Geetanjali Institute of Technical Studies

(Approved by AICTE, New Delhi and Affiliated to Rajasthan Technical University Kota (Raj.))

DABOK, UDAIPUR, RAJASTHAN 313022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B. Tech - VII SEMESTER

ACADEMIC YEAR – 2025-26

CYBER SECURITY LAB

7CS4-22

Submitted to:
Ms. Ayushi Ghill and Ms. Upasana Ameta
Assistant Professor,CSE
GITS

Submitted by:

Roll Number:

INDEX – LAB MANUAL

S. No.	CONTENT / ITEM NO.	PAGE NO.
1.	Vision and Mission of The Institute	3
2.	Vision and Mission of The Department	3
3.	Program Educational Objective of Department (PEO's)	3
4.	Program Outcomes of Department (PO's)	5
5.	Course Outcome (COs)	6
6.	COs mapping with Pos and PSOs	6
7.	Course Syllabus	7
8.	Prescribed Books	7
9.	List of Experiment	8
10.	Practical's Beyond RTU Syllabus	8
11.	Experiment No.1	9
12.	Experiment No.2	12
13.	Experiment No.3	14
14.	Experiment No.4	16
15.	Experiment No.5	26
16.	Experiment No.6	29
17.	Experiment No.7	33
18.	Experiment No.8	37
19.	Outcome of Lab	47
20.	Computer Lab's Do's and Don'ts and Safety Rules	48

VISSION & MISSION OF INSTITUTE

INSTITUTE VISION

TO ACHIEVE EXCELLENCE IN TECHNICAL AND MANAGEMENT EDUCATION THROUGH QUALITY TEACHING AND INNOVATION.

INSTITUTE MISSION

M1: To provide an excellent learning environment to produce socially responsible and productive technical professionals.

M2: To set up the state-of-the-art facilities for quality education and innovation.

M3: To impart knowledge & Skills leading to shaping a budding manager as a quality executive.

M4: To encourage for life-long learning and team-based problem solving through learning environment.

VISION & MISSION OF DEPARTMENT

VISION

To nurture the students to become employable graduates who can provide solutions to the societal issues through ICT.

MISSION

M1: To focus on practical approach towards learning and exposing the students on the latest ICT technologies.

M2: To foster logical thinking among the students to solve real-time problems using innovative approaches.

M3: To provide state-of-the-art resources that contributes to inculcate ethical and life-long learning environment.

PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

The Programme Educational Objectives of the programme offered by the department are listed below:

PEO1: To enable the students to think out-of-the-box solutions for addressing societal issues through ICT.

PEO2: To impart skills in students to analyze, design and implement Software/Hardware solutions to solve interdisciplinary and complex problems.

PEO3: To expose the students towards effective dissemination of research findings in order to become successful entrepreneurs or to pursue higher education.

PROGRAM SPECIFIC OUTCOMES (PSO's)

- **PSO1: Professional Skills:** The ability to understand, analyze and develop electronic systems in the areas related to hardware and software development, communication systems and networking for efficient design of electronic-based systems of varying complexity.

- **PSO2: Problem-Solving Skills:** The ability to apply standard practices and strategies in electronic system project development on both hardware and software environments to deliver a quality product for business success.

- **PSO3: Successful Career and Entrepreneurship:** The ability to employ modern electronic solutions on different platforms, in creating innovative career paths to be an entrepreneur, and a zest for higher studies.

PROGRAMME OUTCOMES (POs)

A student will develop:

1. **ENGINEERING KNOWLEDGE:** An ability to apply knowledge of Mathematics, Science and Engineering Fundamentals in Electronics and Communication Engineering.
2. **PROBLEM ANALYSIS:** An ability to analyze and interpret data by designing and conducting experiments. Develop the knowledge of developing algorithms, designing, implementation and testing applications in electronics and communication related areas.
3. **DESIGN/ DEVELOPMENT OF SOLUTION:** An ability to Design a system Component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability.
4. **CONDUCTION OF INVESTIGATION OF COMPLEX PROBLEMS:** An ability to Identify, formulate and solve engineering problems.
5. **MODERN TOOL USAGE:** An ability to use the techniques, skills and modern engineering tools necessary for engineering practice.
6. **THE ENGINEERING AND SOCIETY:** Broad education necessary to understand the impact of engineering solutions in a global, economic, environmental and societal context.
7. **ENVIRONMENT & SUSTAINABILITY:** Understand the impact of professional engineering solution in societal and environmental contexts, and demonstrate the knowledge of, and need of sustainable development.
8. **ETHICS:** An ability to understand the professional, social and ethical responsibility.
9. **INDIVIDUAL AND TEAM WORK:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **COMMUNICATION:** An ability to Communicate effectively in order to succeed in their profession such as, being able to write effective reports and design documentation, make effective presentations.
11. **PROJECT MANAGEMENT & FINANCE:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in team, to manage projects and in multidisciplinary environment.
12. **LIFE-LONG LEARNING:** Recognize the need and an ability to engage in life-long learning.

COURSE OUTCOMES (COs)

CO1	Students will be able to solve and relate mathematic concepts behind the cryptographic algorithms.
CO2	Students will be able to explain basic operations of cryptographic algorithms.
CO3	Students will be able to describe various private and public key security algorithms used for network security along with its encryption and decryption.
CO4	Students will be able to evaluate various scenarios and apply the required type of algorithm for ensuring security.
CO5	Students will be able analyze protocols for various security objectives with cryptographic tools.

COs MAPPING WITH POs AND PSOs

Course Outcome	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO1	3	2	2	1	1	0	0	0	0	0	0	1	2	1	1
CO2	2	3	3	2	2	0	0	0	0	0	0	1	3	2	2
CO3	1	1	2	2	1	0	0	0	0	0	0	1	2	1	1
CO4	1	1	2	1	0	0	0	0	0	0	0	1	1	2	2
CO5	2	1	2	2	2	0	0	0	0	0	0	1	2	1	1

COURSE SYLLABUS

RAJASTHAN TECHNICAL UNIVERSITY, KOTA

Scheme & Syllabus

IV Year- VII Semester: B. Tech. (Computer Science & Engineering)

7CS4-22: Cyber Security Lab

Credit: 2 **Max. Marks: 100(IA:60, ETE:40)**

0L+0T+4P **End Term Exam: 2 Hours**

SN	List of Experiments
1	Implement the following Substitution & Transposition Techniques concepts: a) Caesar Cipherb) Rail fence row & Column Transformation
2	Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).
3	Implement the following Attack: a) Dictionary Attack b) Brute Force Attack
4	Installation of Wire shark, tcpdump, etc and observe data transferred in client server communication using UDP/TCP and identify the UDP/TCP datagram.
5	Installation of rootkits and study about the variety of options.
6	Perform an Experiment to Sniff Traffic using ARP Poisoning.
7	Demonstrate intrusion detection system using any tool (snort or any other s/w).
8	Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures.
	PROJECT: In a small area location such as a house, office or in a classroom, there is a small network called a Local Area Network (LAN). The project aims to transfer a file peer-to-peer from one computer to another computer in the same LAN. It provides the necessary authentication for file transferring in the network transmission. By implementing the Server-Client technology, use a File Transfer Protocol mechanism and through socket programming, the end user is able to send and receive the encrypted and decrypted file in the LAN. An additional aim of the project is to transfer a file between computers securely in LANs. Elements of security are needed in the project because securing the files is an important task, which ensures files are not captured or altered by anyone on the same network. Whenever you transmit files over a network, there is a good chance your data will be encrypted by encryption technique. Any algorithm like AES is used to encrypt the file that needs to transfer to another computer. The encrypted file is then sent to a receiver computer and will need to be decrypted before the user can open the file.

TEXT / REFERENCE BOOKS

- The Network Security Test Lab: A Step-by-Step Guide, Wiley, Michael Gregg
- AtulKahate, Cryptography and Network Security, TMH Publication.

Scheme & Syllabus of 4 Year B. Tech. (CS) for students admitted in Session 2022-23

7CS4-22: Cyber Security Lab

Credit: 2 **Max. Marks: 100 (IA: 60, ETE: 40) 0L+0T+4P**
End Term Exam: 2 Hours

S.NO.	NAME OF EXPERIMENT	CO Mapped
1	Implement the following Substitution & Transposition Techniques concepts: a) Caesar Cipher b) Rail fence row & Column Transformation	**CO2**
2	Implement the Diffie-Hellman Key Exchange mechanism Consider the end user as one of the parties (Alice) and theother party (bob).	**CO3**
3	Implement the following Attack: Brute Force Attack	**CO1**
4	Installation of Wire shark, tcpdump, etc and observe data transferred in client server communication using UDP/TCP and identify the UDP/TCP datagram.	**CO2**
5	Installation of rootkits and study about the variety of options.	**CO5**
6	Perform an Experiment to Sniff Traffic using ARP Poisoning.	**CO4**
7.	Demonstrate intrusion detection system using any tool (snort or any other s/w).	**CO5**
8.	Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures.	**CO4**

Prerequisites:

- Students are expected to have knowledgeofNetwork and Cyber Security.
- Students are expected to have knowledge of C, Java Programing.

Software requirements:

- Turbo C/ Borland C++ 5.02 or Higher
- JDK Java

PRACTICALS BEYOND RTU SYLLABUS
Not Applicable

EXPERIMENT NO. 1

AIM: Implement the following Substitution & Transposition Techniques concepts:

a) Caesar Cipher

b) Rail fence row & Column Transformation

PROGRAM: (a)

```
#include<stdio.h>
int main()
{
        char msg[100], ch;
        int i,key;
        printf("Enter an Encrypted text:");
        gets(msg);
        printf("Enter key:");
        scanf("%d",&key);
        for(i=0; msg[i]!='\0'; i++)
       {
                ch=msg[i];
                if(ch>='a'&&ch<='z')
                {
                        ch=ch-key;
                        if(ch<'a')
                        {
                                        ch=ch+'z'-'a'+1;
                        }
                                        msg[i]=ch;
                }
                else if(ch>='A'&& ch<'Z')
                {
                        ch=ch+key;
                   if(ch<'A')
                   {
                                        ch=ch+'Z'-'A'+1;
                   }
                                msg[i]=ch;
        }
   }
                   printf("Decrypted message: %s", msg);
   return 0;
}
```

CODE OUTPUT: (a)

```
Enter an Encrypted text:Hello how are you
Enter key:3
Decrypted message: Kbiil elt xob vlr
--------------------------------
Process exited after 13.04 seconds with return value 0
Press any key to continue . . .
```

PROGRAM: (b)

```
#include<stdio.h>
#include<string.h>
void main()
{
	int i,j,k,l;
	char a[20],c[20],d[20];
	printf("\n\t\t RAIL FENCE TECHNIQUE");
	printf("\n\nEnter the input string : ");
	gets(a);
	l=strlen(a);
	/*Ciphering*/
	for(i=0, j=0;i<l; i++)
	{
		if(i%2==0)
		c[j++]=a[i];
	}
	for(i=0;i<l;i++)
	{
		if(i%2==1)
		c[j++]=a[i];
	}
	c[j]='\0';
	printf("\nCipher text after applying rail fence :");
	printf("\n%s",c);
	/*Deciphering*/
	if(l%2==0)
		k=l/2;
	else
		k=(l/2)+1;
	for(i=0,j=0;i<k;i++)
	{
		d[j]=c[i];
		j=j+2;
	}
	for(i=k,j=1;i<l;i++)
	{
		d[j]=c[i];
```

```
            j=j+2;
        }
        d[l]='\0';
        printf("\nText after decryption : ");
        printf("%s",d);
}
```

CODE OUTPUT: (b)

EXPERIMENT NO. 2

AIM: Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider the end user as one of the parties (Alice) and the JavaScript application as other party (bob).

PROGRAM:

```
#include<stdio.h>
#include<conio.h>
long long int power(int a, int b, int mod)
{
	long long int t;
	if(b==1)
			return a;
	t=power(a,b/2,mod);
	if(b%2==0)
		return (t*t)%mod;
	else
		return (((t*t)%mod)*a)%mod;
}
long int calculateKey(int a, int x, int n)
{
	return power(a,x,n);
}
void main()
{
	int n,g,x,a,y,b;
	printf("Enter the value of n and g : ");
	scanf("%d%d",&n,&g);
	printf("Enter the value of x for the first person : ");
	scanf("%d",&x);
	a=power(g,x,n);
	printf("Enter the value of y for the second person : ");
	scanf("%d",&y);
	b=power(g,y,n);
	printf("key for the first person is :%lld\n",power(b,x,n));
	printf("key for the second person is :%lld\n",power(a,y,n));
	getch();
}
```

CODE OUTPUT:

```
Enter the value of n and g : 23
9
Enter the value of x for the first person : 4
Enter the value of y for the second person : 3
key for the first person is :9
key for the second person is :9
```

EXPERIMENT NO. 3

AIM: Implement the following Attack: Brute Force Attack

PROGRAM:

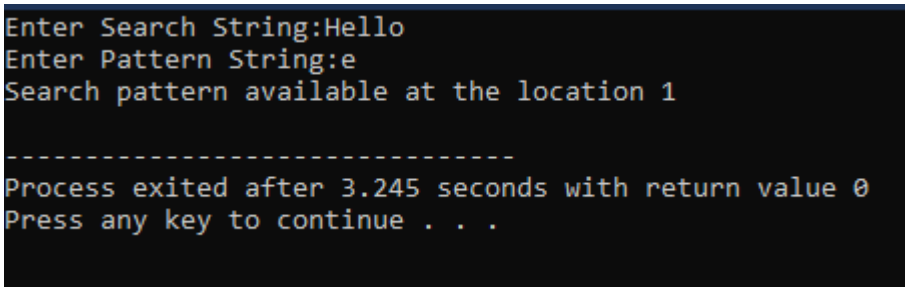
```
#include <stdio.h>
 #include <string.h>
 #define MAX 100

 /* try to find the given pattern in the search string */
int bruteForce(char *search, char *pattern, int slen, int plen) {
int i, j, k;

      for (i = 0; i<= slen - plen; i++) {
            for (j = 0, k = i; (search[k] == pattern[j]) && (j <plen); j++, k++);
            if (j == plen)
                  return j;
      }
      return -1;
 }

int main()
{
      char searchStr[MAX], pattern[MAX];
      int res;
      printf("Enter Search String:");
      fgets(searchStr, MAX, stdin);
      printf("Enter Pattern String:");
     fgets(pattern, MAX, stdin);
     searchStr[strlen(searchStr) - 1] = '\0';
      pattern[strlen(pattern) - 1] = '\0';
      res = bruteForce(searchStr, pattern, strlen(searchStr), strlen(pattern));
      if (res == -1)
      {
         printf("Search pattern is not available\n");
      }
 else
{
         printf("Search pattern available at the location %d\n", res);
 }
      return 0;
}
```

CODE OUTPUT:


Enter Search String:Hello
Enter Pattern String:e
Search pattern available at the location 1

Process exited after 3.245 seconds with return value 0
Press any key to continue . . .

EXPERIMENT NO. 4

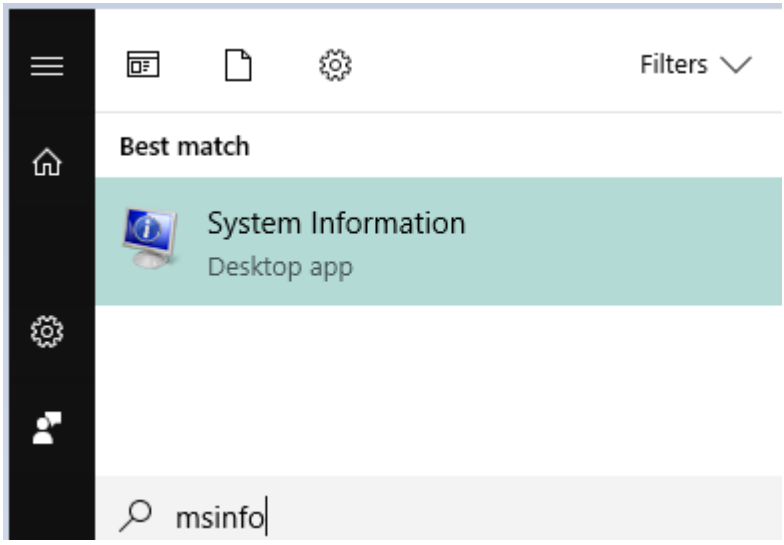
AIM: Installation of Wire shark, tcpdump,

PROGRAM:

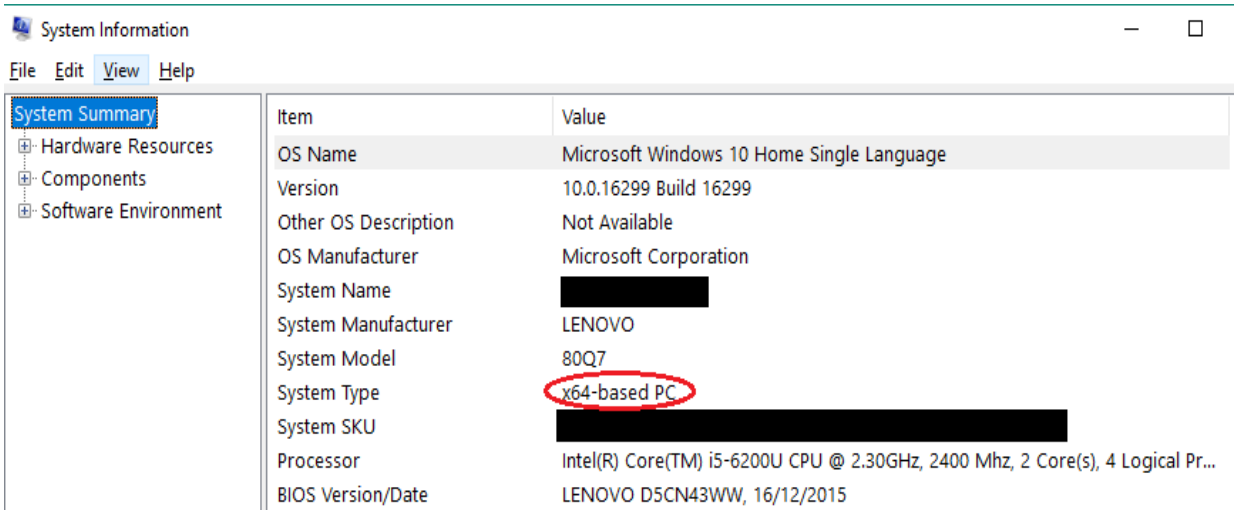
Wireshark is an invaluable packet analyzer used for network troubleshooting and analysis. This tutorial will show you how to download and install Wireshark in a Windows 10 system.

Requirements:
Before installing Wireshark, Determine your system type by pressing the Windows button and typing ‘msinfo’.

CODE OUTPUT:



The Windows system has two types which can be X86-based PC (32-bit system) or X64-based PC (64-bit system).

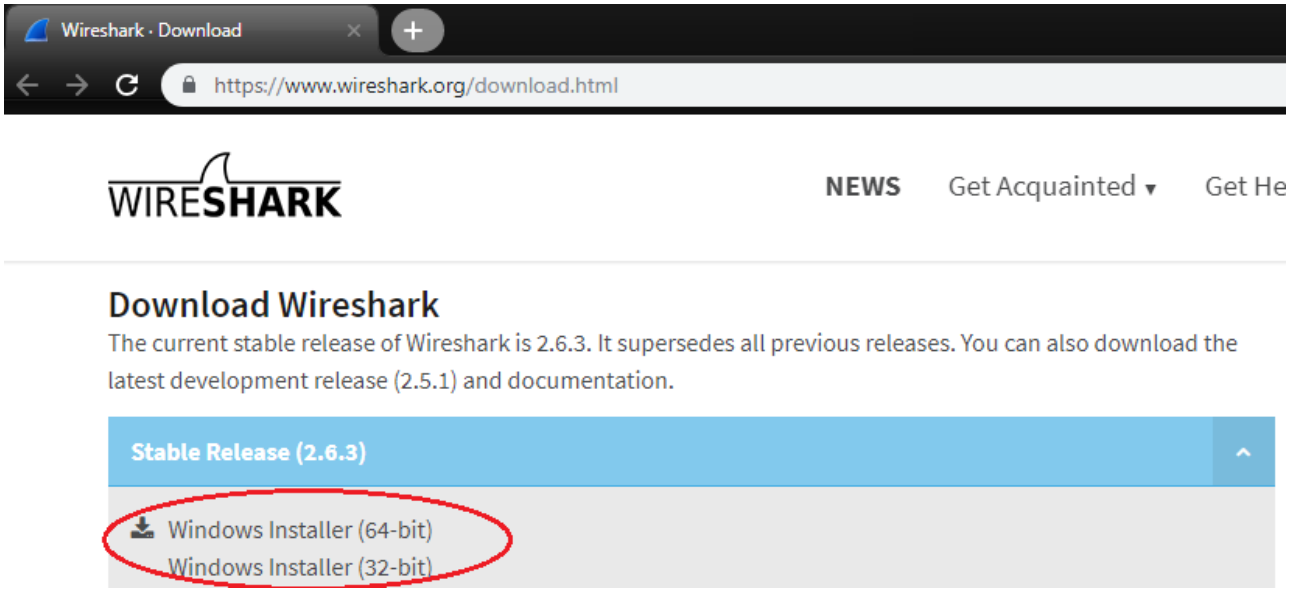


Installing Wireshark on Windows 10

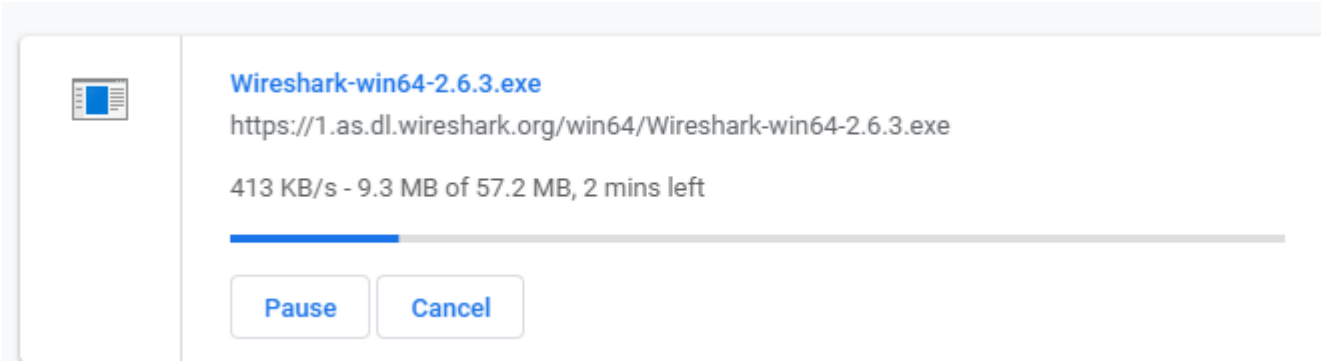
1. Select the Wireshark Windows Installer matching your system type, either 32-bit or 64-bit as determined.

2. Download and install Wireshark

NOTE: If you are a beginner with using Wireshark, please select the stable release version.



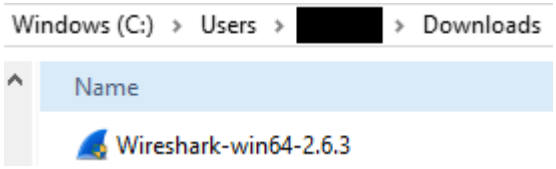
3. The download should start automatically once you selected the compatible Windows Installer for your Windows 10 platform.



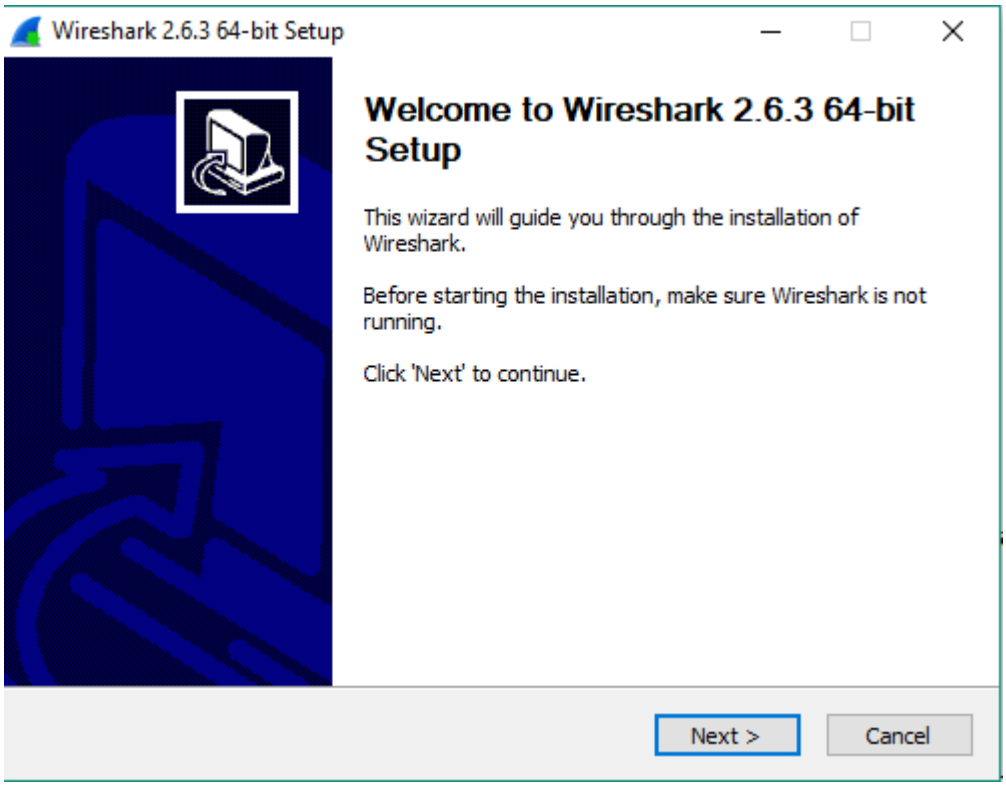
4. Save the program in the Downloads folder, then Close the web browser.

Install Wireshark

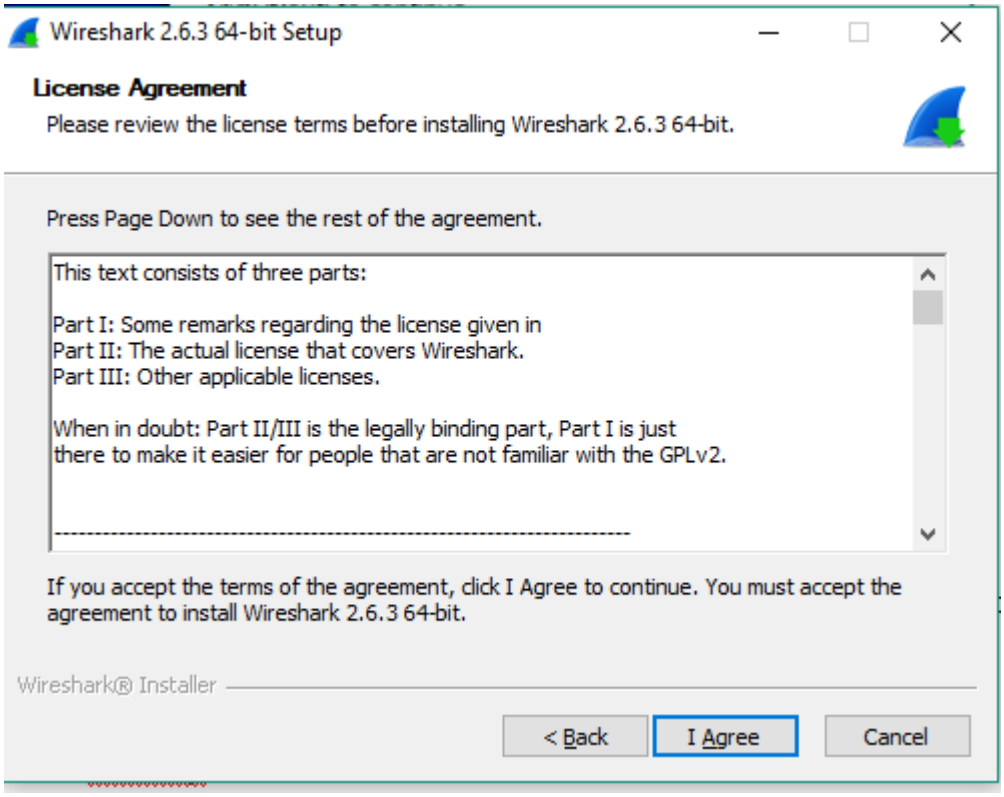
1. Open Windows Explorer.
2. Select the Downloads folder.



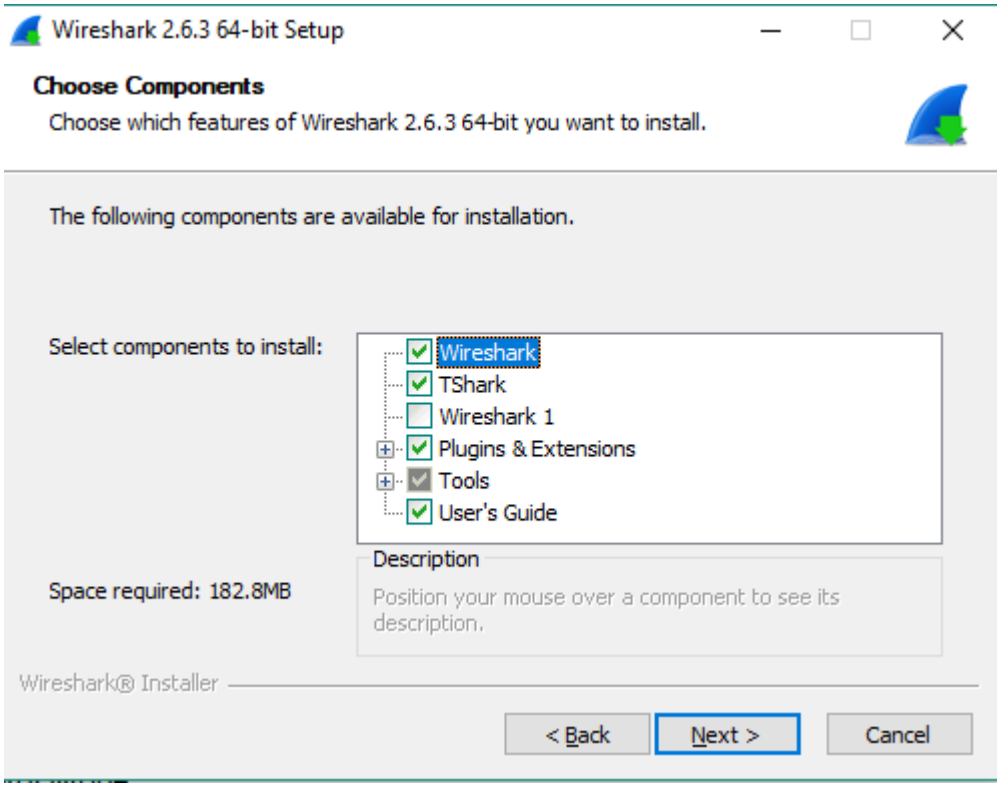
3. Locate the version of Wireshark you downloaded.
4. Double-click on the file to open it. If you see a User Account Control dialog box, select Yes to allow the program to make changes to this computer.
5. Select Next to start the Setup Wizard.



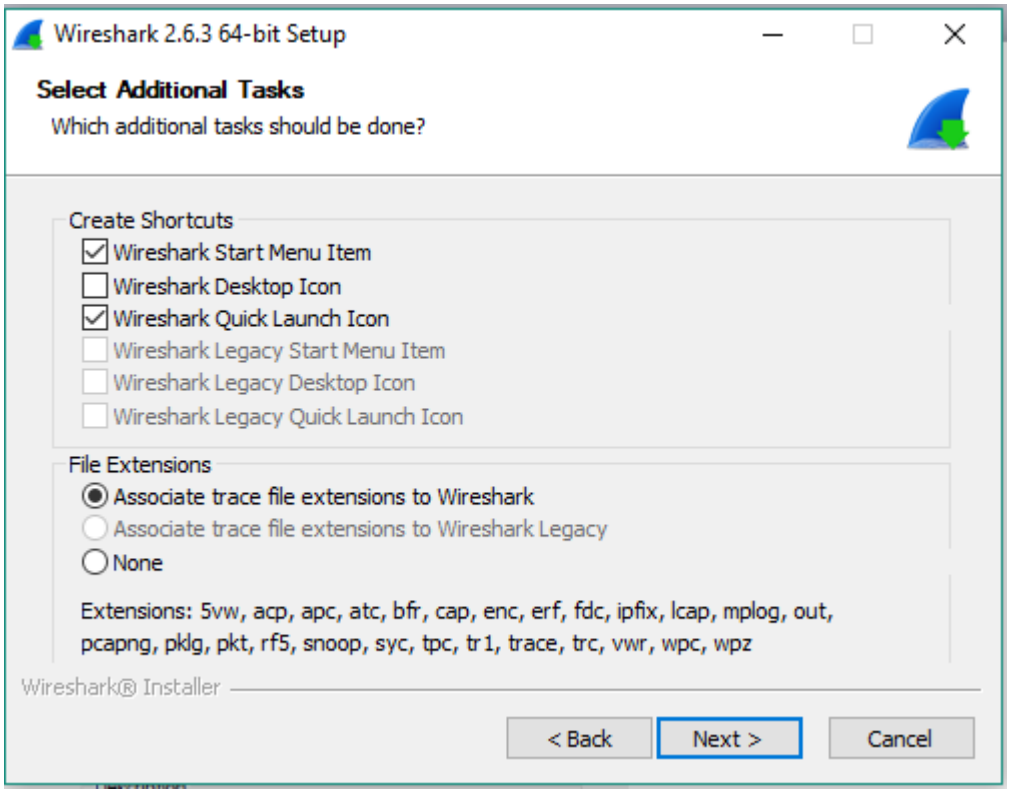
6. Review the license agreement. If you agree, select I Agree to continue.



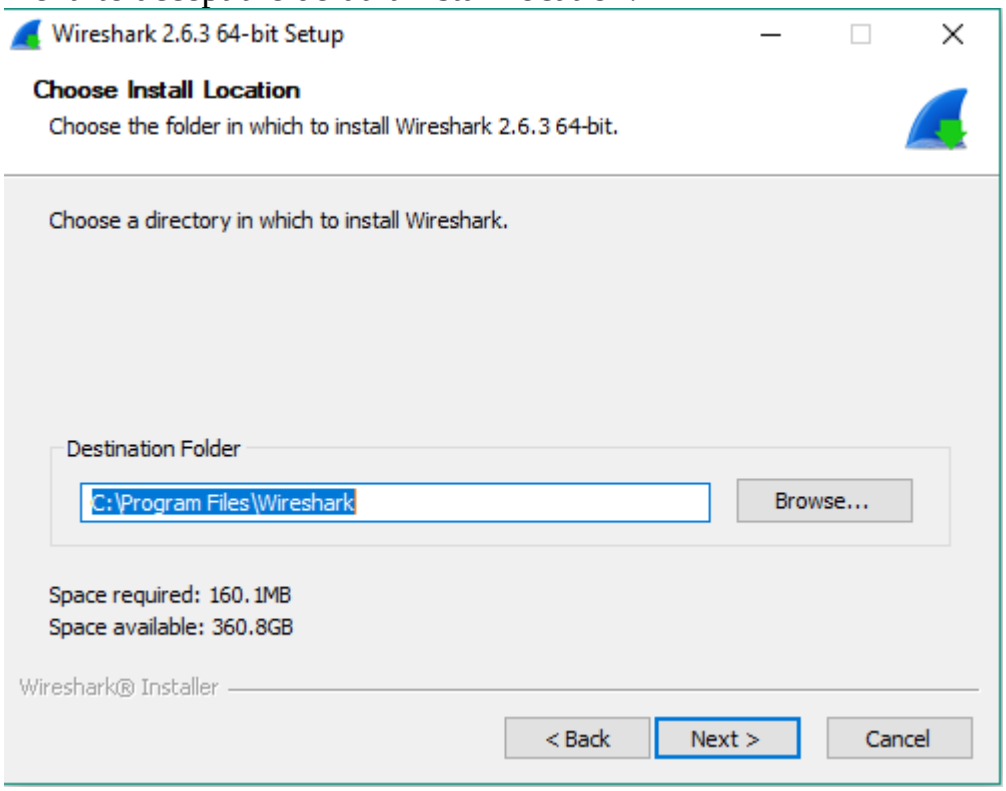
7. Select “Next” to accept the default components.



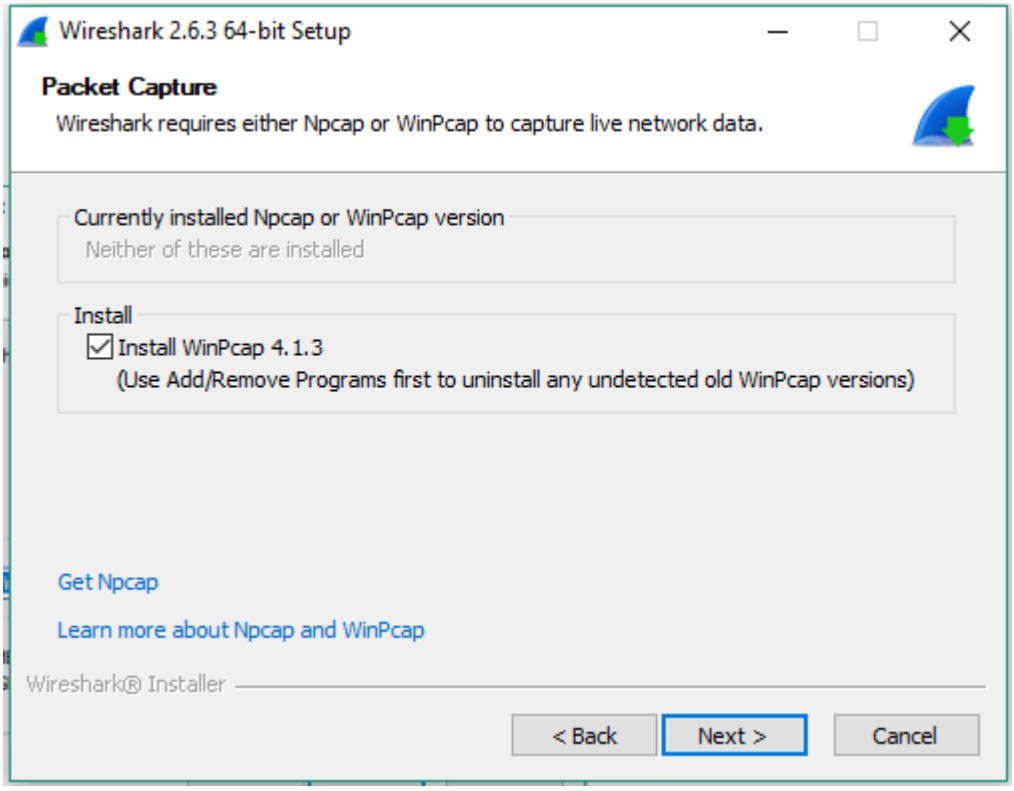
8. Select the shortcuts you would like to have created. Leave the file extensions selected. Select Next to continue.



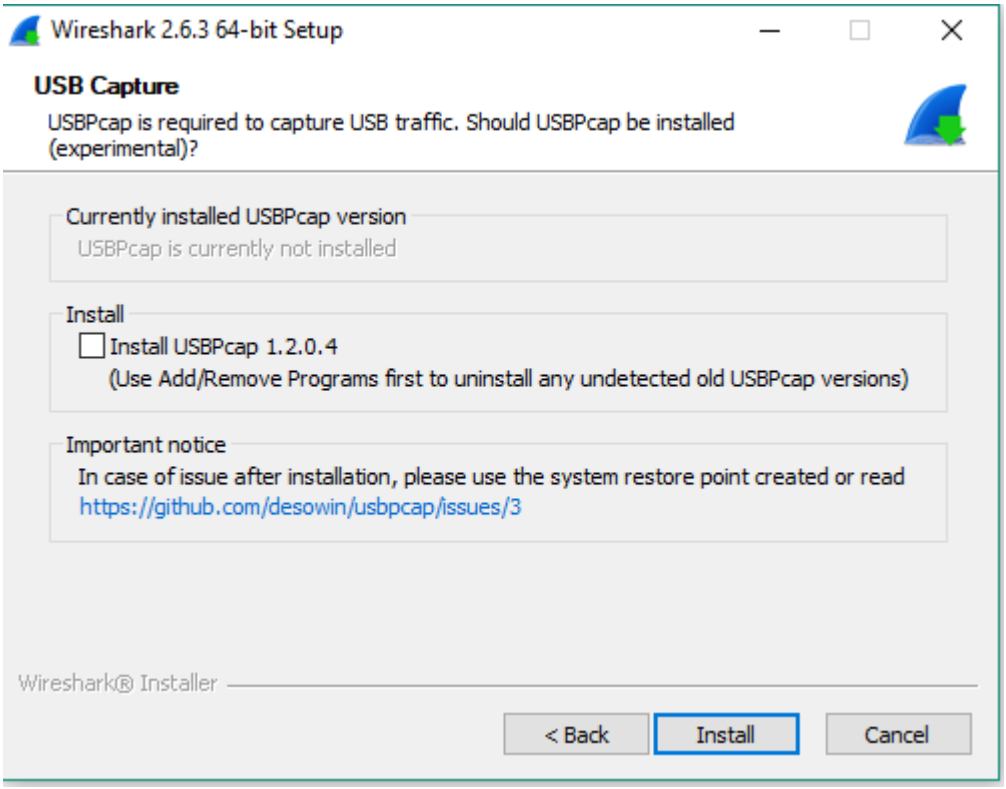
9. Select Next to accept the default install location.



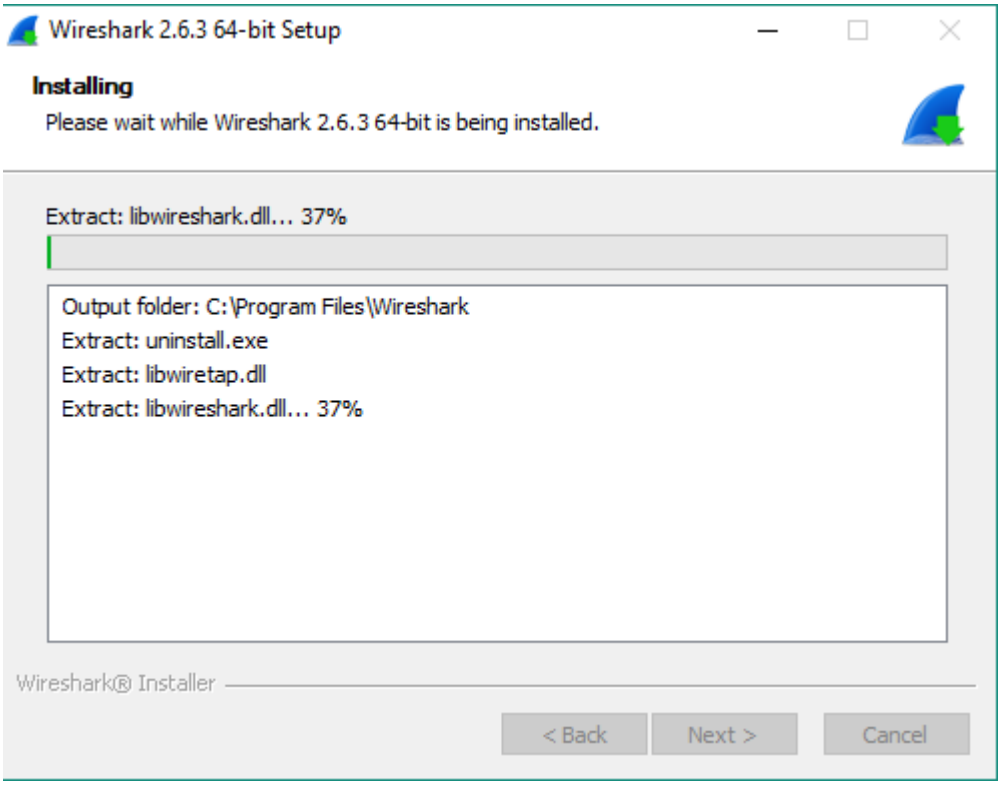
10. Select Next to install WinPcap.



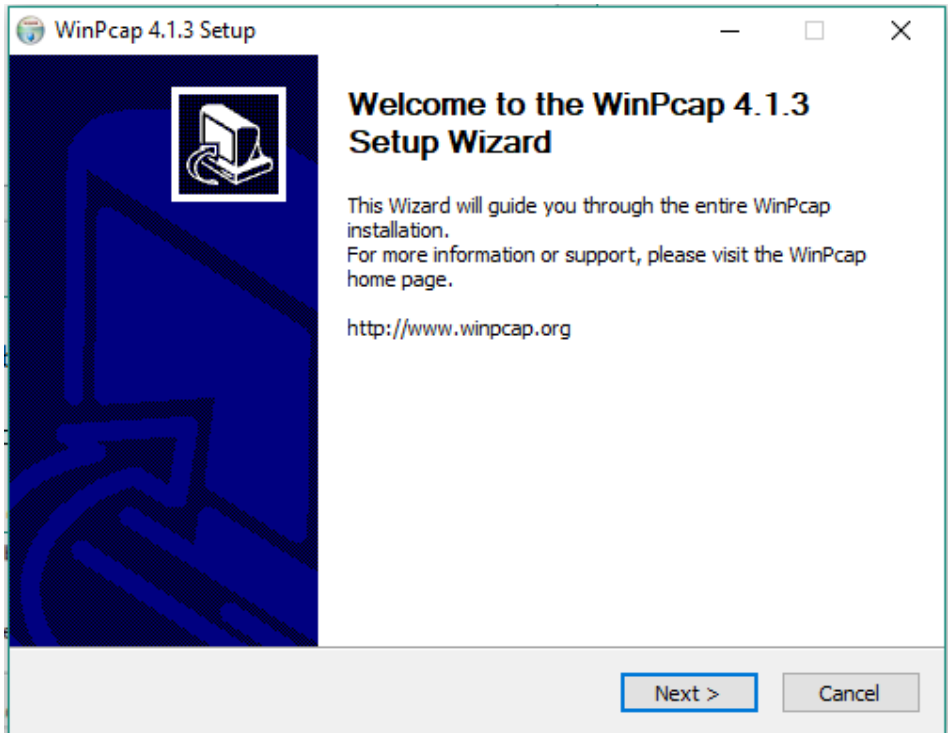
11. If you would like to capture USB traffic, install USBPcap as well.



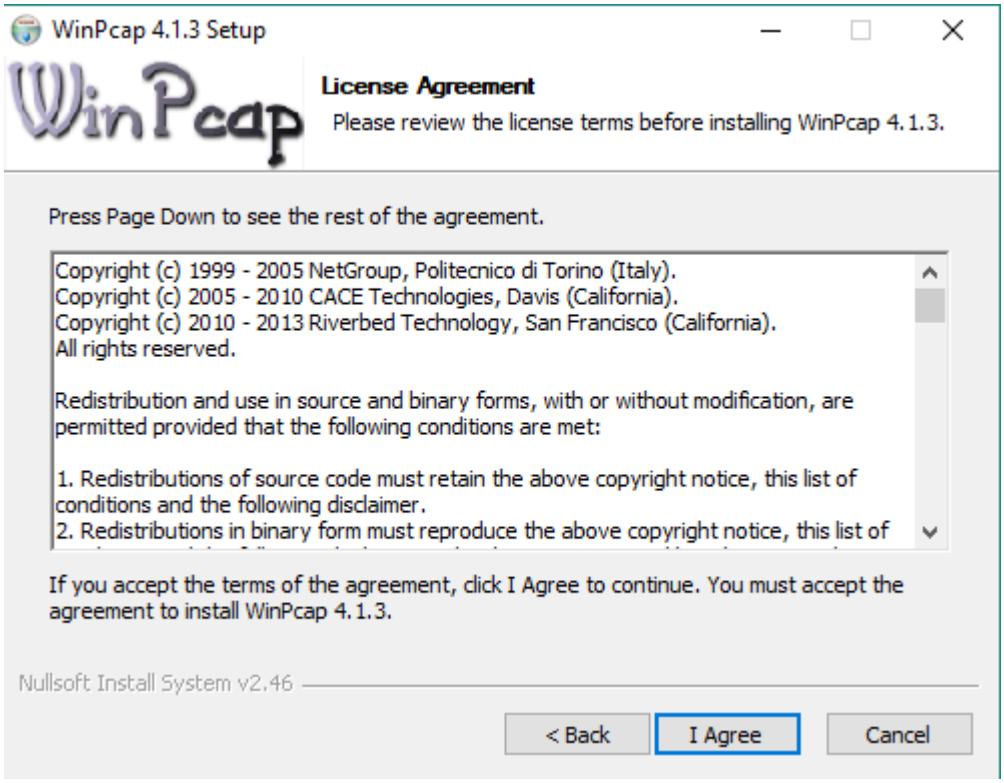
12. Select Next to start the Setup Wizard.



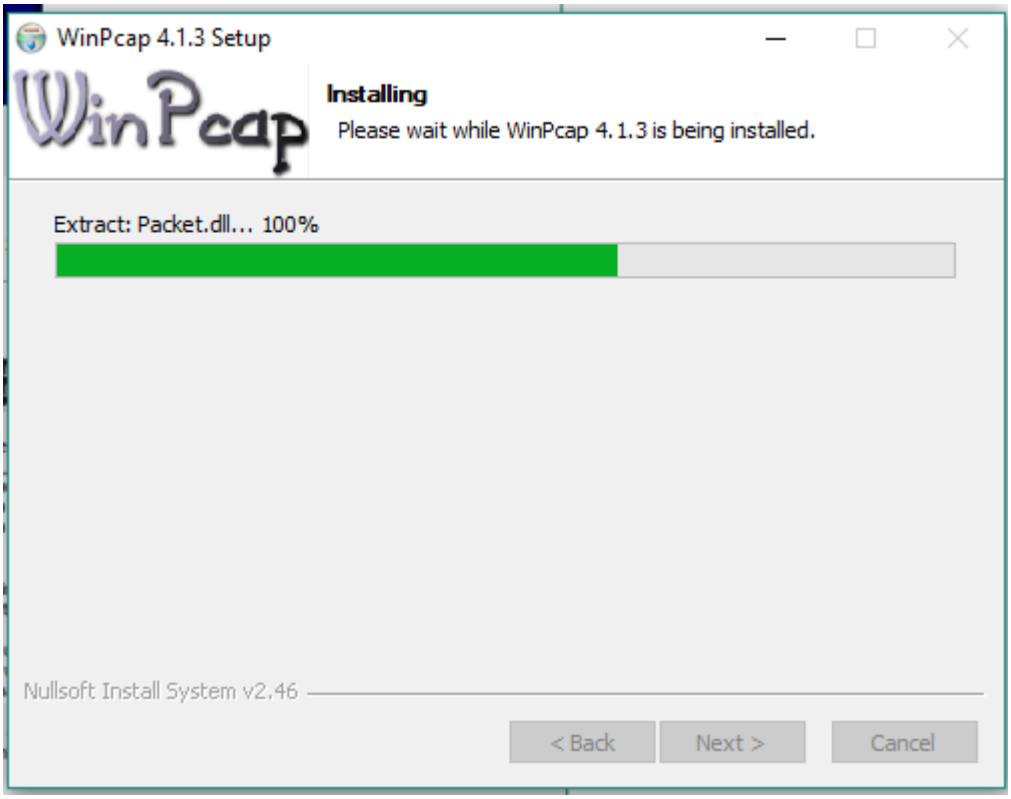
13. Select Install to proceed with the installation the requisite software WinPCap. Please note that WinPcap is a mandatory software to ensure Wireshark Packet Analyzer works properly.



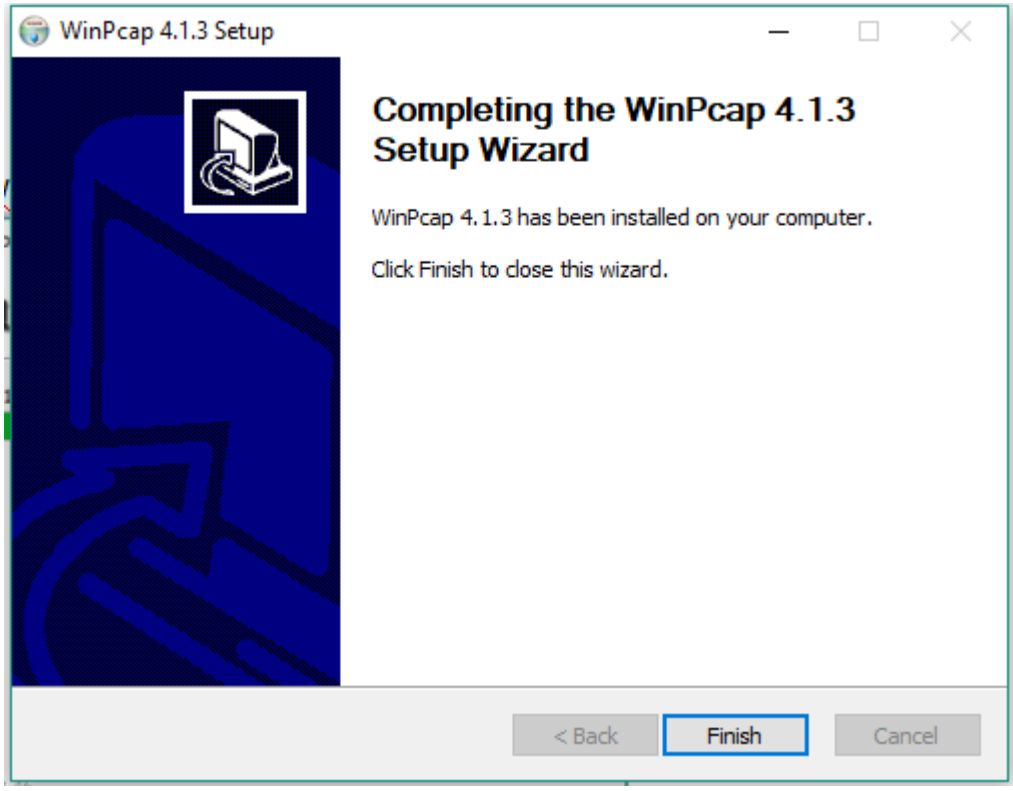
14. Review the license agreement. If you agree, select I Agree to continue.



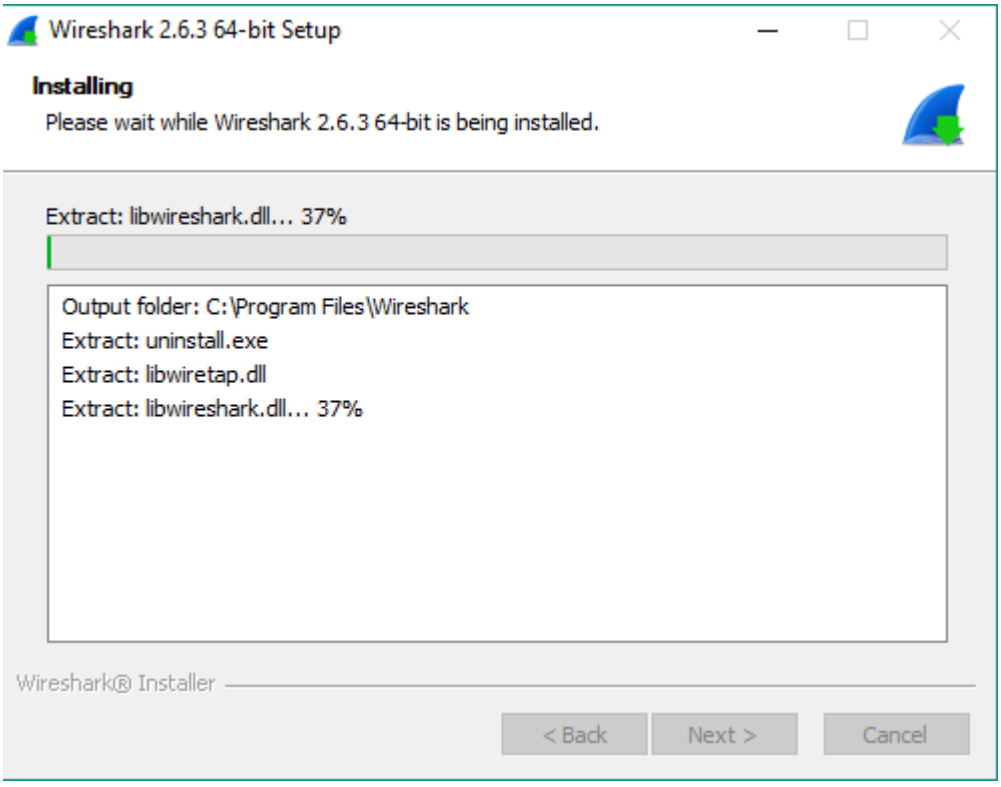
15. Installation of WinPcap should start automatically one you agreed and selected next.



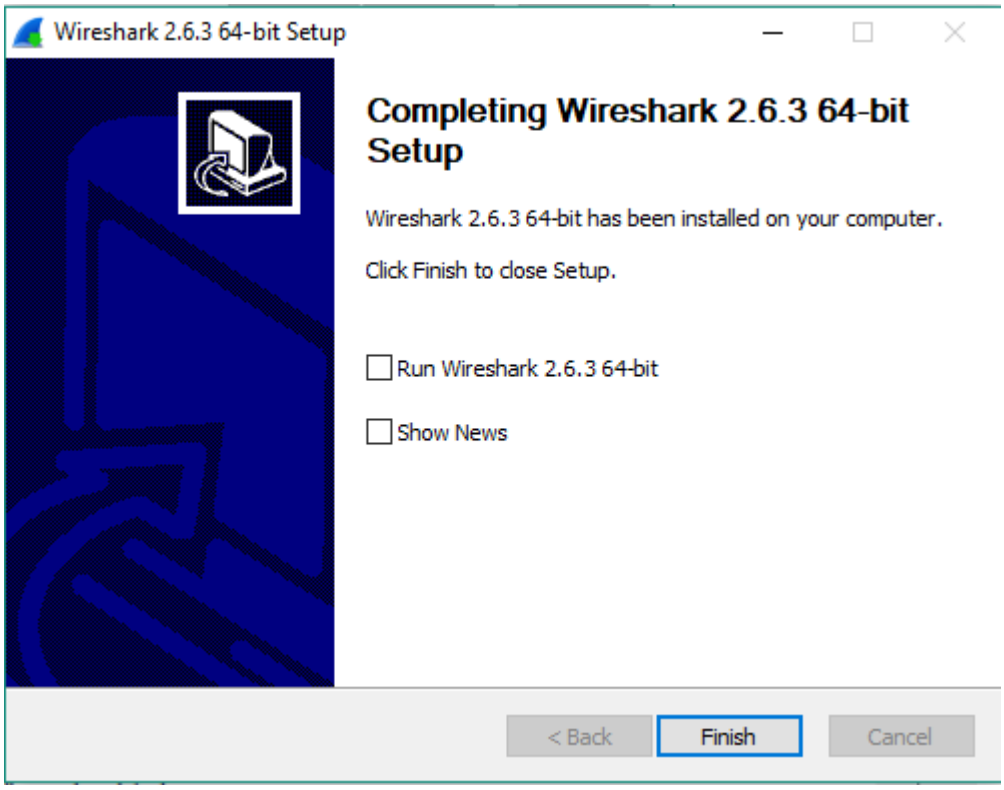
16. Select Finish to complete the installation of WinPcap.



17. Select Next to continue with the installation of Wireshark.



18. Select Finish to complete the installation of Wireshark. Once installed, you can open the Wireshark and start monitoring network traffic.



EXPERIMENT NO. 5

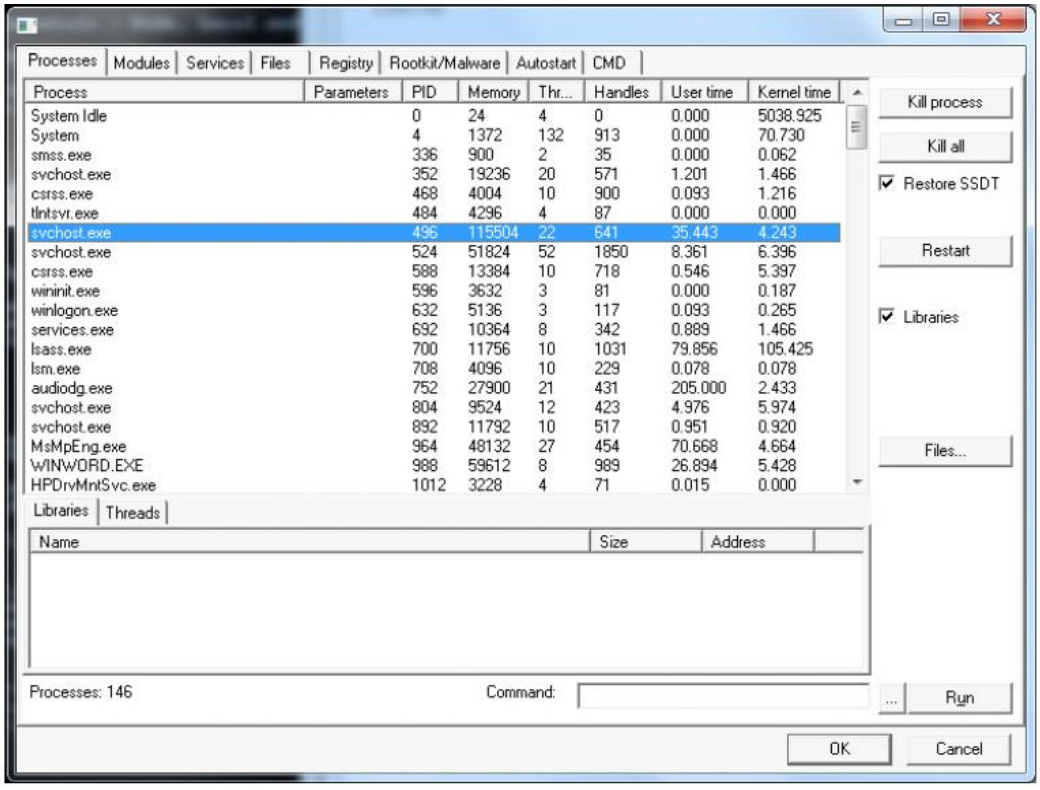
AIM: Installation of rootkits and study about the variety of options.

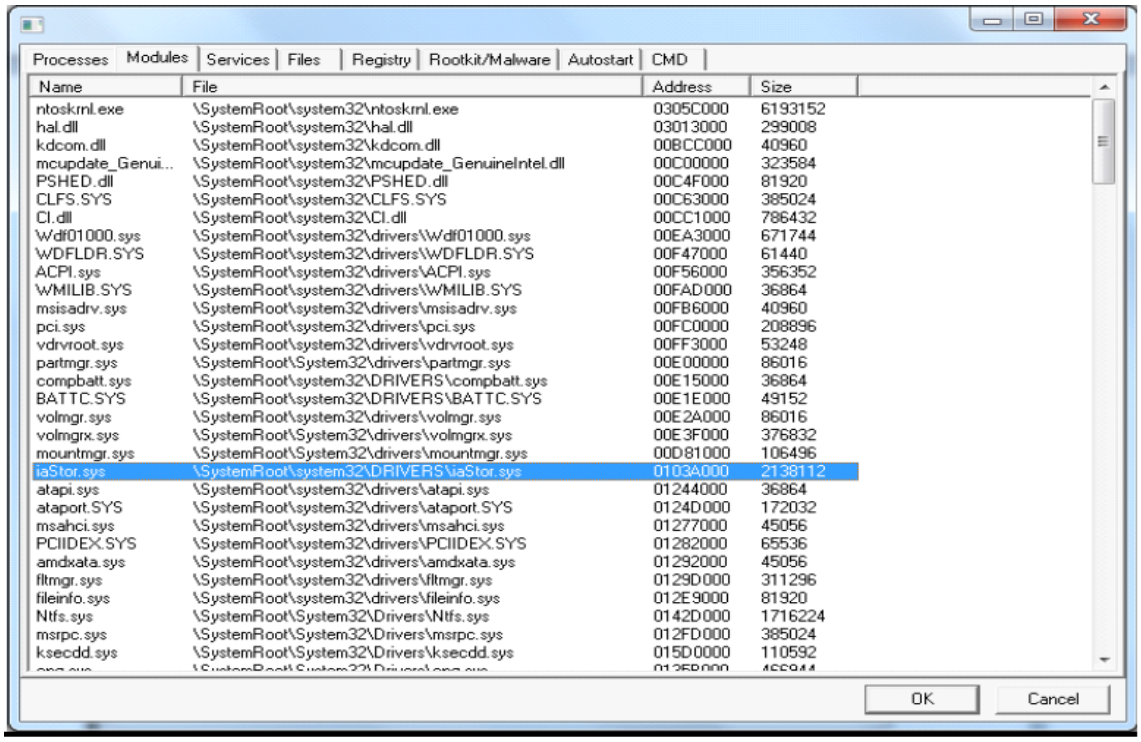
PROGRAM:

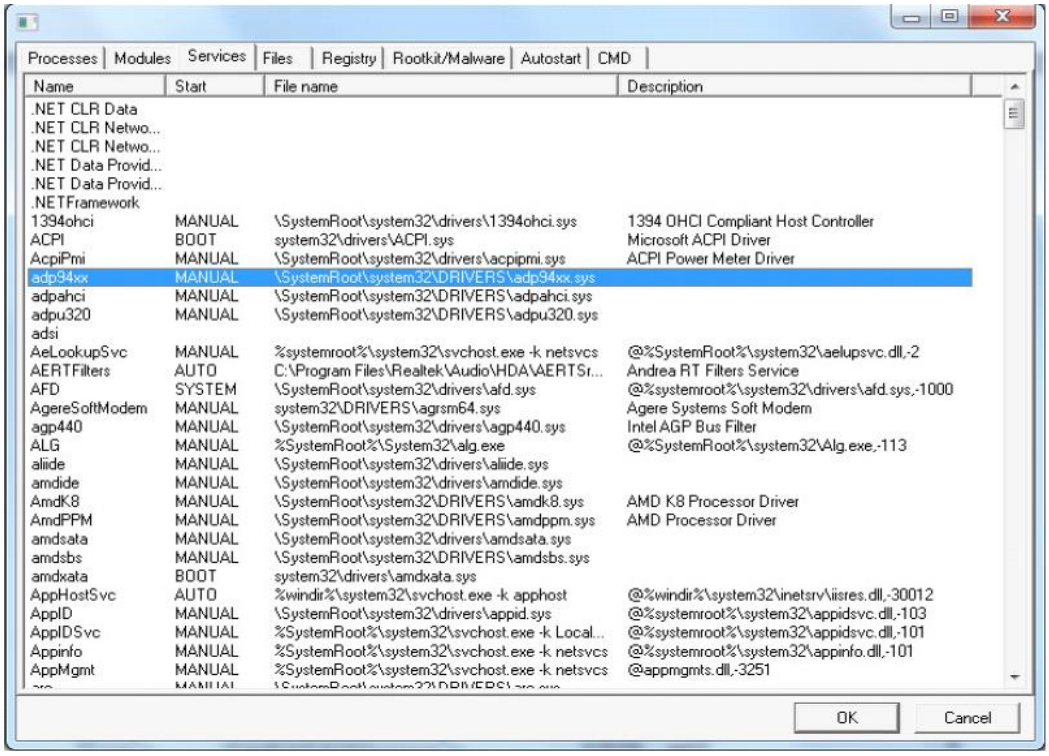
Breaking the term rootkit into the two component words, root and kit, is a useful way to define it. Root is a UNIX/Linux term that's the equivalent of Administrator in Windows. The word kit denotes programs that allow someone to obtain root/admin-level access to the computer by executing the programs in the kit — all of which is done without end-user consent or knowledge. A rootkit is a type of malicious software that is activated each time your system boots up. Rootkits are difficult to detect because they are activated before your system's Operating System has completely booted up. A rootkit often allows the installation of hidden files, processes, hidden user accounts, and more in the systems OS. Rootkits are able to intercept data from terminals, network connections, and the keyboard. Rootkits have two primary functions: remote command/control (back door) and software eavesdropping. Rootkits allow someone, legitimate or otherwise, to administratively control a computer. This means executing files, accessing logs, monitoring user activity, and even changing the computer's configuration. Therefore, in the strictest sense, even versions of VNC are rootkits. This surprises most people, as they consider rootkits to be solely malware, but in of themselves they aren't malicious at all. The presence of a rootkit on a network was first documented in the early 1990s. At that time, Sun and Linux operating systems were the primary targets for a hacker looking to install a rootkit. Today, rootkits are available for a number of operating systems, including Windows, and are increasingly difficult to detect on any network.

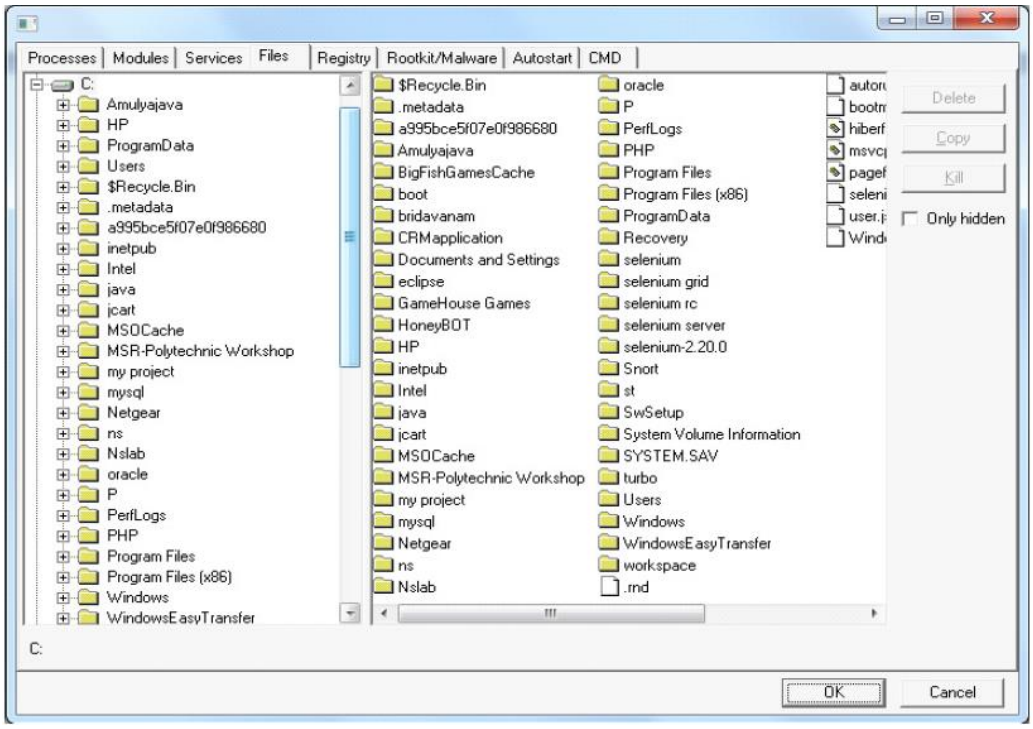
PROCEDURE:
STEP-1: Download Rootkit Tool from GMER website www.gmer.net.
STEP-2: This displays the Processes, Modules, Services, Files, Registry, RootKit / Malwares, Auto start, CMD of local host.
STEP-3: Select Processes menu and kill any unwanted process if any.
STEP-4: Modules menu displays the various system files like .sys, .dll
STEP-5: Services menu displays the complete services running with Autostart, Enable,
Disable, System, Boot.
STEP-6: Files menu displays full files on Hard-Disk volumes.
STEP-7: Registry displays Hkey_Current_user and Hkey_Local_Machine.
STEP-8: Rootkits / Malwares scans the local drives selected.
STEP-9: Autostart displays the registry base Autostart applications.
STEP-10:CMD allows the user to interact with command line utilities or Registry

CODE OUTPUT:


Processes
Modules
Services
Files
Registry
Rootkit/Malware
Autostart
CMD
Process
Parameters
PID
Memory
Thr...
Handles
User time
Kernel time
System Idle 0 24 4 0 0.000 5038.925
System 4 1372 132 913 0.000 70.730
smss.exe 336 900 2 35 0.000 0.062
svchost.exe 352 19236 20 571 1.201 1.466
csrss.exe 468 4004 10 900 0.093 1.216
tlntsvr.exe 484 4296 4 87 0.000 0.000
svchost.exe 496 115504 22 641 35.443 4.243
svchost.exe 524 51824 52 1850 8.361 6.396
csrss.exe 588 13384 10 718 0.546 5.397
wininit.exe 596 3632 3 81 0.000 0.187
winlogon.exe 632 5136 3 117 0.093 0.265
services.exe 692 10364 8 342 0.889 1.466
lsass.exe 700 11756 10 1031 79.856 105.425
lsm.exe 708 4096 10 229 0.078 0.078
audiodg.exe 752 27900 21 431 205.000 2.433
svchost.exe 804 9524 12 423 4.976 5.974
svchost.exe 892 11792 10 517 0.951 0.920
MsMpEng.exe 964 48132 27 454 70.668 4.664
WINWORD.EXE 988 59612 8 989 26.894 5.428
HPDrvMntSvc.exe 1012 3228 4 71 0.015 0.000
Kill process
Kill all
Restore SSDT
Restart
Libraries
Files...
Libraries
Threads
Name
Size
Address
Processes: 146
Command:
Run
OK
Cancel


Processes
Modules
Services
Files
Registry
Rootkit/Malware
Autostart
CMD
Name
File
Address
Size
ntoskrnl.exe \SystemRoot\system32\ntoskrnl.exe 0305C000 6193152
hal.dll \SystemRoot\system32\hal.dll 03013000 299008
kdcom.dll \SystemRoot\system32\kdcom.dll 00BCC000 40960
mcupdate_Genui... \SystemRoot\system32\mcupdate_GenuineIntel.dll 00C00000 323584
PSHED.dll \SystemRoot\system32\PSHED.dll 00C4F000 81920
CLFS.SYS \SystemRoot\system32\CLFS.SYS 00C63000 385024
CI.dll \SystemRoot\system32\CI.dll 00CC1000 786432
Wdf01000.sys \SystemRoot\system32\drivers\Wdf01000.sys 00EA3000 671744
WDFLDR.SYS \SystemRoot\system32\drivers\WDFLDR.SYS 00F47000 61440
ACPI.sys \SystemRoot\system32\drivers\ACPI.sys 00F56000 356352
WMILIB.SYS \SystemRoot\system32\drivers\WMILIB.SYS 00FAD000 36864
msisadrv.sys \SystemRoot\system32\drivers\msisadrv.sys 00FB6000 40960
pci.sys \SystemRoot\system32\drivers\pci.sys 00FC0000 208896
vdrvroot.sys \SystemRoot\system32\drivers\vdrvroot.sys 00FF3000 53248
partmgr.sys \SystemRoot\System32\drivers\partmgr.sys 00E00000 86016
compbatt.sys \SystemRoot\system32\DRIVERS\compbatt.sys 00E15000 36864
BATTC.SYS \SystemRoot\system32\DRIVERS\BATTC.SYS 00E1E000 49152
volmgr.sys \SystemRoot\system32\drivers\volmgr.sys 00E2A000 86016
volmgrx.sys \SystemRoot\System32\drivers\volmgrx.sys 00E3F000 376832
mountmgr.sys \SystemRoot\System32\drivers\mountmgr.sys 00D81000 106496
iaStor.sys \SystemRoot\system32\DRIVERS\iaStor.sys 0103A000 2138112
atapi.sys \SystemRoot\system32\drivers\atapi.sys 01244000 36864
ataport.SYS \SystemRoot\system32\drivers\ataport.SYS 0124D000 172032
msahci.sys \SystemRoot\system32\drivers\msahci.sys 01277000 45056
PCIIDEX.SYS \SystemRoot\system32\drivers\PCIIDEX.SYS 01282000 65536
amdxata.sys \SystemRoot\system32\drivers\amdxata.sys 01292000 45056
fltmgr.sys \SystemRoot\system32\drivers\fltmgr.sys 0129D000 311296
fileinfo.sys \SystemRoot\system32\drivers\fileinfo.sys 012E9000 81920
Ntfs.sys \SystemRoot\System32\Drivers\Ntfs.sys 0142D000 1716224
msrpc.sys \SystemRoot\System32\Drivers\msrpc.sys 012FD000 385024
ksecdd.sys \SystemRoot\System32\Drivers\ksecdd.sys 015D0000 110592
OK
Cancel


Processes
Modules
Services
Files
Registry
Rootkit/Malware
Autostart
CMD
Name
Start
File name
Description
.NET CLR Data
.NET CLR Netwo...
.NET CLR Netwo...
.NET Data Provid...
.NET Data Provid...
.NETFramework
1394ohci MANUAL \SystemRoot\system32\drivers\1394ohci.sys 1394 OHCI Compliant Host Controller
ACPI BOOT system32\drivers\ACPI.sys Microsoft ACPI Driver
AcpiPmi MANUAL \SystemRoot\system32\drivers\acpipmi.sys ACPI Power Meter Driver
adp94xx MANUAL \SystemRoot\system32\DRIVERS\adp94xx.sys
adpahci MANUAL \SystemRoot\system32\DRIVERS\adpahci.sys
adpu320 MANUAL \SystemRoot\system32\DRIVERS\adpu320.sys
adsi
AeLookupSvc MANUAL %systemroot%\system32\svchost.exe -k netsvcs @%SystemRoot%\system32\aelupsvc.dll,-2
AERTFilters AUTO C:\Program Files\Realtek\Audio\HDA\AERTSr... Andrea RT Filters Service
AFD SYSTEM \SystemRoot\system32\drivers\afd.sys @%systemroot%\system32\drivers\afd.sys,-1000
AgereSoftModem MANUAL system32\DRIVERS\agrsm64.sys Agere Systems Soft Modem
agp440 MANUAL \SystemRoot\system32\drivers\agp440.sys Intel AGP Bus Filter
ALG MANUAL %SystemRoot%\System32\alg.exe @%SystemRoot%\system32\Alg.exe,-113
aliide MANUAL \SystemRoot\system32\drivers\aliide.sys
amdide MANUAL \SystemRoot\system32\drivers\amdide.sys
AmdK8 MANUAL \SystemRoot\system32\DRIVERS\amdk8.sys AMD K8 Processor Driver
AmdPPM MANUAL \SystemRoot\system32\DRIVERS\amdppm.sys AMD Processor Driver
amdsata MANUAL \SystemRoot\system32\drivers\amdsata.sys
amdsbs MANUAL \SystemRoot\system32\DRIVERS\amdsbs.sys
amdxata BOOT system32\drivers\amdxata.sys
AppHostSvc AUTO %windir%\system32\svchost.exe -k apphost @%windir%\system32\inetsrv\iisres.dll,-30012
AppID MANUAL \SystemRoot\system32\drivers\appid.sys @%systemroot%\system32\appidsvc.dll,-103
AppIDSvc MANUAL %SystemRoot%\system32\svchost.exe -k Local... @%systemroot%\system32\appidsvc.dll,-101
Appinfo MANUAL %SystemRoot%\system32\svchost.exe -k netsvcs @%systemroot%\system32\appinfo.dll,-101
AppMgmt MANUAL %SystemRoot%\system32\svchost.exe -k netsvcs @appmgmts.dll,-3251
OK
Cancel


Processes
Modules
Services
Files
Registry
Rootkit/Malware
Autostart
CMD
C:
Amulyajava
HP
ProgramData
Users
$Recycle.Bin
.metadata
a995bce5f07e0f986680
inetpub
Intel
java
jcart
MSOCache
MSR-Polytechnic Workshop
my project
mysql
Netgear
ns
Nslab
oracle
P
PerfLogs
PHP
Program Files
Program Files (x86)
Windows
WindowsEasyTransfer
$Recycle.Bin
.metadata
a995bce5f07e0f986680
Amulyajava
BigFishGamesCache
boot
bridavanam
CRMapplication
Documents and Settings
eclipse
GameHouse Games
HoneyBOT
HP
inetpub
Intel
java
jcart
MSOCache
MSR-Polytechnic Workshop
my project
mysql
Netgear
ns
Nslab
oracle
P
PerfLogs
PHP
Program Files
Program Files (x86)
ProgramData
Recovery
selenium
selenium grid
selenium rc
selenium server
selenium-2.20.0
Snort
st
SwSetup
System Volume Information
SYSTEM.SAV
turbo
Users
Windows
WindowsEasyTransfer
workspace
.rnd
Delete
Copy
Kill
Only hidden
C:
OK
Cancel

EXPERIMENT NO. 6

AIM: Perform an Experiment to Sniff Traffic using ARP Poisoning.

PROGRAM:

We have used **Better CAP** to perform ARP poisoning in LAN environment using VMware workstation in which we have installed **Kali** Linux and **Ettercap** tool to sniff the local traffic in LAN.

For this exercise, you would need the following tools −

- VMware workstation
- Kali Linux or Linux Operating system
- Ettercap Tool
- LAN connection

Note − This attack is possible in wired and wireless networks. You can perform this attack in local LAN.

CODE OUTPUT:

Step 1 − Install the VMware workstation and install the Kali Linux operating system.

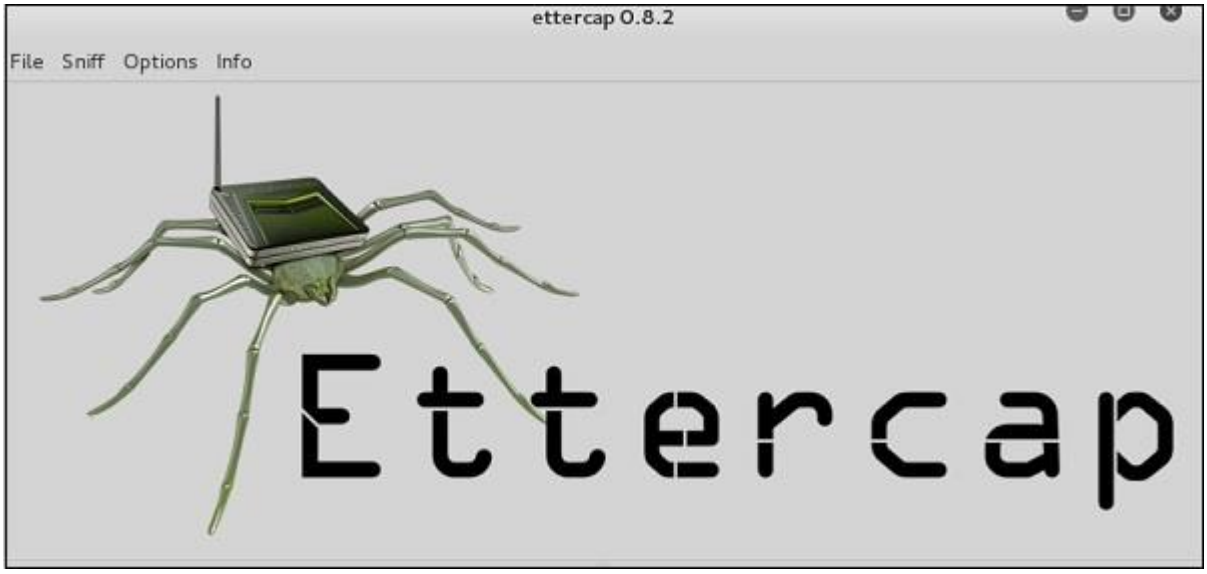
Step 2 − Login into the Kali Linux using username pass "root, toor".

Step 3 − Make sure you are connected to local LAN and check the IP address by typing the command **ifconfig** in the terminal.

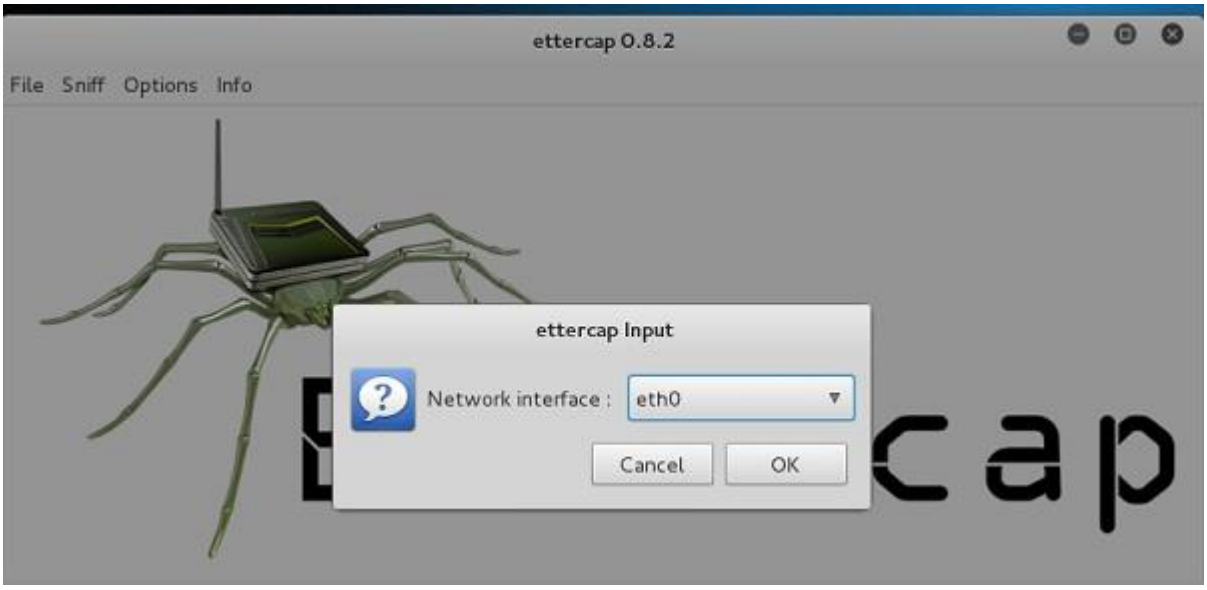
```
root@kali:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:cf:f8:e7
          inet addr:192.168.121.128  Bcast:192.168.121.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fecf:f8e7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:70 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4963 (4.8 KiB)  TX bytes:8868 (8.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:16 errors:0 dropped:0 overruns:0 frame:0
          TX packets:16 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:960 (960.0 B)  TX bytes:960 (960.0 B)
```

Step 4 − Open up the terminal and type "Ettercap –G" to start the graphical version of Ettercap.

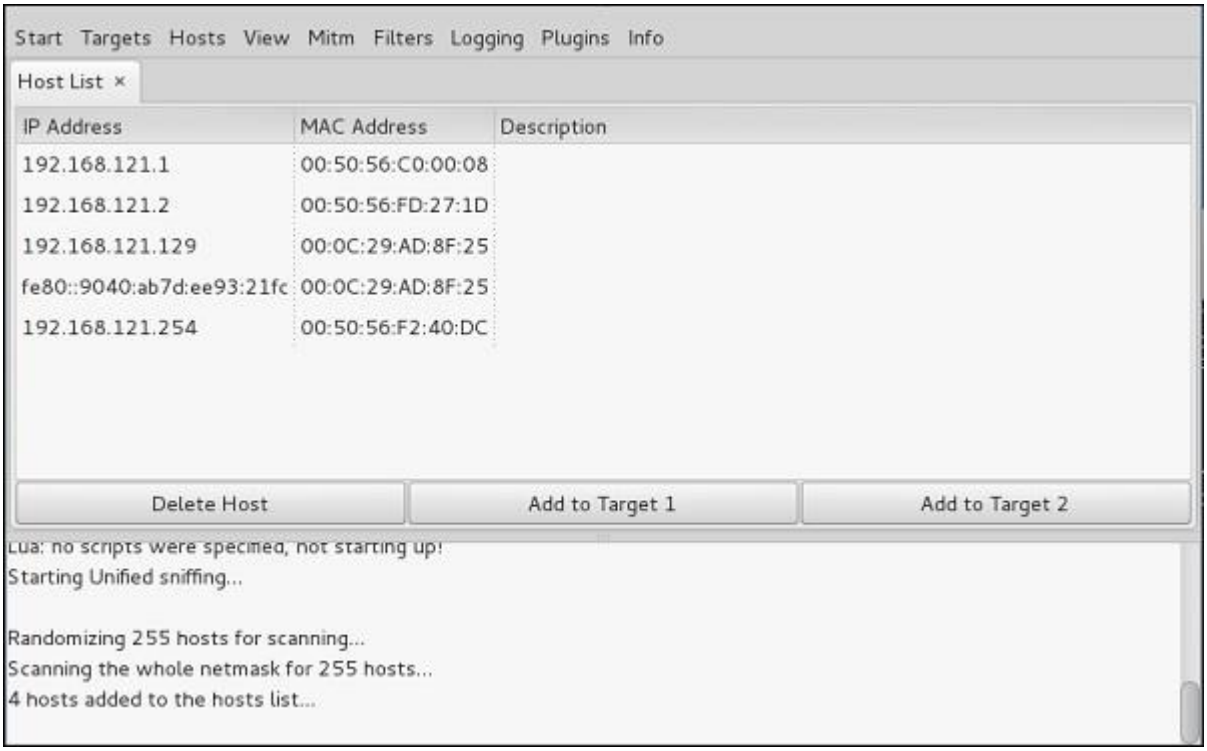


Step 5 − Now click the tab “sniff” in the menu bar and select “unified sniffing” and click OK to select the interface. We are going to use “eth0” which means Ethernet connection.



Step 6 − Now click the “hosts” tab in the menu bar and click “scan for hosts”. It will start scanning the whole network for the alive hosts.

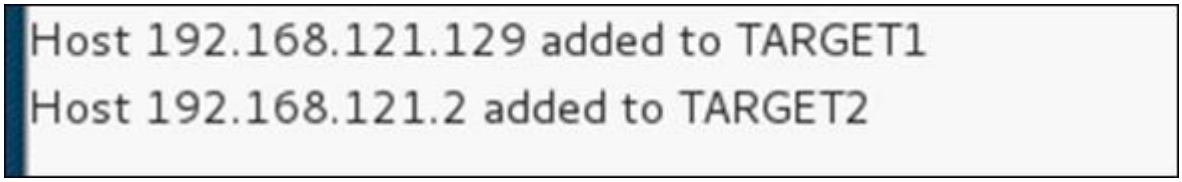
Step 7 − Next, click the “hosts” tab and select “hosts list” to see the number of hosts available in the network. This list also includes the default gateway address. We have to be careful when we select the targets.



Step 8 − Now we have to choose the targets. In MITM, our target is the host machine, and the route will be the router address to forward the traffic. In an MITM attack, the attacker intercepts the network and sniffs the packets. So, we will add the victim as "target 1" and the router address as "target 2."

In VMware environment, the default gateway will always end with "2" because "1" is assigned to the physical machine.

Step 9 − In this scenario, our target is "192.168.121.129" and the router is "192.168.121.2". So we will add target 1 as **victim IP** and target 2 as **router IP**.



Step 10 − Now click on "MITM" and click "ARP poisoning". Thereafter, check the option "Sniff remote connections" and click OK.

Step 11 − Click "start" and select "start sniffing". This will start ARP poisoning in the network which means we have enabled our network card in "promiscuous mode" and now the local traffic can be sniffed.

Note − We have allowed only HTTP sniffing with Ettercap, so don't expect HTTPS packets to be sniffed with this process.

Step 12 − Now it's time to see the results; if our victim logged into some websites. You can see the results in the toolbar of Ettercap.

This is how sniffing works. You must have understood how easy it is to get the HTTP credentials just by enabling ARP poisoning.

ARP Poisoning has the potential to cause huge losses in company environments. This is the place where ethical hackers are appointed to secure the networks.

Like ARP poisoning, there are other attacks such as MAC flooding, MAC spoofing, DNS poisoning, ICMP poisoning, etc. that can cause significant loss to a network.

EXPERIMENT NO. 7

AIM: Demonstrate intrusion detection system using any tool (snort or any other s/w).

PROGRAM:

Intrusion detection is a set of techniques and methods that are used to detectsuspicious activity both at the network and host level. Intrusion detection systems fall into two basic categories:

- Signature-based intrusion detection systems
- Anomaly detection systems.

Intruders have signatures, like computer viruses, that can be detected using software.You try to find data packets that contain any known intrusion-related signatures or anomaliesrelated to Internet protocols. Based upon a set of signatures and rules, the detection system isable to find and log suspicious activity and generate alerts.

Anomaly-based intrusion detection usually depends on packet anomalies present in protocol header parts. In some cases these methods produce better results compared tosignature-based IDS. Usually an intrusion detection system captures data from the networkand applies its rules to that data or detects anomalies in it. Snort is primarily a rule-basedIDS, however input plug-ins are present to detect anomalies in protocol headers.

SNORT TOOL:

Snort is based on libpcap (for library packet capture), a tool that is widely used inTCP/IPtraffic sniffers and analyzers. Through protocol analysis and content searching andmatching, Snort detects attack methods, including denial of service, buffer overflow, CGIattacks, stealthport scans, and SMB probes. When suspicious behavior is detected, Snortsends a real-time alert to syslog, a separate 'alerts' file, or to apop-up window.

Snort is currently the most popular free network intrusion detection software. The advantages of Snort are numerous. According to the snort web site, “It can perform protocol analysis, content searching/matching, and can be used to detect a variety of attacks andprobes, such as buffer overflow, stealth port scans, CGI attacks, SMB probes, OSfingerprinting attempts, and much more” (Caswell).

One of the advantages of Snort is its ease of configuration. Rules are very flexible, easily written, and easily inserted into the rule base. If a new exploit or attack is found a rulefor the attack can be added to the rule base in a matter of seconds. Another advantage ofsnort is that it allows for raw packet data analysis.

STEP-1: Sniffer mode_ snort –v _ Print out the TCP/IP packets header on the screen.

STEP-2: Snort –vd _ Show the TCP/IP ICMP header with application data in transit.

STEP-3: Packet Logger mode _ snort –dev –l c:\log [create this directory in the C drive]and snort will automatically know to go into packet logger mode, it collects everypacket it sees and places it in log directory.

STEP-4: snort –dev –l c:\log –h ipaddress/24 _ This rule tells snort that you want to printout

the data link and TCP/IP headers as well as application data into the log directory.

STEP-5: snort –l c:\log –b _ this binary mode logs everything into a single file.

STEP-6: Network Intrusion Detection System mode _ snort –d c:\log –h ipaddress/24 –csnort.conf _ This is a configuration file that applies rule to each packet to decide it an action based upon the rule type in the file.

STEP-7: snort –d –h ip address/24 –l c:\log –c snort.conf _ This will configure snort to runin its most basic NIDS form, logging packets that trigger rules specifies in the snort.conf.

STEP-8: Download SNORT from snort.org. Install snort with or without database support.

STEP-9: Select all the components and Click Next. Install and Close.

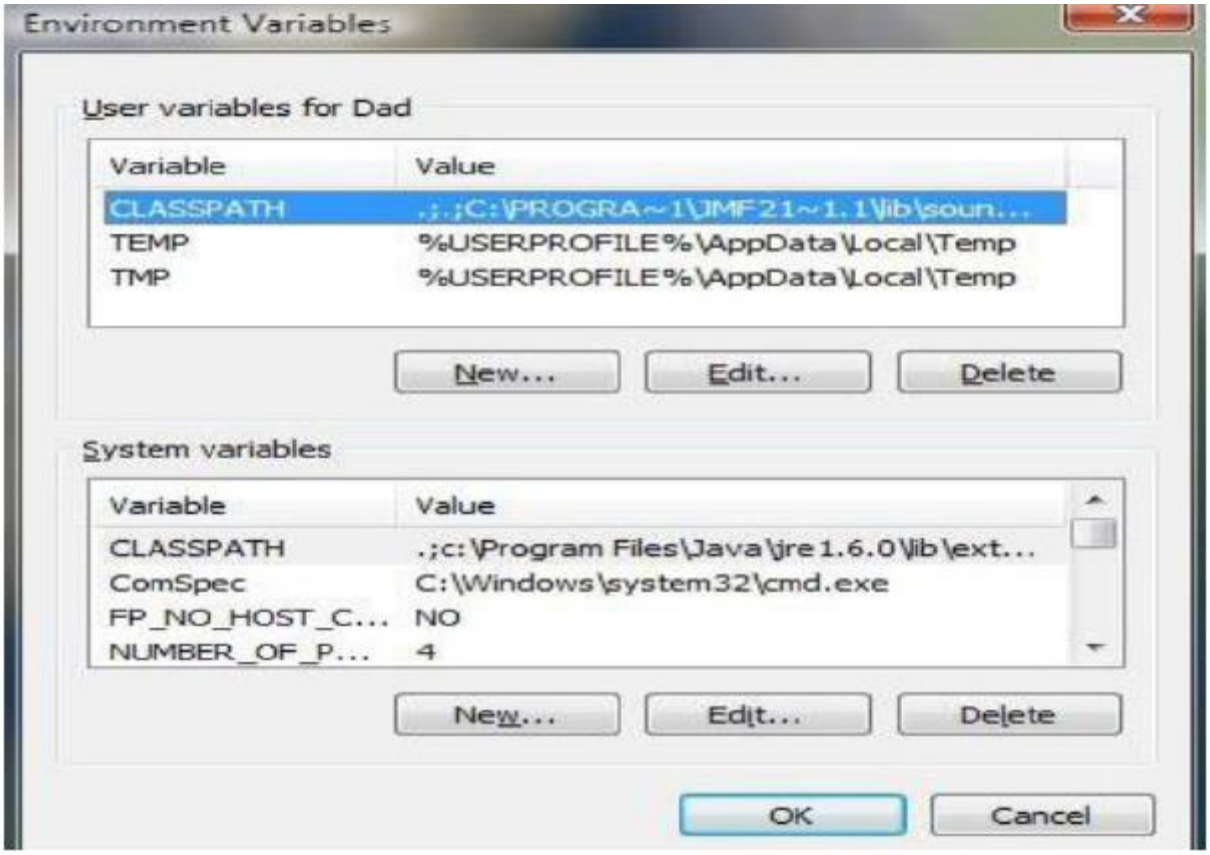
STEP-10: Skip the WinPcap driver installation.

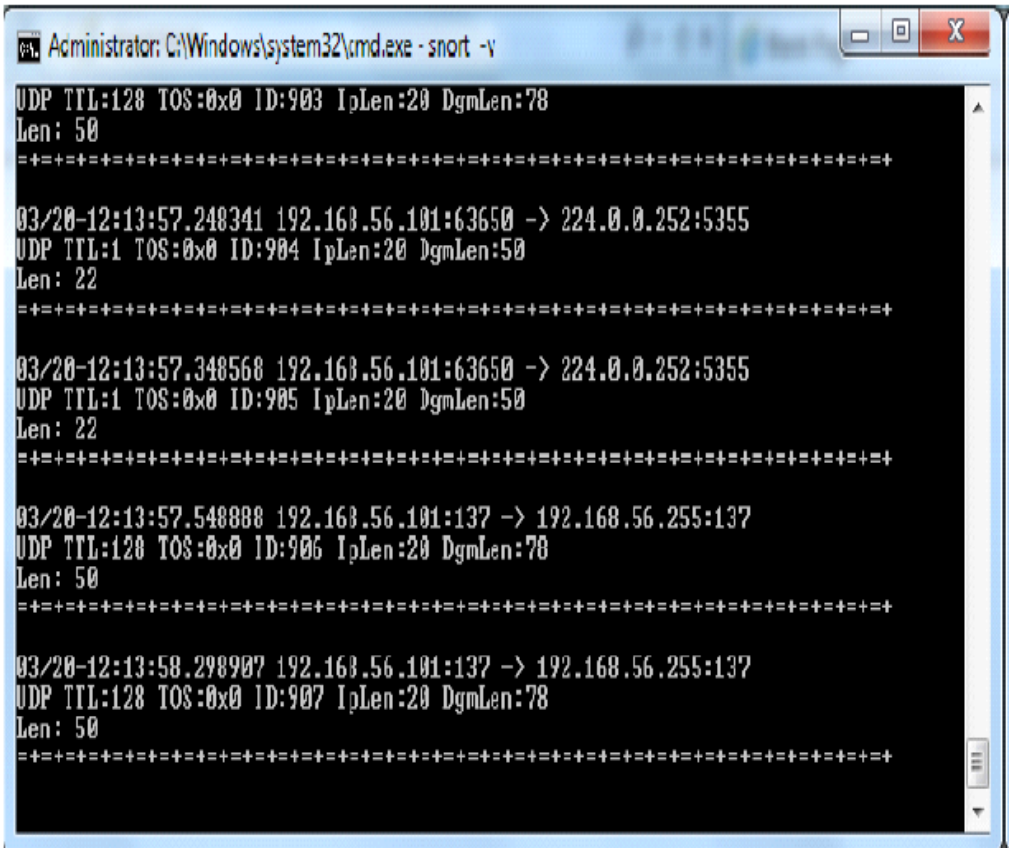
STEP-11: Add the path variable in windows environment variable by selecting new classpath.

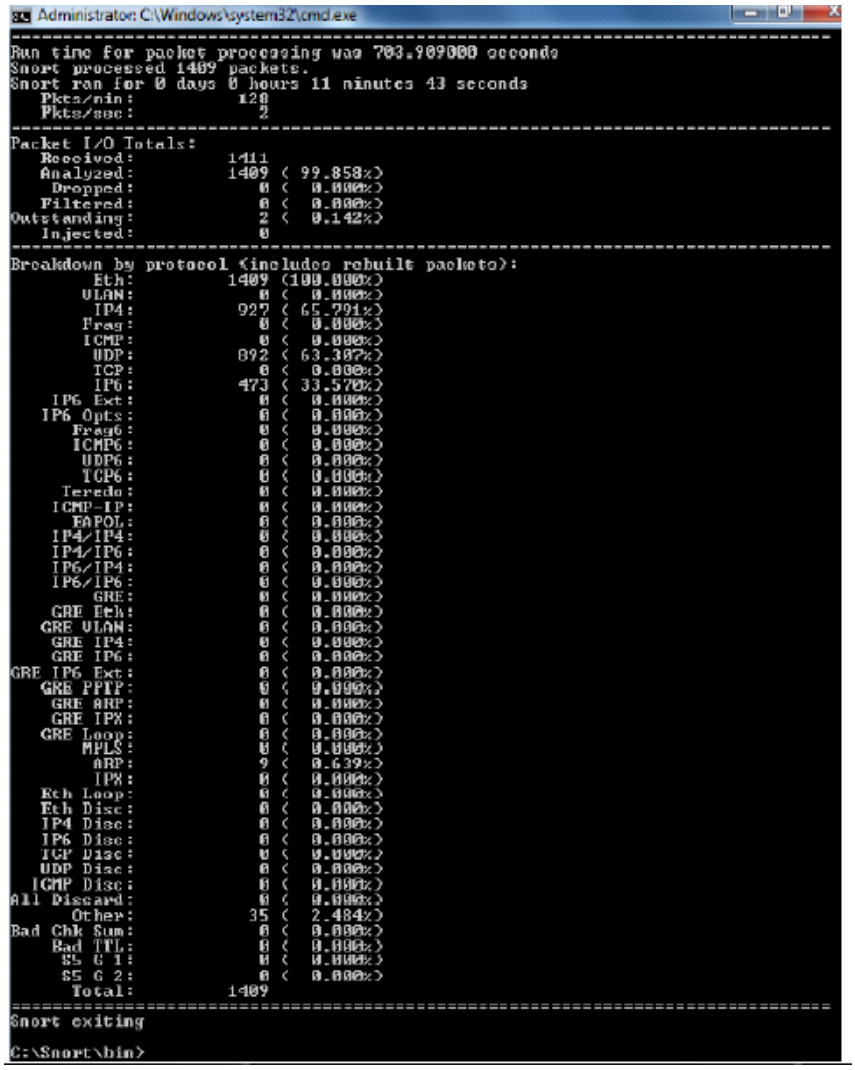
STEP-12: Create a path variable and point it at snort.exe variable name _ path and variablevalue _ c:\snort\bin.

STEP-13: Click OK button and then close all dialog boxes. Open command prompt and typethe following commands:

CODE OUTPUT:


Environment Variables
User variables for Dad
Variable
Value
CLASSPATH
.;.;C:\PROGRA~1\JMF21~1.1\lib\soun...
TEMP
%USERPROFILE%\AppData\Local\Temp
TMP
%USERPROFILE%\AppData\Local\Temp
New...
Edit...
Delete
System variables
Variable
Value
CLASSPATH
.;c:\Program Files\Java\jre1.6.0\lib\ext...
ComSpec
C:\Windows\system32\cmd.exe
FP_NO_HOST_C...
NO
NUMBER_OF_P...
4
New...
Edit...
Delete
OK
Cancel


Administrator: C:\Windows\system32\cmd.exe - snort -v
UDP TTL:128 TOS:0x0 ID:903 IpLen:20 DgmLen:78
Len: 50
03/20-12:13:57.248341 192.168.56.101:63650 -> 224.0.0.252:5355
UDP TTL:1 TOS:0x0 ID:904 IpLen:20 DgmLen:50
Len: 22
03/20-12:13:57.348568 192.168.56.101:63650 -> 224.0.0.252:5355
UDP TTL:1 TOS:0x0 ID:905 IpLen:20 DgmLen:50
Len: 22
03/20-12:13:57.548888 192.168.56.101:137 -> 192.168.56.255:137
UDP TTL:128 TOS:0x0 ID:906 IpLen:20 DgmLen:78
Len: 50
03/20-12:13:58.298907 192.168.56.101:137 -> 192.168.56.255:137
UDP TTL:128 TOS:0x0 ID:907 IpLen:20 DgmLen:78
Len: 50


Administrator: C:\Windows\system32\cmd.exe
Run time for packet processing was 703.909000 seconds
Snort processed 1409 packets.
Snort ran for 0 days 0 hours 11 minutes 43 seconds
Pkts/min: 128
Pkts/sec: 2
Packet I/O Totals:
Received: 1411
Analyzed: 1409 (99.858%)
Dropped: 0 (0.000%)
Filtered: 0 (0.000%)
Outstanding: 2 (0.142%)
Injected: 0
Breakdown by protocol (includes rebuilt packets):
Eth: 1409 (100.000%)
VLAN: 0 (0.000%)
IP4: 927 (65.791%)
Frag: 0 (0.000%)
ICMP: 0 (0.000%)
UDP: 892 (63.307%)
TCP: 0 (0.000%)
IP6: 473 (33.570%)
IP6 Ext: 0 (0.000%)
IP6 Opts: 0 (0.000%)
Frag6: 0 (0.000%)
ICMP6: 0 (0.000%)
UDP6: 0 (0.000%)
TCP6: 0 (0.000%)
Teredo: 0 (0.000%)
ICMP-IP: 0 (0.000%)
EAPOL: 0 (0.000%)
IP4/IP4: 0 (0.000%)
IP4/IP6: 0 (0.000%)
IP6/IP4: 0 (0.000%)
IP6/IP6: 0 (0.000%)
GRE: 0 (0.000%)
GRE Eth: 0 (0.000%)
GRE VLAN: 0 (0.000%)
GRE IP4: 0 (0.000%)
GRE IP6: 0 (0.000%)
GRE IP6 Ext: 0 (0.000%)
GRE PPTP: 0 (0.000%)
GRE ARP: 0 (0.000%)
GRE IPX: 0 (0.000%)
GRE Loop: 0 (0.000%)
MPLS: 0 (0.000%)
ARP: 9 (0.639%)
IPX: 0 (0.000%)
Eth Loop: 0 (0.000%)
Eth Disc: 0 (0.000%)
IP4 Disc: 0 (0.000%)
IP6 Disc: 0 (0.000%)
TCP Disc: 0 (0.000%)
UDP Disc: 0 (0.000%)
ICMP Disc: 0 (0.000%)
All Discard: 0 (0.000%)
Other: 35 (2.484%)
Bad Chk Sum: 0 (0.000%)
Bad TTL: 0 (0.000%)
S5 G 1: 0 (0.000%)
S5 G 2: 0 (0.000%)
Total: 1409
Snort exiting
C:\Snort\bin>

EXPERIMENT NO. 8

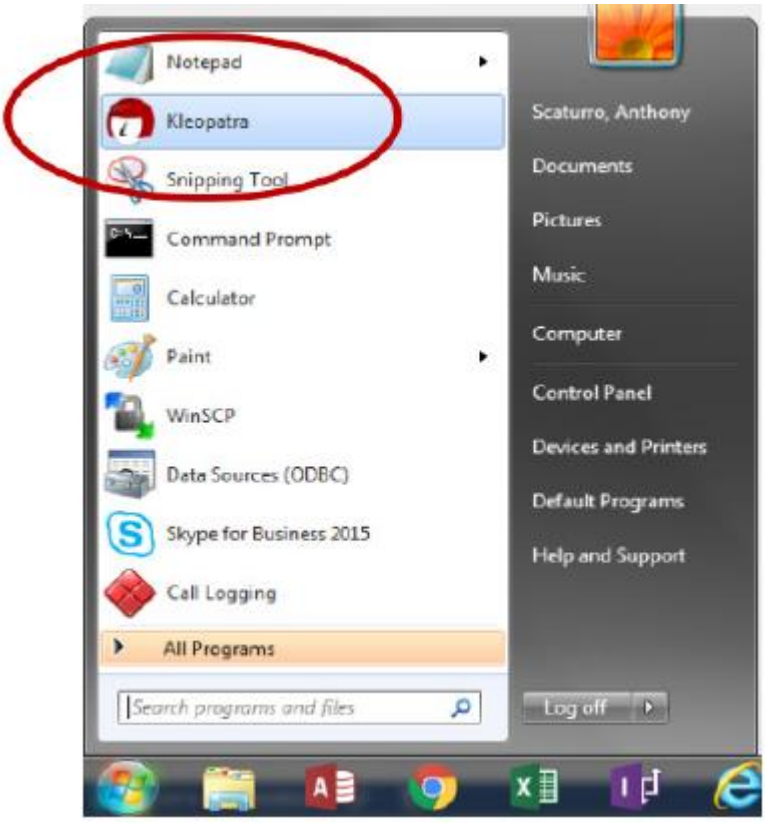
AIM: Demonstrate how to provide secure data storage, secure data transmissionand for creating digital signatures.

PROGRAM:

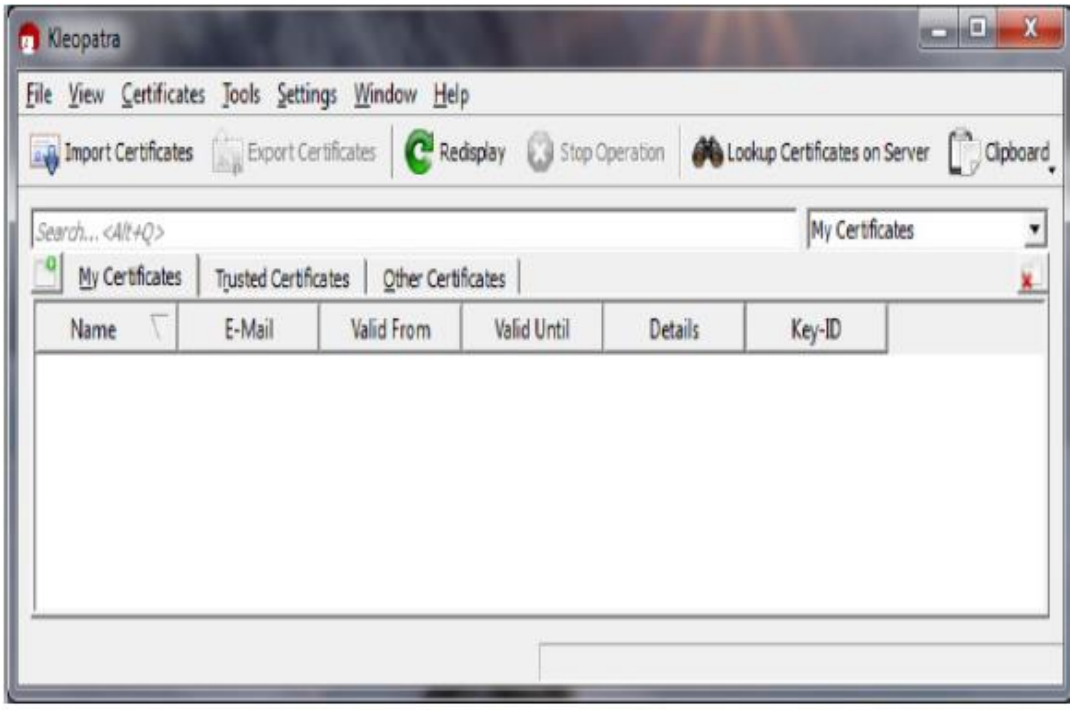
CREATING YOUR PUBLIC AND PRIVATE KEYS
GPG encryption and decryption is based upon the keys of the person who will be receiving the encrypted file or message. Any individual who wants to send the person an encrypted file or message must possess the recipient's public key certificate to encrypt the message. The recipient must have the associated private key, which is different than the public key, to be able to decrypt the file. The public and private key pair for an individual is usually generated by the individual on his or her computer using the installed GPG program, called "Kleopatra" and the following procedure:

CODE OUTPUT:

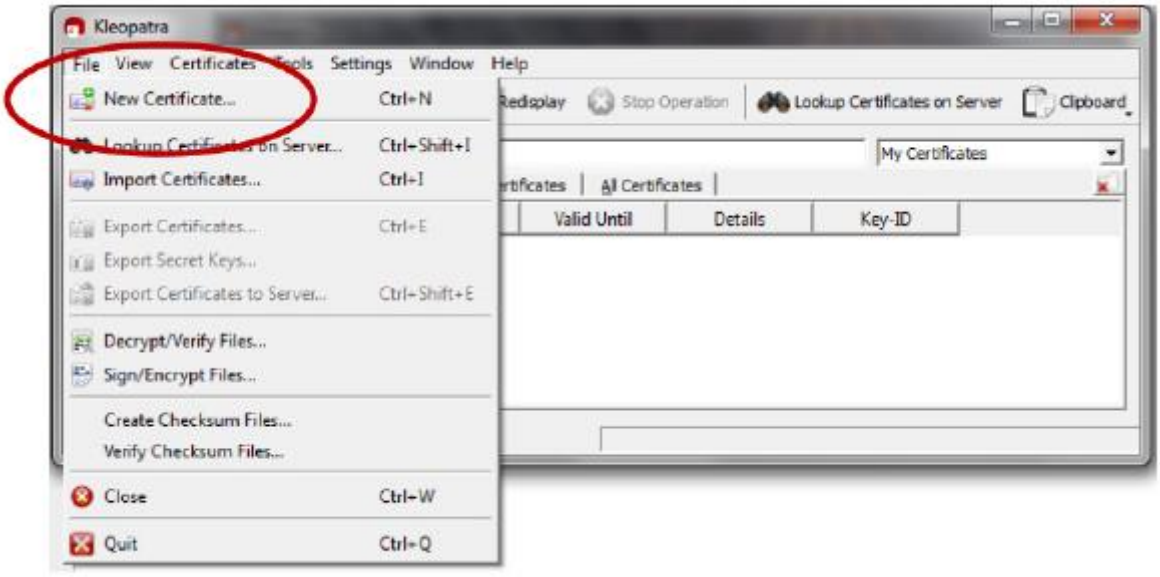
1. From your start bar, select the "Kleopatra" icon to start the Kleopatra certificate management software



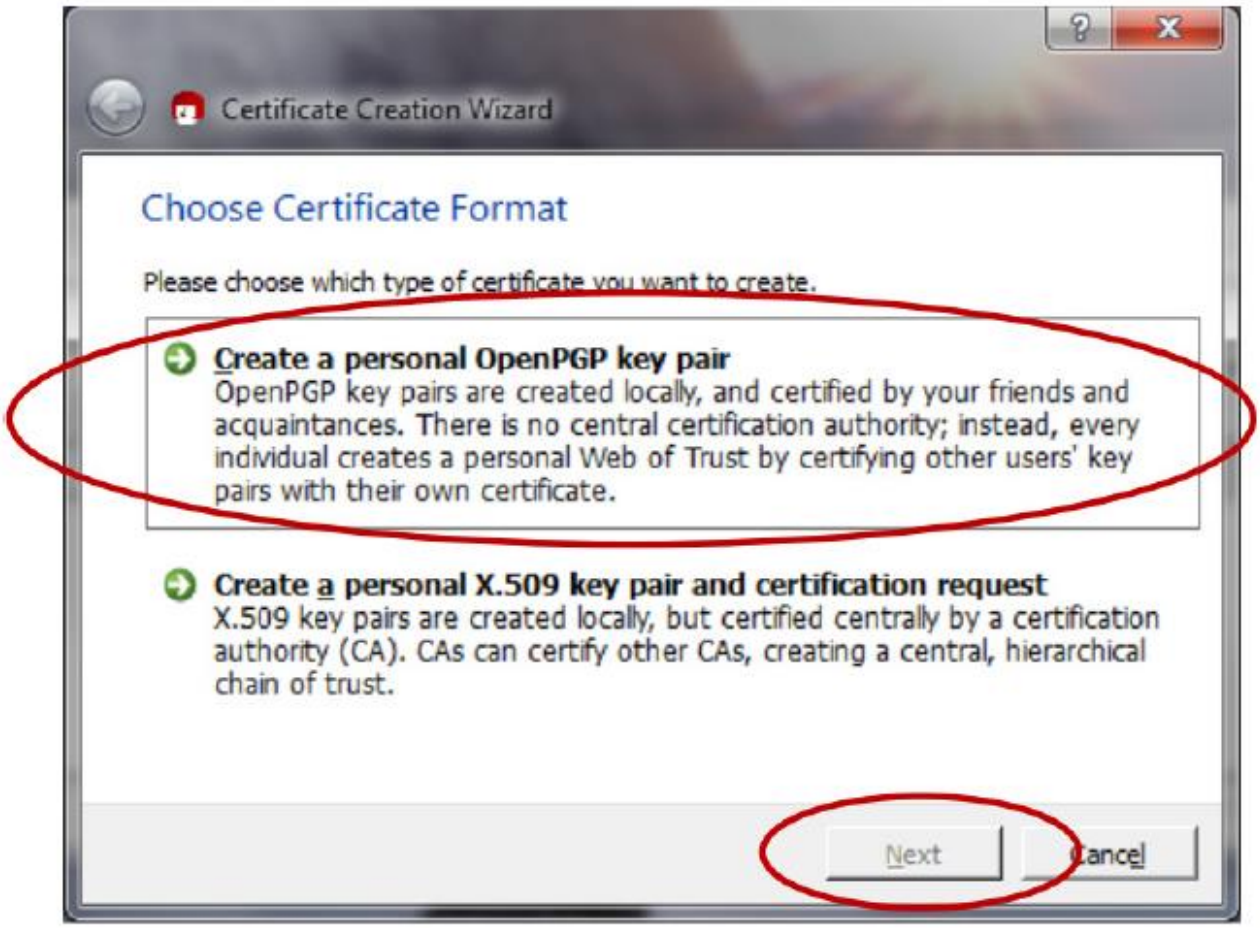
2. The following screen will be displayed



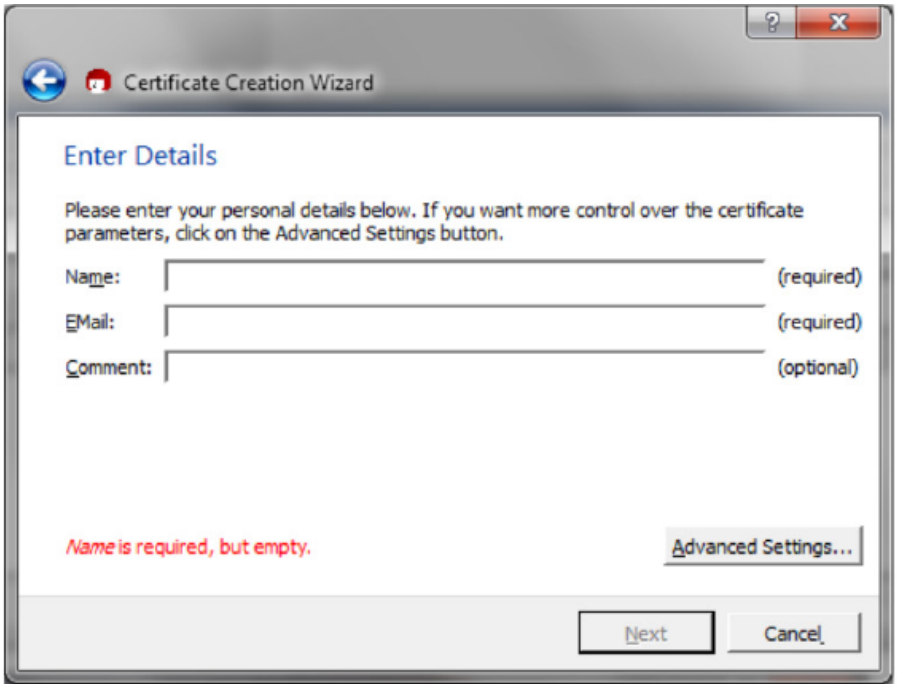
3. From the “File” dropdown, click on the “New Certificate” option



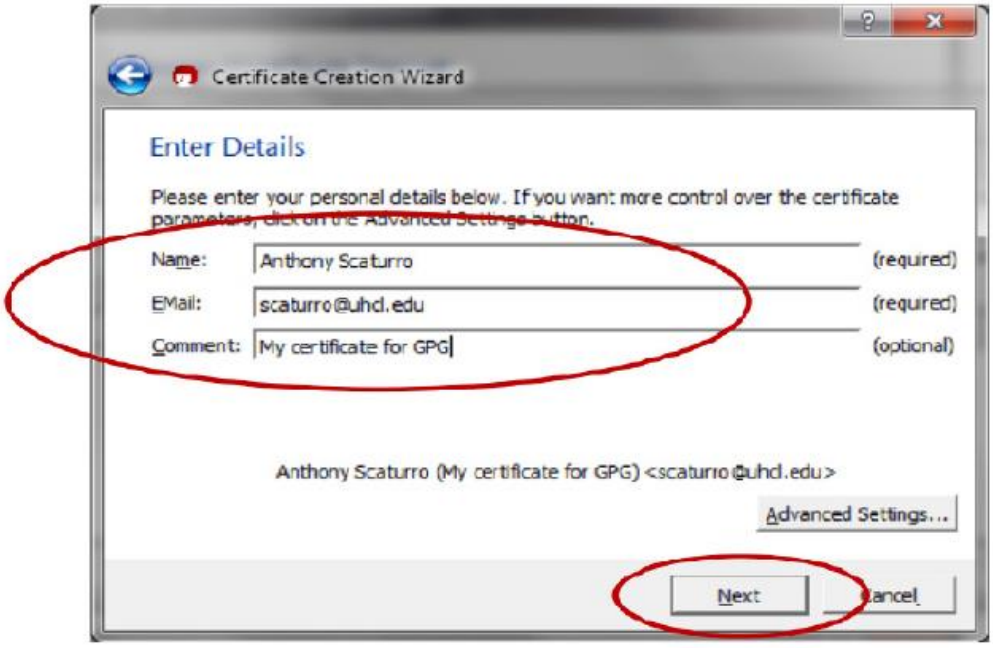
4. The following screen will be displayed. Click on “Create a personal OpenGPG key pair” and the “Next” button



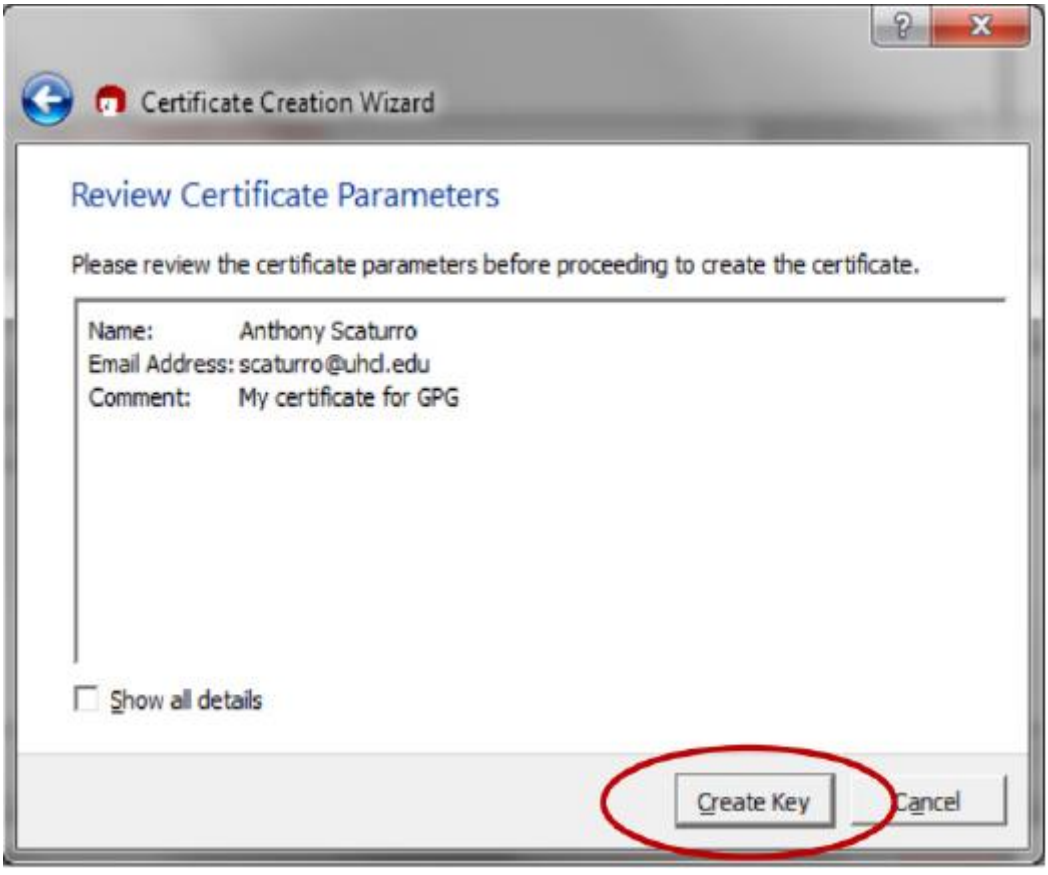
5. The Certificate Creation Wizard will start and display the following:



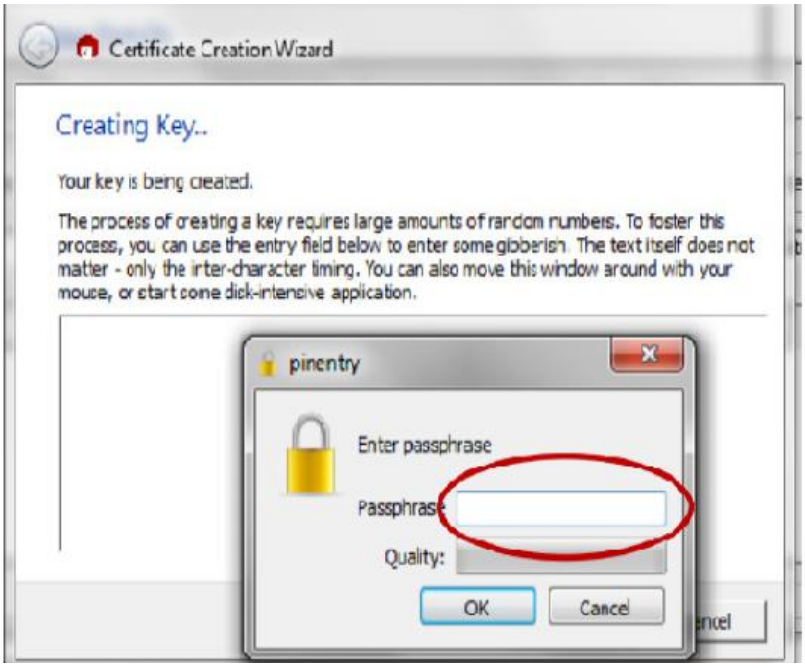
6. Enter your name and e-mail address. You may also enter an optional comment. Then,click the "Next" buttonReview your entered values. If OK, click the "Create Key" button



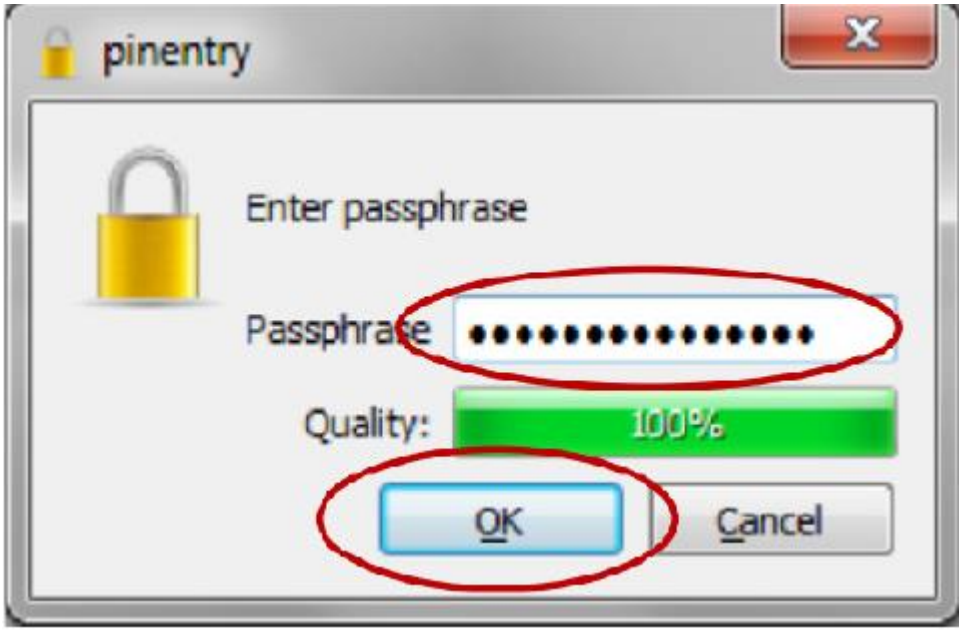
7. Review your entered values. If OK, click the "Create Key" button



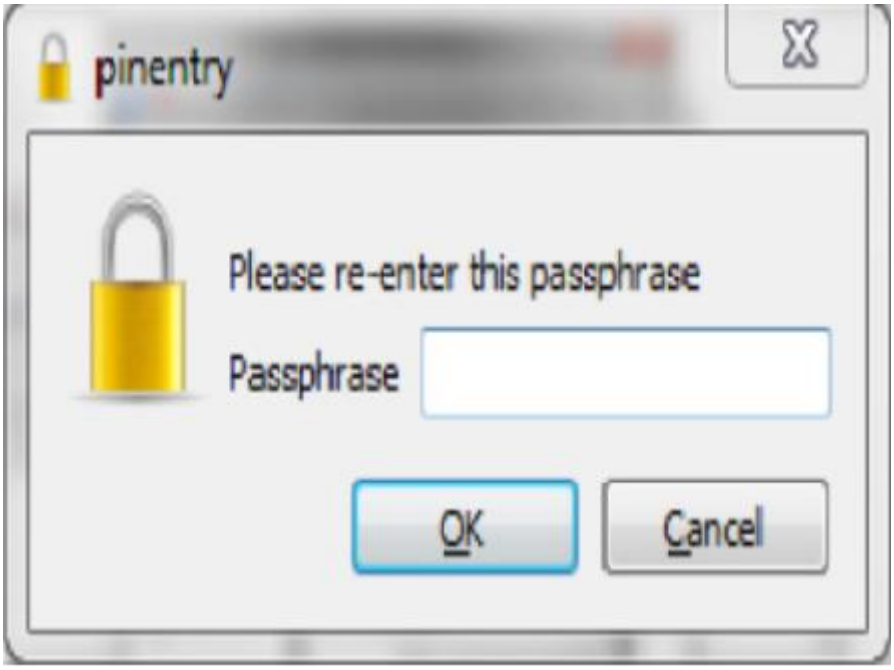
8. You will be asked to enter a passphrase



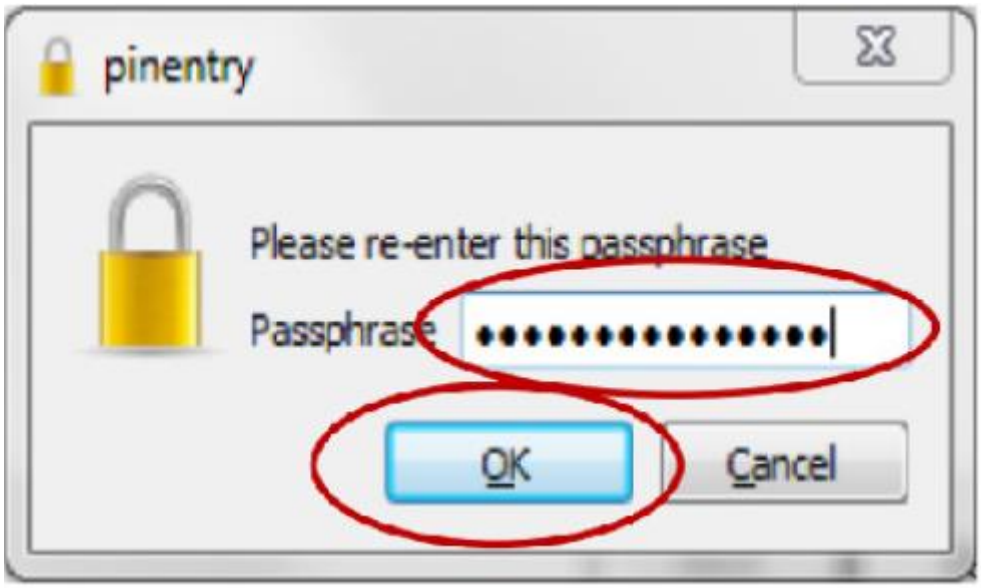
9. The passphrase should follow strong password standards. After you've entered your passphrase, click the "OK" button.



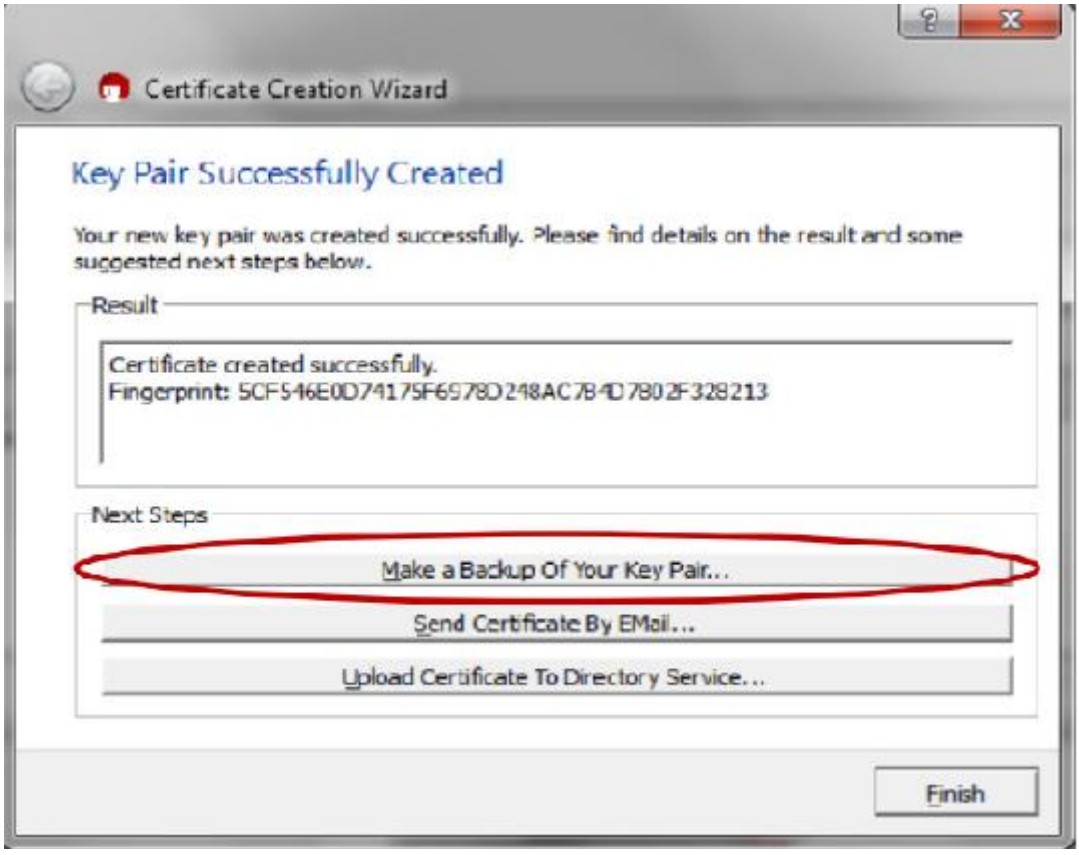
10. You will be asked to re-enter the passphrase



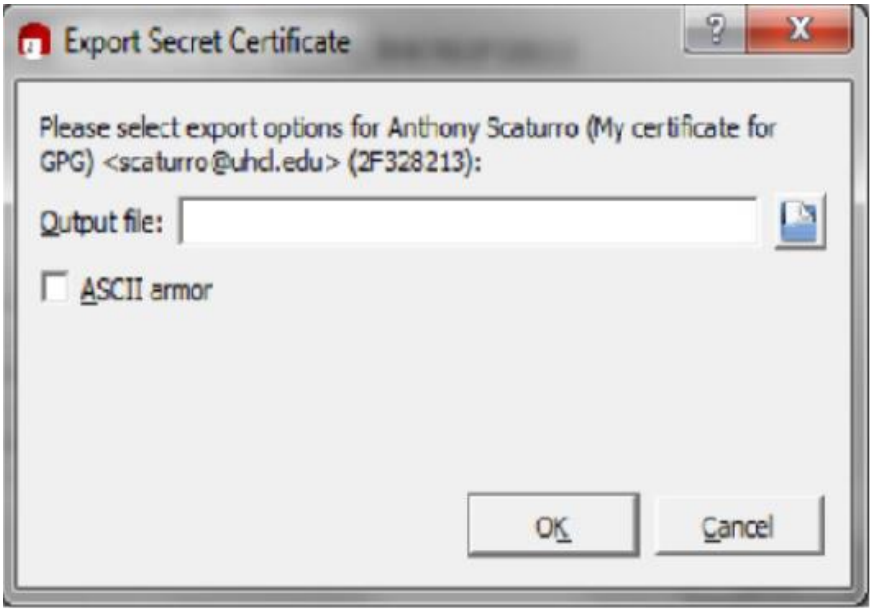
11. Re-enter the passphrase value. Then click the “OK” button. If the passphrases match,the certificate will be created.



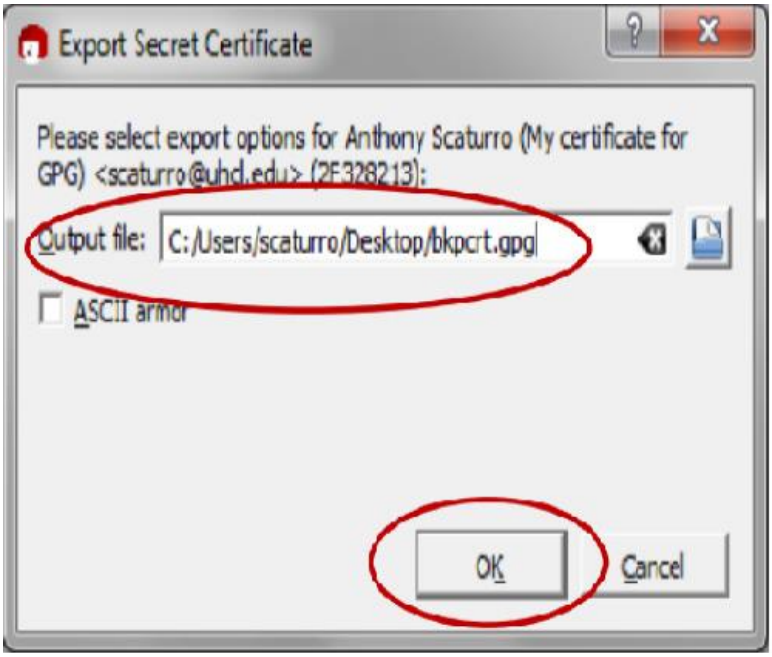
12. Once the certificate is created, the following screen will be displayed. You cansave abackup of your public and private keys by clicking the “Make a backup Of Your Key Pair” button. This backup can be used to copy certificates onto other authorized computers.



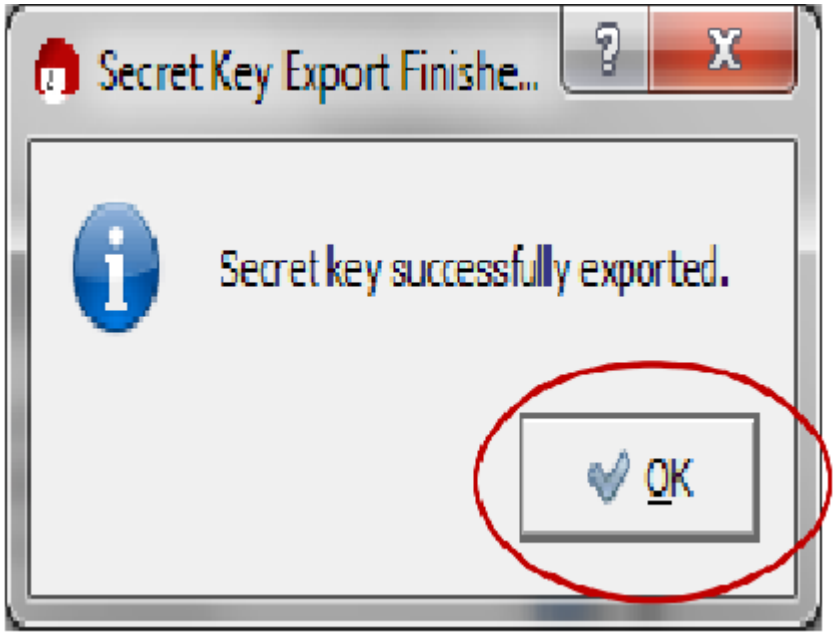
13. If you choose to backup your key pair, you will be presented with the following screen:



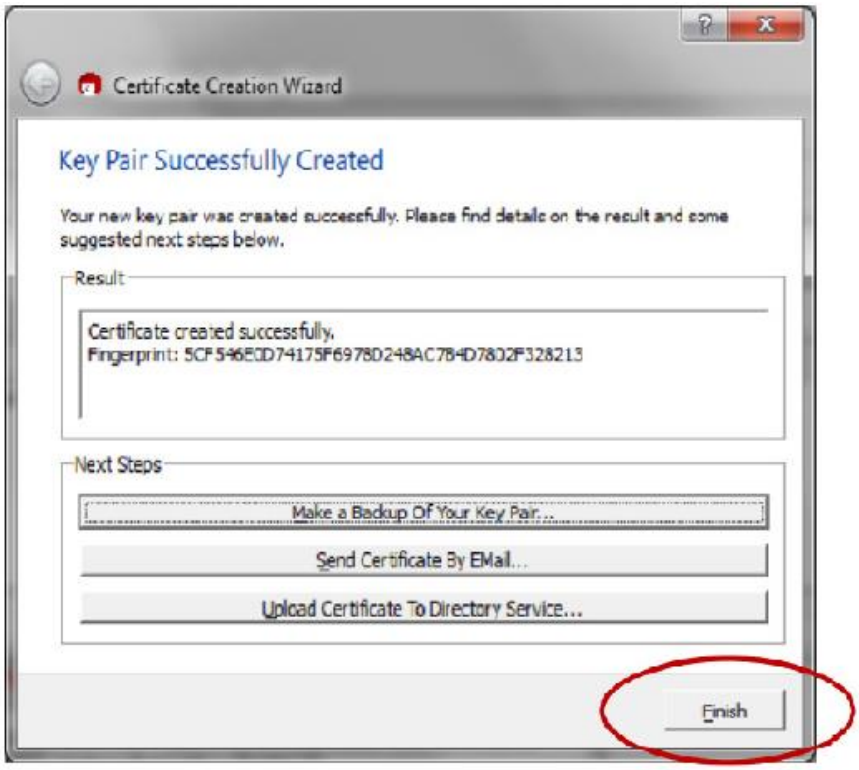
14. Specify the folder and name the file. Then click the “OK” button.



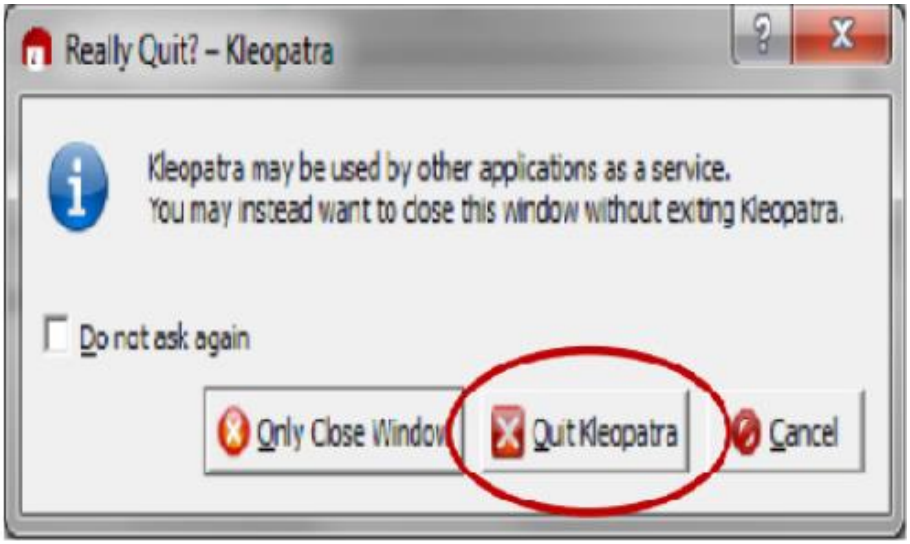
15. After the key is exported, the following will be displayed. Click the "OK" button.



16. You will be returned to the "Key Pair Successfully Created" screen. Click the "Finish" button.



17. Before the program closes, you will need to confirm that you want to close the program by clicking on the "Quit Kleopatra" button



RUBRICS EVALUATION

Performance Criteria	Scale 1 (0-25%)	Scale 2 (26-50%)	Scale 3 (51-75%)	Scale 4 (76-100%)	Score (Numerical)
Understandability Ability to analyse Problem and Identify solution	Unable to understand the problem.	Able to understand the problem partially and unable to identify the solution	Able to understand the problem completely but unable to identify the solution	Able to understand the problem completely and able to provide alternative solution too.	
Logic Ability to specify Conditions & control flow that are appropriate for the problem domain.	Program logic is incorrect	Program logic is on the right track but has several errors	Program logic is mostly correct, but may contain an occasional boundary error or redundant or contradictory condition.	Program logic is correct, with no known boundary errors, and no redundant or contradictory conditions.	
Debugging Ability to execute /debug	Unable to execute program	Unable to debug several errors.	Able to execute program with several warnings.	Able to execute program completely	
Correctness Ability to code formulae and algorithms that reliably produce correct answers or appropriate results.	Program does not produce correct answers or appropriate results for most inputs.	Program approaches correct answers or appropriate results for most inputs, but can contain miscalculations in some cases.	Program produces correct answers or appropriate results for most inputs.	Program produces correct answers or appropriate results for all inputs tested.	
Completeness Ability to demonstrate and deliver on time.	Unable to explain the code.and the code was overdue.	Unable to explain the code and the code submission was late.	Able to explain code and the program was delivered within the due date.	Able to explain code and the program was delivered on time.	
				TOTAL	

OUTCOMES OF LAB

After Completion of all the practical experiment students have achieved:

- The student should be able to implement the cipher techniques.
- Develop the various security algorithms.
- Use different open source tools for network security and analysis
- To learn & use the new tools and technologies used for designing some security principles.
- Analyse and resolve security issues in networks and computer systems to secure an IT infrastructure.

Computer Lab's Do's and Don'ts and Safety Rules

DO's

- Please switch off the Mobile/Cell phone before entering Lab.
- Check whether all peripheral are available at your desktop before proceeding for the session
- Arrange all the peripheral and seats before leaving the lab.
- Properly shutdown the system before leaving the lab.
- Keep the bag outside in the racks.
- Enter the lab on time and leave at proper time.
- Maintain the decorum of the lab.

DON'TS

- Don't mishandle the system.
- Don't leave the system on standing for long
- Don't bring any external material in the lab.
- Don't make noise in the lab.
- Don't bring the mobile in the lab.
- Don't enter in the lab without permission of lecturer/laboratory technician immediately
- Don't delete or make any modification in system files.
- Don't bring storage devices like pen drive without permission of lecturer/laboratory technician.

Computer Lab Safety Rules

- Know the location of the fire extinguisher and how to use them in case of an emergency.
- Report fires or accidents to your lecturer/laboratory technician immediately
- Report any broken plugs or exposed electrical wires to your lecturer/laboratory technician immediately.
- Avoid stepping on electrical wires or any other computer cables.
- Do not open the system unit casing or monitor casing particularly when the power is turned on.
- Do not touch, connect or disconnect any plug or cable without your lecturer/laboratory technician's permission.
- Do not bring any food or drinks near the machine.