gain information. This process is generally called information gathering. Social engineering and dumpster diving are also considered passive information-gathering methods. Sniffing the network is another means of passive reconnaissance and can yield useful information such as IP address ranges, naming conventions, hidden servers or networks, and other available services on the system or network. Sniffing network traffic is similar to building monitoring: A hacker watches the flow of data to see what time certain transactions take place and where the traffic is going [2].

Active reconnaissance involves probing the network to discover individual hosts, IP addresses, and services on the network. This usually involves more risk of detection than passive reconnaissance. Active reconnaissance can give a hacker an indication of security measures in place, but the process also increases the chance of being caught or at least raising suspicion.

Both passive and active reconnaissance can lead to the discovery of useful information to use in an attack. For example, it's usually easy to find the type of web server and the operating system (OS) version number that a company is using. This information may enable a hacker to find vulnerability in that OS version and exploit the vulnerability to gain more access.

4.3.2. Stage Two Scanning

Scanning involves taking the information discovered during reconnaissance and using it to examine the network. Tools that a hacker may employ during the scanning phase can include dialers, port scanners, network mappers, sweepers, and vulnerability scanners. Hackers are seeking any information that can help them perpetrate attack such as computer names, IP addresses, and user accounts.

4.3.3. Stage Three Gaining Access

This is the phase where the real hacking takes place. Vulnerabilities discovered during the reconnaissance and scanning phase are now exploited to gain access. The method of connection the hacker uses for an exploit can be a local area network (LAN, either wired or wireless), local access to a PC, the Internet, or offline. Examples include stack-based buffer overflows, denial of service (DoS), and session hijacking. These topics will be discussed in later chapters. Gaining access is known in the hacker world as owning the system. We do not operate this stage in our research.

4.3.4. Stage Four Maintaining Access

Once a hacker has gained access, they want to keep that access for future exploitation and attacks. Sometimes, hackers harden the system from other hackers or security personnel by securing their exclusive access with backdoors, rootkits, and Trojans. Once the hacker owns the system, they can use it as a base to launch additional attacks. In this case, the owned system is sometimes referred to as a zombie system. We do not operate this stage in our research.

4.3.5. Stage Five Covering Tracks

Once hackers have been able to gain and maintain access, they cover their tracks to avoid detection by security personnel, to continue to use the owned system, to remove evidence of hacking, or to avoid legal action. Hackers try to remove all traces of the attack, such as log files or intrusion detection system (IDS) alarms [2].

4.4. The Scope of Work (Penetration Testing Scope)

If we do these five phases it will lead us to Extensive information to hardening SUST network security by suggesting the suitable countermeasures [2], but we will just perform the first two stages, (Reconnaissance and Scanning) because of the magnitude of work, and effort and the amount of information we assume we will gain (because of the amount of the information we find in the mentioned scope are huge therefore any extra work will creep the scope), we assure that amount will be over of the research scope and it will creep it. A typical hacker spends 90 percent of the time profiling and gathering information on a target(i.e. Reconnaissance and Scanning) and 10 percent of the time launching the attack (i.e. rest stages) [2], also there is a high impact resulting from performing other stages to availability of SUST network services (see 1.4 Risks), and finally the legal issues of doing the penetration test, these activities perform by legitimate contract, and it is better to do these stages by administrators themselves (see 2.14 Understanding the Legal Implications of Hacking).

We perform a remote network ethical hacking type (see 2.11 Types of Ethical Hacks), with black box testing (see 2.12 Testing Types), this mean we do the ethical hacking test over the Internet i.e. outside SUST defenses and we assume we know nothings about SUST network except its URL (www.sustech.edu). We try to break or find vulnerability in the outside defenses of the network, such as firewall and webserver vulnerabilities.

4.5. Tools and Utilities

We perform stage one and stage two (gather information) using the following suitable tools. Here we will describe these tools for these two stages.

4.5.1. For Reconnaissance and Footprinting We Use the Following Tools:

4.5.2. Nslookup

One of the powerful tools queries DNS servers for record information. It’s included in UNIX, Linux, and Windows operating systems. Nslookup is a network administration command-line tool available in many computer operating systems for querying the Domain Name System (DNS) to obtain domain name or IP address mapping or for any other specific DNS record [2].

4.5.3. WHOIS

WHOIS (pronounced "who is") is an Internet database that contains information on domain names including the name servers associated with the domain name, the domain registrar and the Administrative, Billing and Technical contacts with postal and email addresses. (http://en.wikipedia.org/wiki/Whois)

The WHOIS is also a tool or an application which searches the domain name information contained in WHOIS databases. It is generally used to check either the availability of a domain name or the ownership of a domain name. The tool requires you to enter a domain name such as sustech.edu (without the www prefix). If the domain is available you will be informed of the same, else, you would be displayed one or more details:

1. The registrant information. Details of the person who registered the domain name including their postal and email addresses and phone number.
2. The contacts: Each domain name is associated with three contacts - Administrative, Billing and Technical. In most cases, all the three would belong to the same person (the registrant).
3. The creation and expiration date of the domain name.
4. The name servers associated with the domain name [10].

4.5.4. Traceroute

Traceroute is a utility that sends a sequence of Internet Control Message Protocol (ICMP) packets addressed to a destination host. Tracing the intermediate routers traversed involves control of the time-to-live (TTL) Internet Protocol parameter. Traceroute is often used for network troubleshooting. By showing a list of routers traversed, it allows the user to identify the path taken to reach a particular destination on the network. This can help identify routing problems or firewalls that may be blocking ICMP traffic. Traceroute is also used by penetration testers to gather information about network infrastructure and IP ranges around a given host [2].

4.5.5. VisualRoute

Because routers are generally named according to their physical location, the Traceroute results help us locate these devices.

VisualRoute is a packet-tracking tool with a GUI or visual interface. They plot the path the packets travel on a map and can visually identify the locations of routers and other internetworking devices. These tools operate similarly to Traceroute and perform the same information gathering; however, they provide a visual representation of the results [11].

4.6. Tools we used for Scanning and Enumeration

4.6.1. Acunetix

Acunetix Web Vulnerability Scanner (WVS) broadens the scope of vulnerability scanning by introducing highly advanced heuristic and rigorous technologies designed to tackle the complexities of today's web-based environment. Acunetix WVS automatically checks web applications for vulnerabilities such as SQL Injections, cross site scripting, arbitrary file creation/deletion, and weak password strength on authentication pages. AcuSensor technology detects vulnerabilities which typical black box scanners miss. Acunetix WVS boasts a comfortable GUI, an ability to create professional security audit and compliance reports, and tools for advanced manual webapp testing [12].

4.7. SUST Network:

SUST Network is one of the biggest networks in Sudan. It is a very useful and complicated network, because of the variety of users and services.

SUST network consists of core devices as illustrated in figure (4-2), two main Routers (Huawei quid way AR 4680), two firewalls (Huawei Eudemon

300), two core switches (Huawei S8505) and SUST Zones (more than 14 zones).

SUST network has many services like internet service, mailing service, domain service, hosting service, file transferring service, web service, and many other local systems implemented on the network like accounting system, students results, registration system, hiring and human resource system. Figure (4-3) shows SUST DMZ (more than eight servers: webserver, mail server, DNS server, FTP server, and SUST local servers like SMS server, model server, Registration server and Students Result server).

The OS used in SUST servers is Linux Redhat5 and Centos; the database is Oracle 10G used in Registration server and Students Results server and use Apache webserver.

SUST network extends in 14 different Campuses; the figure (4.4) shows SUST WAN connected by wireless WiMax Motorola and fiber Optics, the hub center is the Data Center located in Laser Building. The zones include:

[1] The Data center (laser building),
[2] college of Business studies campus,
[3] library affairs deanship campus,
[4] West campus (headquarter),
[5] College of Education campus,
[6] South campus (College of Engineering),
[7] North campus,
[8] Shambat campus (College of agricultural studies),
[9] East of Kadaru or Wad Elmagbool campus (College of Water and environmental engineering),
[10] Hilat Kuku campus (College of Veterinary medicine and College of Animal production science and technology),
[11] College of Music and drama campus,
[12] Soba campus (College of Forestry and range science),

[13] College of Medical radiological sciences campus

[14] and College of Technology campus,

SUST network is connected to internet by two fiber optics leased lines from the major ISPs in Sudan “Sudatel and Canar” which are illustrated in Figure (4-5). There are many proxies serving distribution and internet caching.

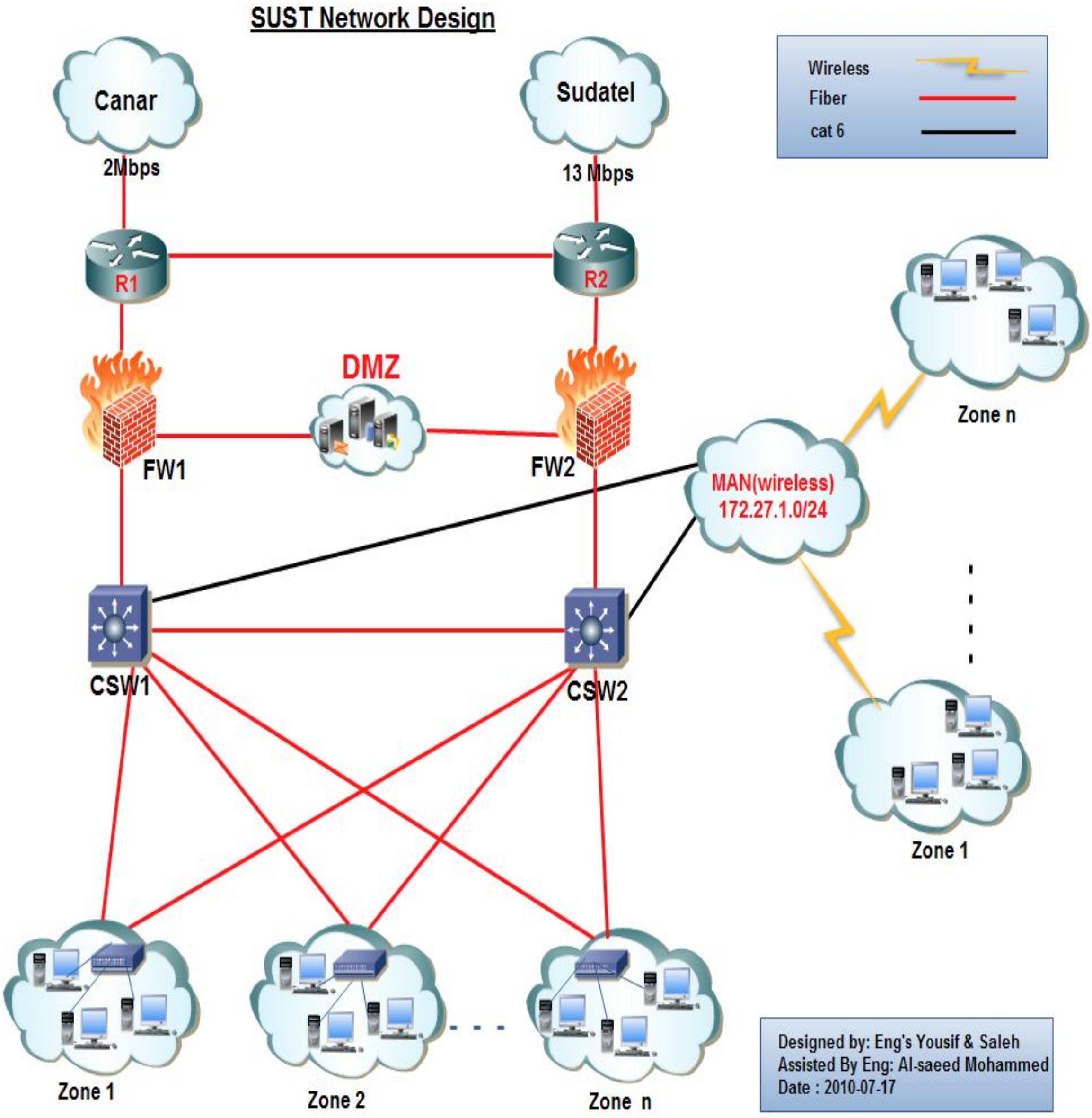


Figure 4-2 Layout of SUST Network

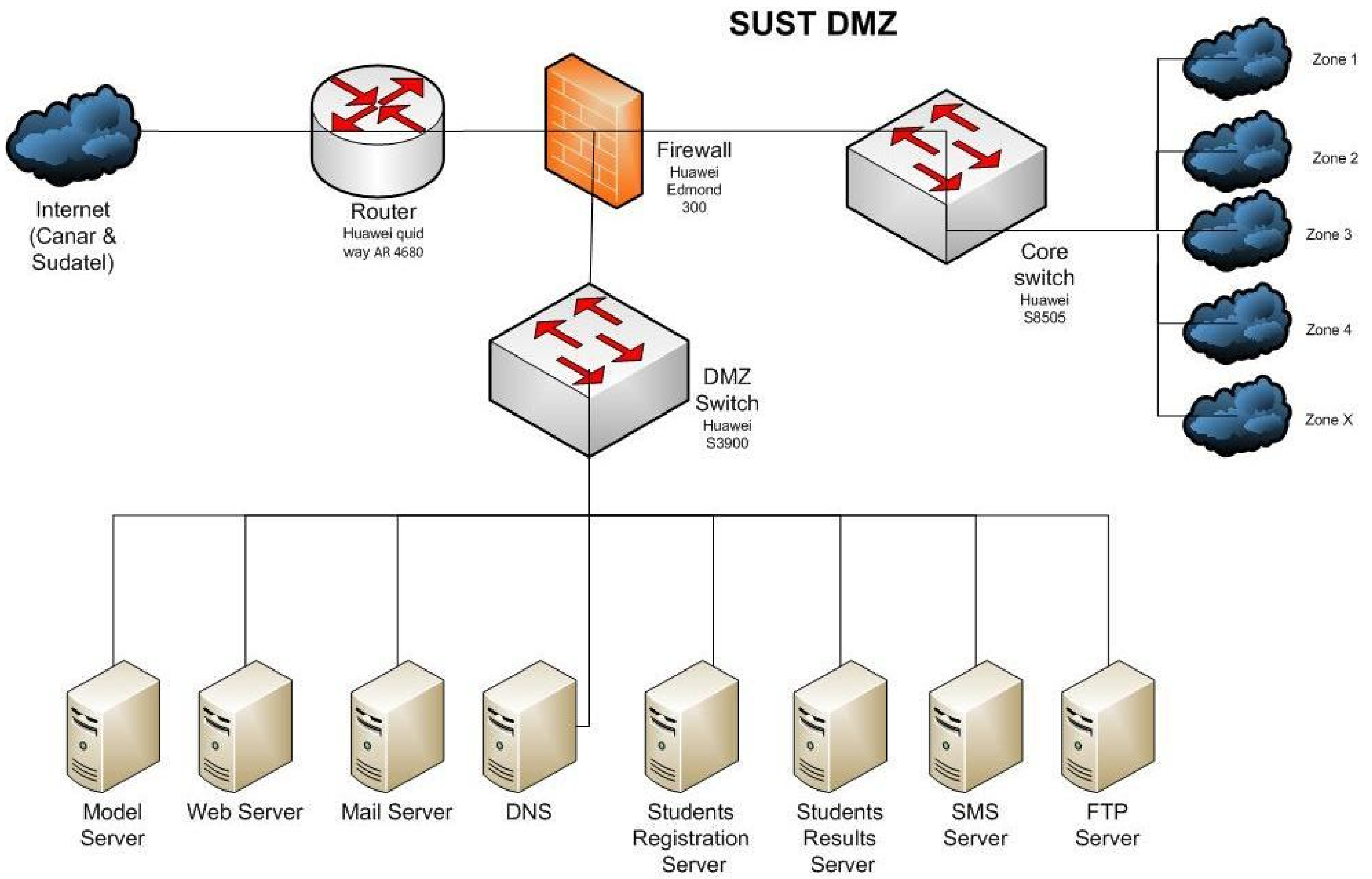


Figure 4-3 SUST DMZ

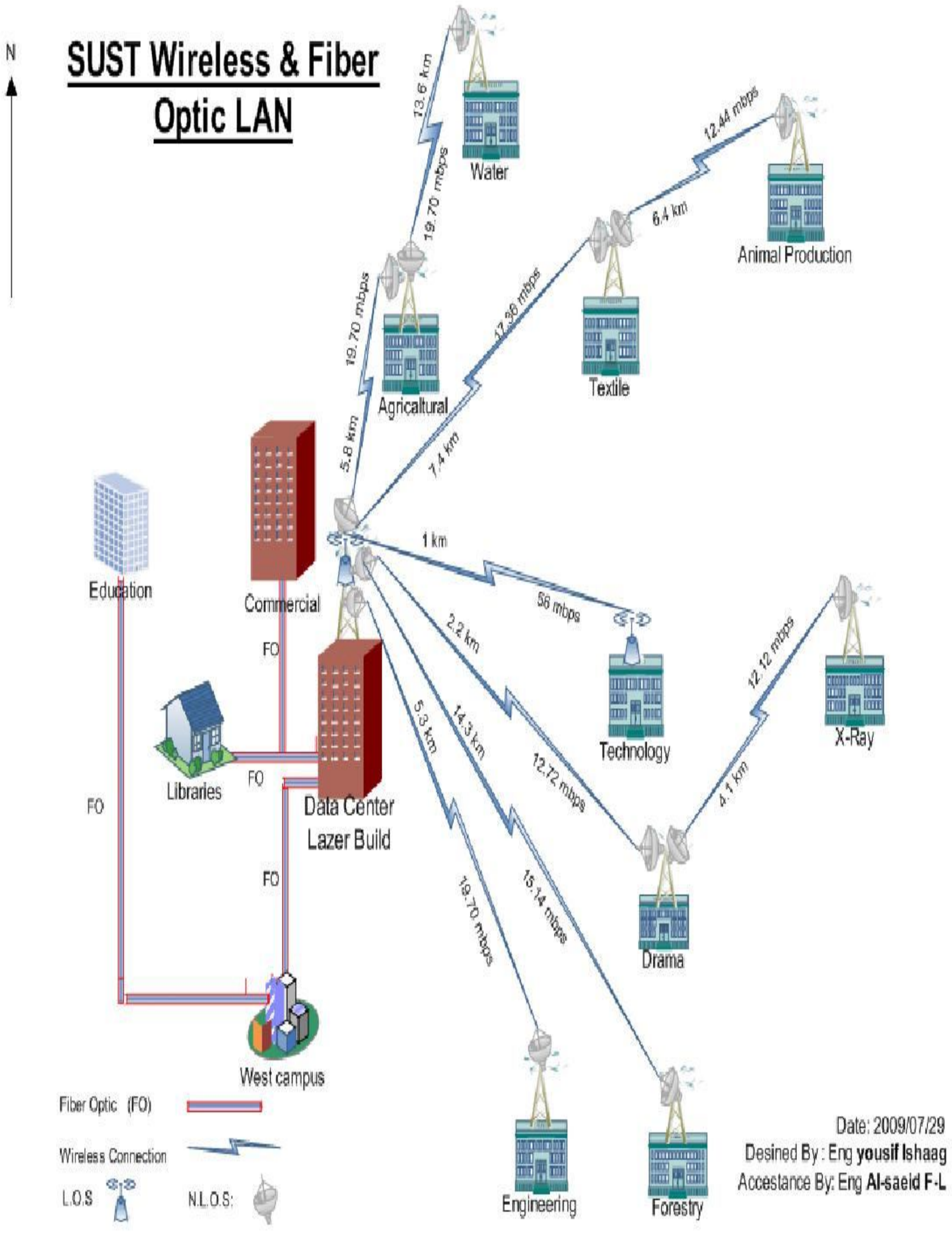


Figure 4-4 SUST WAN

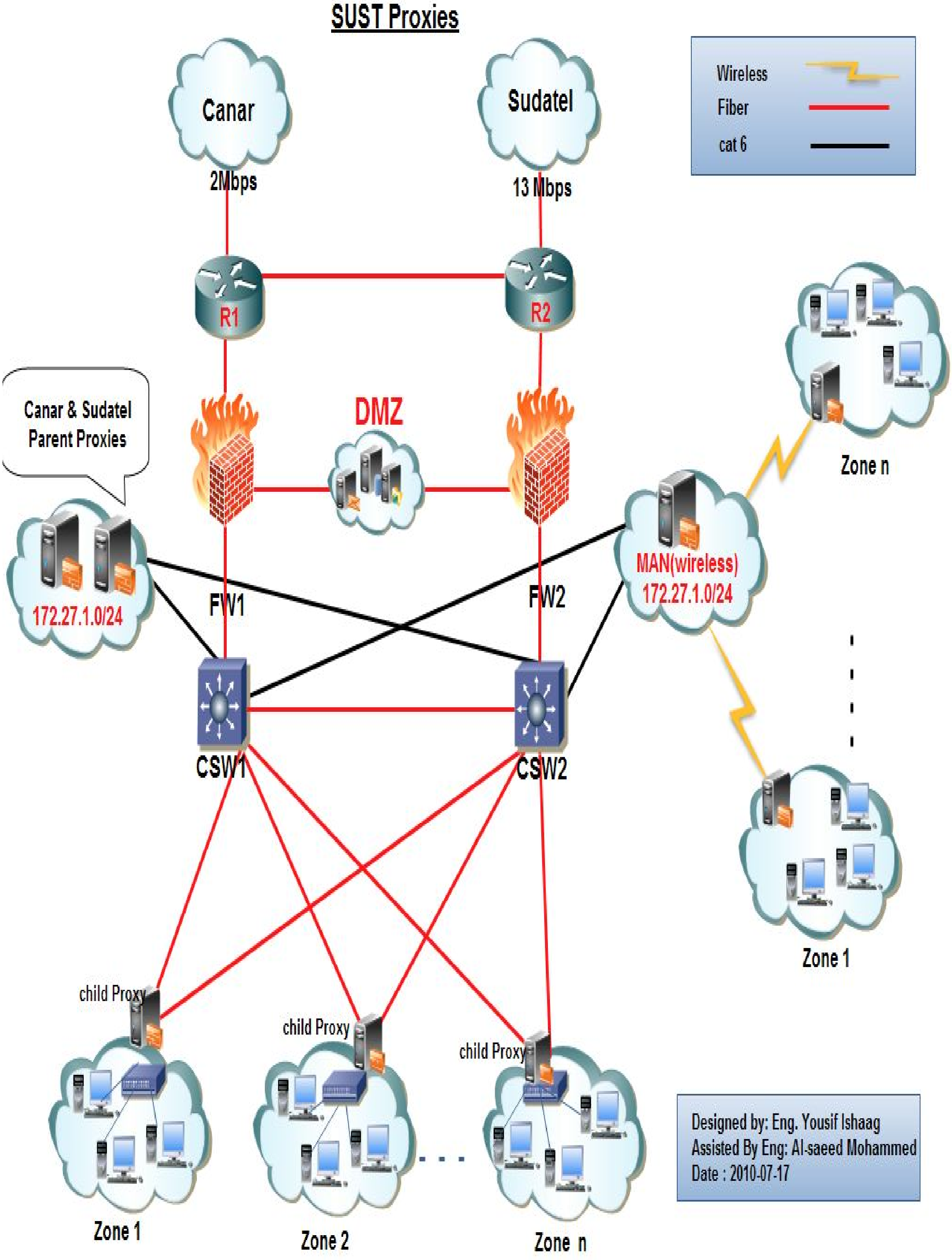


Figure 4-5 SUST Proxies

CHAPTER FIVE

RESULTS and DISCUSSIONS

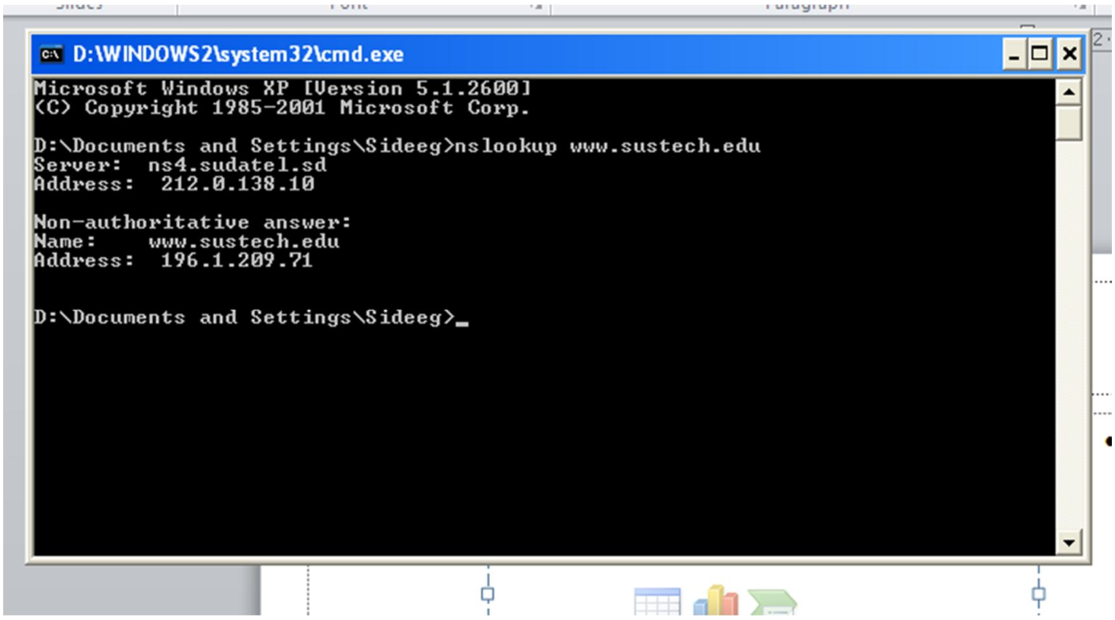
5. RESULTS and DISCUSSIONS

5.1. Introduction

This chapter discusses the Results of ethical hacking and penetration testing of SUST network. The tools mentioned in chapter three are used for investigation.

5.2. Nslookup Results

The result of Nslookup command in windows XP is illustrated in figure (5-1) below [13].



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

D:\Documents and Settings\Sideeg>nslookup www.sustech.edu
Server:  ns4.sudatel.sd
Address:  212.0.138.10

Non-authoritative answer:
Name:    www.sustech.edu
Address:  196.1.209.71


D:\Documents and Settings\Sideeg>_
```

Figure 5-1 Nslookup

The details of the figure are as follows.

Microsoft Windows XP [Version 5.1.2600]

(C) Copyright 1985-2001 Microsoft Corp.

D:\Documents and Settings\Sideeg>Nslookup www.sustech.edu

Server: ns4.sudatel.sd

Address: 212.0.138.10

Non-authoritative answer:

Name: www.sustech.edu

Address: 196.1.209.71

This investigation reveals all the alias records for http://www.sustech.edu and the IP address of the web server. We can even discover all the name servers and associated IP addresses.

The Nslookup result for sustech.edu from

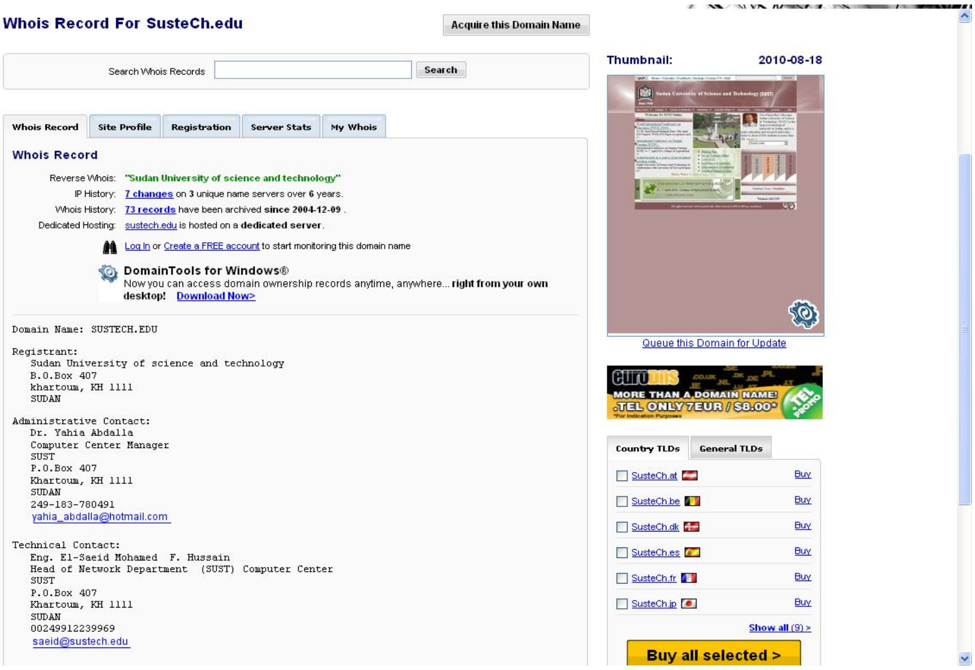
http://www.kloth.net/services/nslookup.php entering [susteh.edu] with querytype=[ANY:] [14] is:

```
 DNS server handling your query: localhost
 DNS server's address:  127.0.0.1#53
  Non-authoritative answer:
 sustech.edu     text = "v=spf1 ip4:196.29.170.205 ipv4:196.1.209.75
a:mail.sustech.edu a:mail.suin.edu.sd mx:suin.edu.sd mx:mail.suin.edu.sd
~all"
 sustech.edu
           origin = ns1.sustech.edu
           mail addr = netadmin.sustech.edu
           serial = 2010070705
           refresh = 1
           retry = 15
           expire = 604800
           minimum = 86400
 sustech.edu     mail exchanger = 10 mail.sustech.edu.
 Name: sustech.edu
 Address: 196.1.209.71
 sustech.edu     nameserver = ns1.sustech.edu.
  Authoritative answers can be found from:
```

Now we knew that the SUST web server has had the IP address of 196.1.209.71.

5.3. WHOIS Results

The WHOIS utility from the site http://whois.domaintools.com/ as shown in figure (5-2) is used to get the important and critical results without a big effort [15].



Whois Record For SusteCh.edu

Search Whois Records

Whois Record

Reverse Whois: "Sudan University of science and technology"

IP History: 7 changes on 3 unique name servers over 6 years.

Whois History: 73 records have been archived since 2004-12-09 .

Dedicated Hosting: sustech.edu is hosted on a dedicated server.

Log In or Create a FREE account to start monitoring this domain name

DomainTools for Windows®
Now you can access domain ownership records anytime, anywhere... right from your own desktop! Download Now>

```
Domain Name: SUSTECH.EDU

Registrant:
   Sudan University of science and technology
   B.O.Box 407
   khartoum, KH 1111
   SUDAN

Administrative Contact:
   Dr. Yahia Abdalla
   Computer Center Manager
   SUST
   P.O.Box 407
   Khartoum, KH 1111
   SUDAN
   249-183-780491
   yahia_abdalla@hotmail.com

Technical Contact:
   Eng. El-Saeid Mohamed  F. Hussain
   Head of Network Department  (SUST) Computer Center
   SUST
   P.O.Box 407
   Khartoum, KH 1111
   SUDAN
   00249912239969
   saeid@sustech.edu
```

Thumbnail: 2010-08-18

□	SusteCh.at	Buy
□	SusteCh.be	Buy
□	SusteCh.dk	Buy
□	SusteCh.es	Buy
□	SusteCh.fr	Buy
□	SusteCh.jp	Buy

Figure 5-2 Whois output for http://www.sustech.edu from

The details of the figure are as follows.

Domain Name: SUSTECH.EDU

Registrant:

Sudan University of science and technology

B.O.Box 407

khartoum, KH 1111

SUDAN

Administrative Contact:

Dr. Yahia Abdalla

Computer Center Manager

SUST

P.O.Box 407

Khartoum, KH 1111

SUDAN

249-183-780491

yahia_abdalla@hotmail.com

Technical Contact:

Eng. El-Saeid Mohamed F. Hussain
Head of Network Department (SUST) Computer Center
SUST
P.O.Box 407
Khartoum, KH 1111
SUDAN
00249912239969

saeid@sustech.edu

Name Servers:

NS1.SUSTECH.EDU	196.1.209.76
NS2.SUSTECH.EDU	196.29.170.206
NS1.SUST.EDU.SD	
NS2.SUST.EDU.SD	

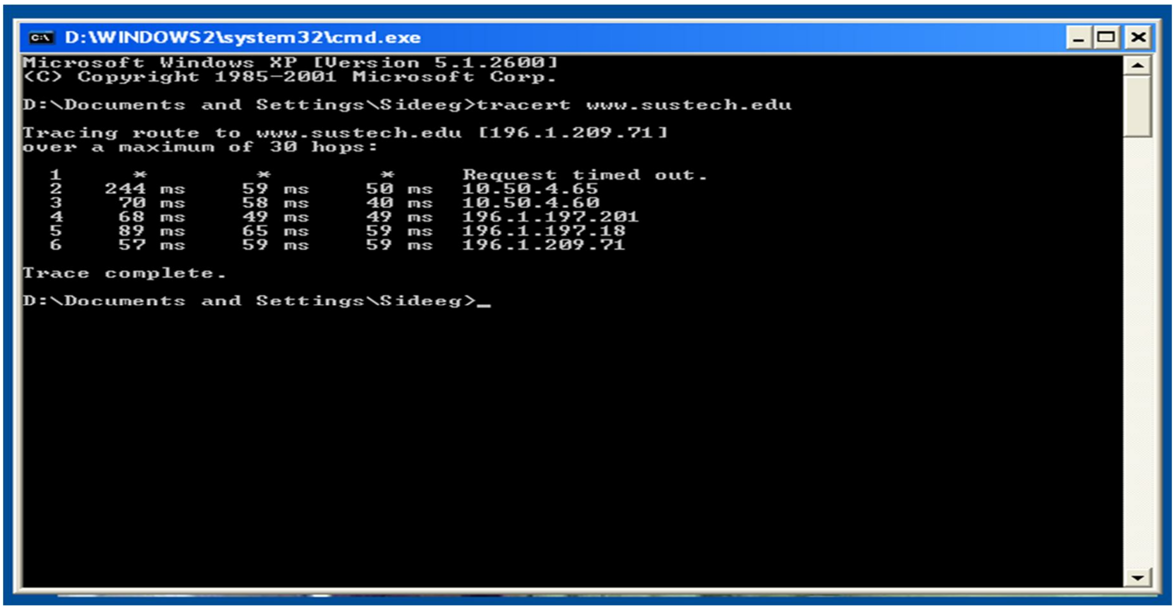
Domain record activated: 17-May-2000

Domain record last updated: 26-May-2010

Domain expires: 31-Jul-2011

5.4. Traceroute Results

The result of "tracert" command in windows XP is illustrated in figure (5-3) below [3].



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

D:\Documents and Settings\Sideeg>tracert www.sustech.edu

Tracing route to www.sustech.edu [196.1.209.71]
over a maximum of 30 hops:

  1     *        *        *     Request timed out.
  2   244 ms    59 ms    50 ms  10.50.4.65
  3    70 ms    58 ms    40 ms  10.50.4.60
  4    68 ms    49 ms    49 ms  196.1.197.201
  5    89 ms    65 ms    59 ms  196.1.197.18
  6    57 ms    59 ms    59 ms  196.1.209.71

Trace complete.

D:\Documents and Settings\Sideeg>_
```

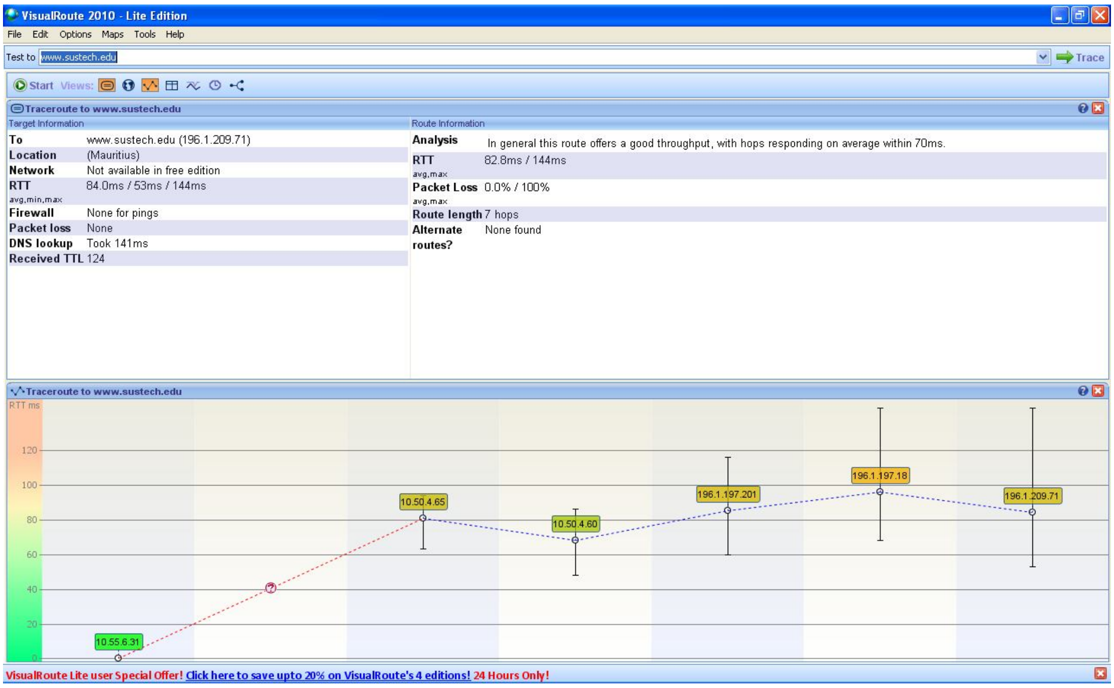
Figure 5-3 Traceroute output for www.sustech.edu

The details of the figure are as follows.

```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

D:\Documents and Settings\Sideeg>tracert www.sustech.edu
Tracing route to www.sustech.edu [196.1.209.71]
over a maximum of 30 hops:

 1      *         *         *      Request timed out.
 2  244 ms    59 ms    50 ms  10.50.4.65
 3    70 ms   58 ms    40 ms  10.50.4.60
 4    68 ms   49 ms    49 ms  196.1.197.201
 5    89 ms   65 ms    59 ms  196.1.197.18
 6    57 ms   59 ms    59 ms  196.1.209.71

Trace complete.

D:\Documents and Settings\Sideeg>
```

Notice in Figure (5-3) that the message first encounters the outbound ISP to reach the SUST web server, and that the server's IP address is revealed as 196.1.209.71. Knowing this IP address enables the ethical hacker to perform additional scanning on that host during the scanning phase of the attack.

The "VisualRoute" utility resulted in figure (5-4).



Test to www.sustech.edu

Traceroute to www.sustech.edu

Target Information

To	www.sustech.edu (196.1.209.71)
Location	(Mauritius)
Network	Not available in free edition
RTT avg,min,max	84.0ms / 53ms / 144ms
Firewall	None for pings
Packet loss	None
DNS lookup	Took 141ms
Received TTL	124

Route Information

Analysis	In general this route offers a good throughput, with hops responding on average within 70ms.
RTT avg,max	82.8ms / 144ms
Packet Loss avg,max	0.0% / 100%
Route length	7 hops
Alternate routes?	None found

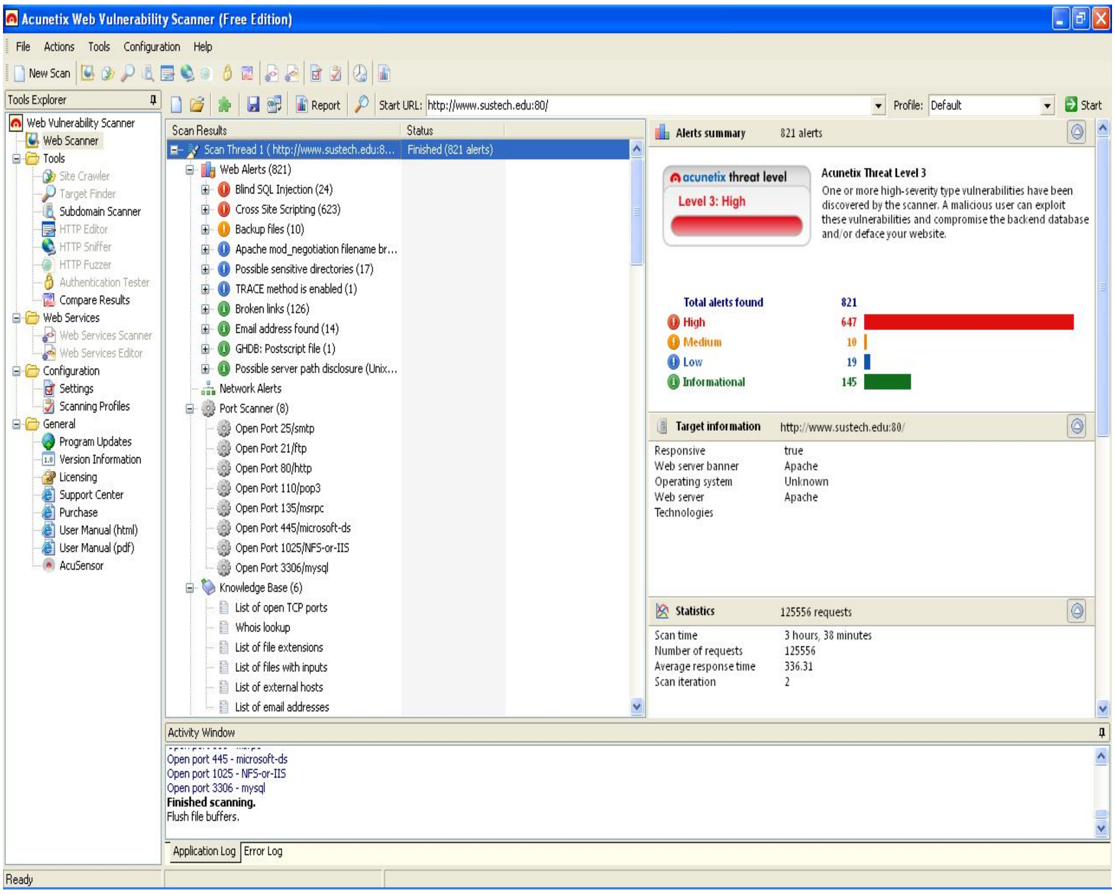
Traceroute to www.sustech.edu

VisualRoute Lite user Special Offer! Click here to save upto 20% on VisualRoute's 4 editions! 24 Hours Only!

Figure 5-4 Traceroute output by VisualRoute for www.sustech.edu

5.5. Acunetix Results

The result of Acunetix utility is illustrated in figure (5-5) below.



Start URL: http://www.sustech.edu:80/

Scan Results	Status
Scan Thread 1 (http://www.sustech.edu:8...	Finished (821 alerts)

```
Web Alerts (821)
    Blind SQL Injection (24)
    Cross Site Scripting (623)
    Backup files (10)
    Apache mod_negotiation filename br...
    Possible sensitive directories (17)
    TRACE method is enabled (1)
    Broken links (126)
    Email address found (14)
    GHDB: Postscript file (1)
    Possible server path disclosure (Unix...
Network Alerts
Port Scanner (8)
    Open Port 25/smtp
    Open Port 21/ftp
    Open Port 80/http
    Open Port 110/pop3
    Open Port 135/msrpc
    Open Port 445/microsoft-ds
    Open Port 1025/NFS-or-IIS
    Open Port 3306/mysql
Knowledge Base (6)
    List of open TCP ports
    Whois lookup
    List of file extensions
    List of files with inputs
    List of external hosts
    List of email addresses
```

Alerts summary 821 alerts

acunetix threat level
Level 3: High

Acunetix Threat Level 3
One or more high-severity type vulnerabilities have been discovered by the scanner. A malicious user can exploit these vulnerabilities and compromise the backend database and/or deface your website.

Total alerts found	821
High	647
Medium	10
Low	19
Informational	145

Target information	http://www.sustech.edu:80/
Responsive	true
Web server banner	Apache
Operating system	Unknown
Web server	Apache
Technologies	

Statistics	125556 requests
Scan time	3 hours, 38 minutes
Number of requests	125556
Average response time	336.31
Scan iteration	2

Activity Window

```
Open port 445 - microsoft-ds
Open port 1025 - NFS-or-IIS
Open port 3306 - mysql
Finished scanning.
Flush file buffers.
```

Figure 5-5 Acunetix output for www.sustech.edu

The search started at 09/11/2010 12:23:44 pm and finished at 09/11/2010 04:02:36 pm. The Scan duration is 3 hours, 38 minutes. Acunetix gives us four types of rich reports (detailed scan report of 309 pages, developer scan report of 361 pages, executive summary of 2 pages and quick report of 79 pages).

The executive summary report is discussed here because the other reports are too large and more detailed. Samples of the other reports are included in (Appendix B. Acunetix Other reports Samples).The executive report is shown in table (5-1).

Table 5-1 Acunetix Executive Report

Scan of http://www.sustech.edu:80/

Scan details

Scan information	
09/11/2010 12:23:44 م	Starttime
09/11/2010 04:02:36 م	Finish time
3 hours, 38 minutes	Scan time
Default	Profile

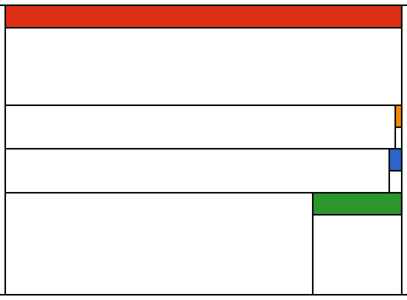
Server information	
True	Responsive
Apache	Server banner
Unknown	Server OS
	Server technologies

Threat level

Acunetix Threat Level 3

One or more high-severity type vulnerabilities have been discovered by the scanner. A malicious user can exploit these vulnerabilities and compromise the backend database and/or deface your website.

Alerts distribution



821	**Total alerts**
647	**High**
10	**Medium**
19	**Low**
145	**Information**

Executive summary

Alert count	Severity	Alert group
24	High	Blind SQL Injection
623	High	Cross Site Scripting
10	Medium	Backup files
1	Low	Apache mod_negotiation filename

17	Low	Possible sensitive directories
1	Low	TRACE method is enabled
126	Informational	Broken links
14	Informational	Email address found
1	Informational	GHDB
4	Informational	Possible server path disclosure (Unix)

This information is obtained from the shortest report:

- There are 821 issues 647 of them are high risk vulnerabilities.
- Apache server is used while the operating system is anonymous.
- The possible vulnerabilities include SQL Injection, Cross Site Scripting and Backup files described below.

5.5.1. Blind SQL Injection

The number of findings is 24 and the severity is high..

5.5.1.1. Description

This script is possibly vulnerable to SQL Injection attacks. SQL injection is a vulnerability that allows an attacker to alter backend SQL statements by manipulating the user input. An SQL injection occurs when web applications accept user input that is directly placed into a SQL statement and doesn't properly filter out dangerous characters.

This is one of the most common application layer attacks currently being used on the Internet. Despite the fact that it is relatively easy to protect against, there is a large number of web applications vulnerable.

5.5.1.2. Impact

An attacker may execute arbitrary SQL statements on the vulnerable system. This may compromise the integrity of SUST database and/or expose sensitive information.

Depending on the back-end database in use, SQL injection vulnerabilities lead to varying levels of data/system access for the attacker. It may be possible to not only manipulate existing queries, but to UNION in arbitrary data, use subselects, or append additional queries. In some cases, it may be possible to read in or write out to files, or to execute shell commands on the underlying operating system.

Certain SQL Servers such as Microsoft SQL Server contain stored and extended procedures (database server functions). If an attacker can obtain access to these procedures it may be possible to compromise the entire machine.

5.5.1.3. SQL Injection Mitigations

Web application developers often simply do not think about "surprise inputs", but security people do (including the hackers), so there are many approaches that can be applied here.

5.5.1.3.1. Sanitize the input

The script should filter metacharacters from user input. It's absolutely vital to sanitize user inputs to insure that they do not contain dangerous codes, whether to the SQL server or to HTML itself. One's first idea is to strip out "bad stuff", such as quotes or semicolons or escapes, but this is a misguided attempt. Though it's easy to point out some dangerous characters, it's harder to point to all of them.

The language of the web is full of special characters and strange markup (including alternate ways of representing the same characters), and efforts to authoritatively identify all "bad stuff" are unlikely to be successful.

Instead, rather than "remove known bad data", it's better to "remove everything but known good data": this distinction is crucial. Since - in our example - an email address can contain only these characters:

```
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
0123456789
@.-_+
```

There is really no benefit in allowing characters that could not be valid, and rejecting them early - presumably with an error message - not only helps forestall SQL Injection, but also catches mere typos early rather than stores them into the database.

Be aware that "sanitizing the input" doesn't mean merely "remove the quotes", because even "regular" characters can be troublesome. In an example where an integer ID value is being compared against the user input (say, a numeric PIN):

```
SELECT fieldlist
  FROM table
 WHERE id = 23 OR 1=1;  -- Boom! Always matches!
```

In practice, however, this approach is highly limited because there are so few fields for which it's possible to outright exclude many of the dangerous characters. For "dates" or "email addresses" or "integers" it may have merit, but for any kind of real application, one simply cannot avoid the other mitigations.

5.5.1.3.2. Escape/Quotesafe the Input

Even if one might be able to sanitize a phone number or email address, one cannot take this approach with a "name" field lest one wishes to exclude the likes of Bill O'Reilly from one's application: a quote is simply a valid character for this field.

One includes an actual single quote in an SQL string by putting two of them together, so this suggests the obvious - but wrong! - technique of preprocessing every string to replicate the single quotes:

```
SELECT fieldlist
  FROM customers
 WHERE name = 'Bill O''Reilly';  -- works OK
```

However, this naive approach can be beaten because most databases support other string escape mechanisms. MySQL, for instance, also permits \' to escape a quote, so after input of \'; DROP TABLE users; -- is "protected" by doubling the quotes, we get:

```
SELECT fieldlist
  FROM customers
 WHERE name = '\''; DROP TABLE users; --';  -- Boom!
```

The expression '\'' is a complete string (containing just one single quote), and the usual SQL shenanigans follow. It doesn't stop with backslashes either: there is Unicode, other encodings, and parsing oddities all hiding in the weeds to trip up the application designer.

Getting quotes right is notoriously difficult, which is why many database interface languages provide a function that does it for you. When the same internal code is used for "string quoting" and "string parsing", it's much more likely that the process will be done properly and safely. [9]

Some examples are the MySQL function **mysql_real_escape_string()** and Perl DBD method **$dbh->quote($value).** These methods must be used.

5.5.1.3.3. Use Bound Parameters (the PREPARE Statement)

Though quotesafing is a good mechanism, we're still in the area of "considering user input as SQL", and a much better approach exists: bound parameters, which are supported by essentially all database programming interfaces. In this technique, an SQL statement string is created with placeholders - a question mark for each parameter - and it's compiled ("prepared", in SQL parlance) into an internal form. Later, this prepared query is "executed" with a list of parameters:

Example in Perl

```
   $sth->execute ($email);
```

Thanks to Stefan Wagner, this demonstrates bound parameters in Java:

Insecure version

```
   ResultSet rs = s.executeQuery("SELECT email FROM
member WHERE name = "
                          + formField); // *boom*
```

Secure version

```
      "SELECT email FROM member WHERE name = ?");
   ps.setString(1, formField);
   ResultSet rs = ps.executeQuery();
```

Here, $email is the data obtained from the user's form, and it is passed as positional parameter #1 (the first question mark), and at no point do the contents of this variable have anything to do with SQL statement parsing. Quotes, semicolons, backslashes, SQL comment notation - none of this has any impact, because it's "just data". There simply is nothing to subvert, so the application is largely immune to SQL injection attacks.

There also may be some performance benefits if this prepared query is reused multiple times (it only has to be parsed once), but this is minor compared to the enormous security benefits. This is probably the single most important step one can take to secure a web application.

5.5.1.3.4. Limit Database Permissions and Segregate Users

In the case at hand, we observed just two interactions that are made not in the context of a logged-in user: "log in" and "send me password". The web application ought to use a database connection with the most limited rights possible: query-only access to the members table, and no access to any other table.

The effect here is that even a "successful" SQL injection attack is going to have much more limited success. Here, we'd not have been able to do the UPDATE request that ultimately granted us access, so we'd have had to resort to other avenues.

Once the web application determined that a set of valid credentials had been passed via the login form, it would then switch that session to a database connection with more rights.

It should go almost without saying that rights should never be used for any web-based application.

5.5.1.3.5. Use Stored Procedures for Database Access

When the database server supports them, use stored procedures for performing access on the application's behalf, which can eliminate SQL entirely (assuming the stored procedures themselves are written properly). By encapsulating the rules for a certain action-query, update, delete, etc. Into a single procedure, it can be tested and documented on a standalone basis and business rules enforced (for instance, the "add new order" procedure might reject that order if the customer were over his credit limit).

For simple queries this might be only a minor benefit, but as the operations become more complicated (or are used in more than one place), having a single definition for the operation means it's going to be more robust and easier to maintain.

Note: it's always possible to write a stored procedure that itself constructs a query dynamically: this provides no protection against SQL Injection - it's only proper binding with prepare/execute or direct SQL statements with bound variables that provide this protection.

5.5.1.3.6. Isolate the Webserver

Even having taken all these mitigation steps, it's nevertheless still possible to miss something and leave the server open to compromise. One ought to design the network infrastructure to assume that the hacker will have full administrator access to the machine, and then attempt to limit how that can be leveraged to compromise other things.

For instance, putting the machine in a DMZ with extremely limited pinholes "inside" the network means that even getting complete control of the webserver doesn't automatically grant full access to everything else. This won't stop everything, of course, but it makes it a lot harder.

5.5.1.3.7. Configure Error Reporting

The default error reporting for some frameworks includes developer debugging information, and this cannot be shown to outside users. Imagine how much easier a time it makes for an attacker if the full query is shown, pointing to the syntax error involved. This information is useful to developers, but it should be restricted - if possible - to just internal users [9].

5.5.2. Cross Site Scripting

The number of findings is 623 and the severity is high.

5.5.2.1. Description

This script is possibly vulnerable to Cross Site Scripting (XSS) attacks.

Cross site scripting (also referred to as XSS) is a vulnerability that allows an attacker to send malicious code (usually in the form of JavaScript) to another user. Because a browser cannot know if the script should be trusted or not, it will execute the script in the user context allowing the attacker to access any cookies or session tokens retained by the browser.

5.5.2.2. Impact

Malicious users may inject JavaScript, VBScript, ActiveX, HTML or Flash into a vulnerable application to fool a user in order to gather data from them. An attacker can steal the session cookie and take over the account, impersonating the user. It is also possible to modify the content of the page presented to the user.

5.5.2.3. Cross Site Scripting Avoidances

The script should filter metacharacters from user input.

5.5.2.3.1. Advice.

There is a simple advice. Never trust user input and always filter metacharacters. This will eliminate the majority of XSS attacks.

5.5.2.3.2. Parameters Conversion.

Remember XSS holes can be damaging and costly to business if abused. Often attackers will disclose these holes to the public, which can erode customer and public confidence in the security and privacy of organization's site. Filtering < and > alone will not solve all cross site scripting attacks and it is suggested you also attempt to filter out (and) by translating them to &#40; and &#41;, and also # and & by translating them to &#35 (#) and &#38 (&)[8].

5.5.3. Backup files

The number of finding is 10 and the severity is medium.

5.5.3.1. Description

A possible backup file was found on SUST webserver. These files are usually created by developers to backup their work

5.5.3.2. Impact

Backup files can contain script sources, configuration files or other sensitive information that may help malicious user to prepare more advanced attacks.

5.5.3.3. Protection

Remove the file(s) if they are not required on SUST website. As an additional step, it is recommended to implement a security policy within SUST organization to disallow creation of backup files in directories accessible from the web [7].

CHPTER SIX

CONCLUSION and RECOMMENDATION

6. CONCLUSION and RECOMMENDATION

6.1. Conclusion

The (CEH) certified Ethical Hacker methodology for penetration testing has been chosen due to simplicity and availability of references and tools. Using common gathering information tools in Footprinting phase and Acunetx WVS for the scanning phase on SUST network, results in a rich information report.

The report generated shows in, chapter five, a high level of risk with large amount of information that describes the current security situation of the SUST network and represents how much SUST network is vulnerable to attacks of malicious hackers. This information could become very dangerous to SUST network if gathered by a hacker or any malicious hand. The good thing is that this information can help SUST network security administrators to tighten up their security system as we recommended below.

6.2. Recommendations

In the future if someone interesting in this research area and for SUST network administrators and developers we recommend the following:

6.2.1. For the network administrators and developers we recommend that they learn and follow the “Acunetix” results in the reports and follow the recommendations below.

6.2.1.1. For SQL Injection Mitigations do [9]:

1. Sanitize the input.
2. Escape /Quotesafe the input.
3. Use bound parameters (the PREPARE statement).
4. Limit database permissions and segregate users.
5. Use stored procedures for database access.

6. Isolate the webserver from the database server.
7. Configure error reporting.
8. Parameterized Queries with Bound Parameters.
9. Careful use of parameterized stored procedures.

6.2.1.2. For Cross Site Scripting do [8]:

1. Never trust user input and always filter metacharacters.
2. Converting < and > to &lt; and &gt; is also suggested when it comes to script output.

6.2.1.3. For Backup files do [7]:

1. Remove the file(s) if they are not required on SUST website.
2. Implement a security policy within SUST organization to disallow creation of backup files in directories accessible from the web.

6.2.2. Recommendations for the SUST Computer Center managers and network administrators and developers:

1. Train and Increase the knowledge, skills and capacity of the administrators in the fields of IT security, security management, system hardening and security testing.
2. Establishment of monitoring and auditing and test systems for SUST network.
3. Do the other Types of Ethical Hacks like Local Network (internal penetration testing) and Social Engineering testing.
4. Do a periodic audit and performing penetration testing for SUST network according to a plan prepared in advance and executed, maintained and improved by the senior administrators.
5. Serious attention to the results of these audits and tests.
6. Secure the University network well by re-modifying firewall IDS and IPS and configure them properly.
7. Enhance (RAS Remote Access Server) to increase the security layers.

8. Issues like physical security of the Data Center of SUST network, besides the availability of the servers and business continuity of the service offered by SUST network should be examined and analyzed for future work.

6.2.3. Recommendations for the sake of science and research:

1. Work should be extended on this project in the near future so as to complete the remaining stages (stage three, four and five) and the use of other tools that have not been used here like Nessus, Solarwinds, Metasploit framework and Nikto [16].
2. The research area of ethical hacking and penetration testing need to be emphasized.

REFERENCES

[1] Chris McNab*," Network Security Assessment",* O'Reilly Media, Inc., ISBN-13: 978-0-596-51030-5, 2nd ed, 2008.

[2] Kimberly Graves, "*CEH Official Certified Ethical Hacker Review Guide* EC-COUNCIL CEH EXAM 312-50", Wiley Publishing, Inc. , ISBN-13: 978-0-7821-4437-6, 2007.

[3] Thomas Ruder*, "A Guide to Penetration Testing* ", http://www.crazytrain.com/penetration.html, visited on 17/11/2010 at 12:24 pm.

[4] Kevin Beaver, "*Hacking For Dummies"*, Wiley Publishing, Inc., ISBN: 0-7645-5784-X, 2004.

[5] Corsaire security consultancy, " *Penetration testing guide* ", http://www.penetration-testing.com/, visited on 18/11/2010 at 10:40 am.

[6] legal definitions and legal terms dictionary of U.S. Legal Forms, Inc., "*computer hacking law and legal definition*", http://definitions.uslegal.com/c/computer-hacking/ , visited on 17/11/2010 at 05:40 pm

[7] Sudhanshu Kairab, *"A practical guide to security assessments",* CRC Press LLC, ISBN 0-8493-1706-1, 2005.

[8] Robert Auger, " *THE CROSS-SITE SCRIPTING (XSS) FAQ* ", http://www.cgisecurity.com/xss-faq.html , visited on 17/11/2010 at 5:55 pm

[9] Steve Friedl, "*SQL Injection Attacks by Example*", http://www.unixwiz.net/techtips/sql-injection.html, visited on 17/11/2010 at 10:09 pm

[10] Manish Sharma, Web Developers Notes , "*What is WHOIS?*" http://www.webdevelopersnotes.com/hosting/what_is_whois.php3, visited on 17/11/2010 at 05:30 pm.

[11] Visualware, Inc., http://www.visualroute.com/detail.html visited on 17/11/2010 at 08:30 pm

[12] Acunetix, http://www.acunetix.com/, visited on 09/11/2010 at 10:30 am.

[13] Microsoft Windows XP, Version 5.1.2600, 2001.

[14] Ralf D. Kloth, Ludwigsburg, DE (QRQ.software), http://www.kloth.net/services/nslookup.php, looking for [sustech.edu] with parameter ANY (any type), visited on 07/11/2010 5:00pm.

[15] DomainTools, LLC, http://whois.domaintools.com/sustech.edu, looking for [sustech.edu], visited on 07/11/2010 5:15pm.

[16] Gordon Lyon," *Top 100 Network Security Tools "*http://sectools.org/, visited on 07/11/2010 5:15pm.

BIBLIOGRAPHY

Listed by important.

[1] Steve Manzuik, André Gold, et al, Network *Security Assessment: From Vulnerability to Patch* Syngress Publishing,Inc, ISBN-13:978-1-59749-101-3, 2007.

[2] Russ Rogers, Greg Miles, et al, *Network Security Evaluation Using the NSA IEM*, Syngress Publishing,Inc, ISBN:1-597490-35-0, 2005.

[3] Bryan Burns, Jennifer Stisa Granick, et al, *Security Power Tools*, O'Reilly Media, Inc, ISBN:1-597490-35-0, 2007.

[4] Pete Herzog, *OSSTMM 2.2. Open-Source Security Testing Methodology Manual*, http://www.isecom.org/osstmm/, 2006

[5] Karen Scarfone, Murugiah Souppaya, et al, *Technical Guide to Information Security Testing and Assessment*, Recommendations of the National Institute of Standards and Technology (NIST) (Special Publication 800-115), 2008.

APPENDICES

A. ACUNETIX EXECUTIVE SUMMARY REPORT

B. ACUNETIX OTHER REPORTS SAMPLE