

1. Conversion of IP addresses (e.g. I/P: 154.31.16.13 O/P: 10011010.00011111.00010000.00001101 and I/P: binary dotted O/P: decimal dotted)

Aim: To write a program to implement IP address conversion from decimal to Binary

Description:

Decimal to binary conversion is an important task to understand in IP addressing and Subnetting. IP addressing is a core functionality of networking. The knowledge of how to assign an IP address, or determine the network or host ID via a subnet is vital to any network engineer. The functionality of Decimal to binary conversion is implemented through C with modular mechanism by passing dotted Decimal IP address between the functions and then achieve in binary 10011010.00011111.00010000.00001101

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int convert_binary(int a,int b,int c,int d);

main()
{
    convert_binary(154,31,16,31);
    getch()
}

int convert_binary(int a,int b, int c, int d)
{
    int i,j,k,l;
    int num[10];
```

```
for(i=0;i<=7;i++)  
  
{  
  
num[i]=a%2;  
  
a=a/2;  
  
}  
  
//return (num);  
  
for(i=7;i>=0;i--)  
  
{  
printf("%d", num[i]);printf(".");  
  
for(j=0;j<=7;j++)  
  
{  
  
num[j]=b%2;  
  
b=b/2;  
  
}  
  
for(j=7;j>=0;j--)  
  
{  
  
printf("%d", num[j]);printf(".");  
  
for(k=0;k<=7;k++)  
  
{  
  
num[k]=c%2;  
  
c=c/2;  
  
}  
  
for(k=7;k>=0;k--)  
  
{
```

```
printf("%d", num[k]);}printf(".");  
  
for(l=0;l<=7;l++)  
{  
  
num[l]=d%2;  
  
d=d/2;  
  
}  
  
for(l=7;l>=0;l--)  
{  
printf("%d", num[l]);  
}  
}
```

Output1: 10011010.00011111.00010000.00001101

2. Conversion of IP addresses from decimal to binary (e.g. I/P: 154.31.16.13 O/P: 10011010.00011111.00010000.00001101 and also Binary to Decimal I/P: binary dotted O/P: decimal dotted)

Aim: To write a program to implement IP address conversion from decimal to Binary and Binary to Decimal

Description:

Decimal to binary conversion is an important task to understand in IP addressing and Subnetting. IP addressing is a core functionality of networking. The knowledge of how to assign an IP address, or determine the network or host ID via a subnet is vital to any network engineer. The functionality of Decimal to binary and binary to decimal conversion is implemented through C with modular mechanism by using the `inet_addr()`, `strtok()` and `atoi()` functions to convert the string into an `in_addr_t`, and then convert that to Decimal to binary and binary to decimal

Program:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

int convert_binary(int a,int b,int c,int d);

int * convert_binary1(int a);

main()
{
    int i,j;    //loop index

    int ip[4]; // array to store ip seperated by "."

    int binary[32]; //store ip in binary

    int *bin;

    char *piece;

    char input[20];
```

```

printf("\nEnter ip address:");

scanf("%s",input);          //read ip as string


piece = strtok(input,"."); //split . seperated ip , first call to the function strtok will return
first piece in the string, subsequent calls to the function with null as first parameter will
return subsequent pieces
i=0;

do
{

ip[i++]=atoi(piece);      // atoi(string) converts string to integer, store
                             pieced string into integer array

piece=strtok(NULL,".");    // this is subsequent call

} while(piece && i<4);


for(j=0;j<4;j++) // loop through ips e.g 192.168.0.1 first loop will process ip[0] which has
192 stored

{

bin = convert_binary1( ip[j] );    //convert integer to binary , stored in integer array

for(i=0;i<8;i++)

{

binary[j*8+i]=bin[i]; // store converted binary into correct position in 32 binary
array

}

}


for(i=0;i<32;i++)
{
if(i>0 && (i)%8==0)

printf("."); //print a "." after printing 8 digits

```

```

printf("%d",binary[i]);           //print binary values of given ip saved in this int array
}

printf("\n");

}

//takes an integer as input
//returns binary stored in integer array
int * convert_binary1(int a)

{

int i,j,k,l;

int *num=(int *)malloc(sizeof(int)*8); //allocating memory of 8 integers

for(i=0;i<=7;i++)

{

num[7-i]=a%2;

a=a/2;

}

return num;

}

int convert_binary(int a,int b, int c, int d)

{

int i,j,k,l;

int num[10];

for(i=0;i<=7;i++)

{

num[i]=a%2;

a=a/2;

```

```
}  
//return(num);  
  
for(i=7;i>=0;i--)  
  
{  
printf("%d", num[i]);  
  
}  
printf(".");  
  
for(j=0;j<=7;j++)  
{  
  
num[j]=b%2;  
  
b=b/2;  
}  
for(j=7;j>=0;j--)  
  
{  
printf("%d", num[j]);  
}  
printf(".");  
  
for(k=0;k<=7;k++)  
  
{  
  
num[k]=c%2;  
  
c=c/2;  
  
}  
  
for(k=7;k>=0;k--)  
  
{  
printf("%d", num[k]);}printf(".");  
  
for(l=0;l<=7;l++)  
  
{  
  
num[l]=d%2;
```

```
d=d/2;  
}  
  
for(l=7;l>=0;l--)  
  
{  
printf("%d", num[l]);  
}  
  
}
```

Input : Enter a IP Address : 10011010.00011111.00010000.00001101

Output: 154.31.16.13

3. Configuration of IP address, Subnet Mask and Default Gateway

Aim: To configure IP address, Subnet Mask and Default Gateway in the Network Machines

Description:

IP (Internet Protocol) addresses are used to identify hardware devices on a network. The addresses allow these devices to connect to one another and transfer data on a local network or over the internet..Example: 192.169.16.89. IP address are two types they are IPV4 is 32 bit IP address and IPV6 is 128 bit IP address IP address also classify various ways Class A ,Class B, Class C, Class D and Class E,

A subnet mask is a 32-bit number created by setting host bits to all 0s and setting network bits to all 1s. In this way, the subnet mask separates the IP address into the network and host addresses. The “255” address is always assigned to a broadcast address, and the “0” address is always assigned to a network address. Neither can be assigned to hosts, as they are reserved for these special purposes. The IP address, subnet mask and gateway or router comprise an underlying structure—the Internet Protocol—that most networks use to facilitate inter-device communication.

When organizations need additional sub networking, subletting divides the host element of the IP address further into a subnet. The goal of subnet masks include: Reducing broadcast volume and thus network traffic Enabling work from home Allowing organizations to surpass LAN constraints such as maximum number of hosts

A default gateway is the node in a computer network using the Internet protocol suite that serves as the forwarding host (router) to other networks when no other route specification matches the destination IP address of a packet. Internet default gateways are typically one of two types:

In home or small business networks with a broadband router to share the internet connection, the home router serves as the default gateway.

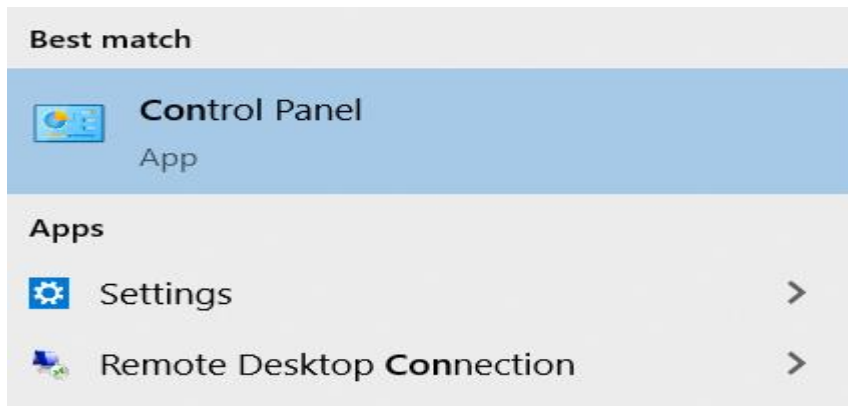
In home or small business networks without a router, such as for residences with dial-up internet access, a router at the internet service provider location serves as the default gateway.

Default network gateways can also be configured using a computer instead of a router. These gateways use two network adapters: one is connected to the local subnet and the other is connected to the outside network.

Procedure:

Steps to follow:

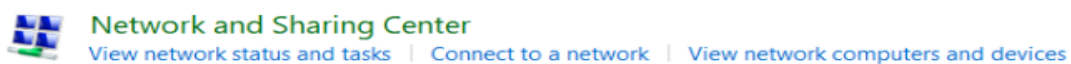
1. Go to the search bar and search for control panel. Once it appears then double click on it. A dialogue box will appear.



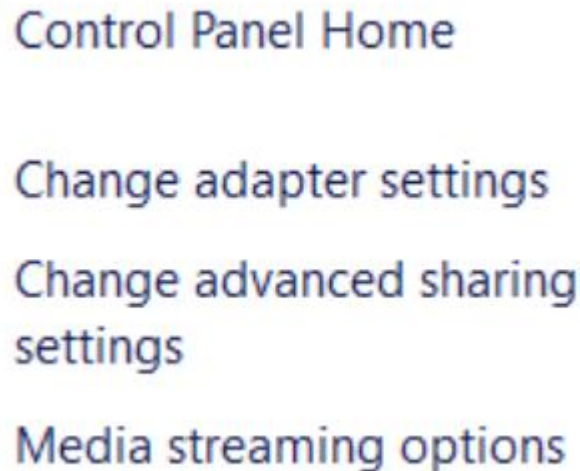
2. In the dialogue box, find “Networks and Internet” option and click on it.



3. A new menu will appear, find “Network and Sharing Center”, present in the left hand side. Click on it.



4. Click on “Change adapter settings” at the left hand side of the menu.

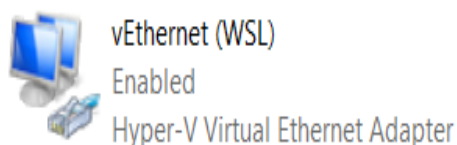
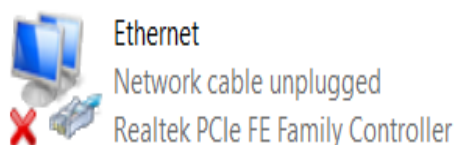


5. Right click on the desired network to which you want your computer to be connected. If you are unable to find the desired network then do the following –

Check whether the router is switched on or not.

Check if the ethernet wire has been properly plugged or not.

If you are using hotspot, then check whether the device is visible to other devices or not. You can check it by going to the tethering hotspot settings of your phone.

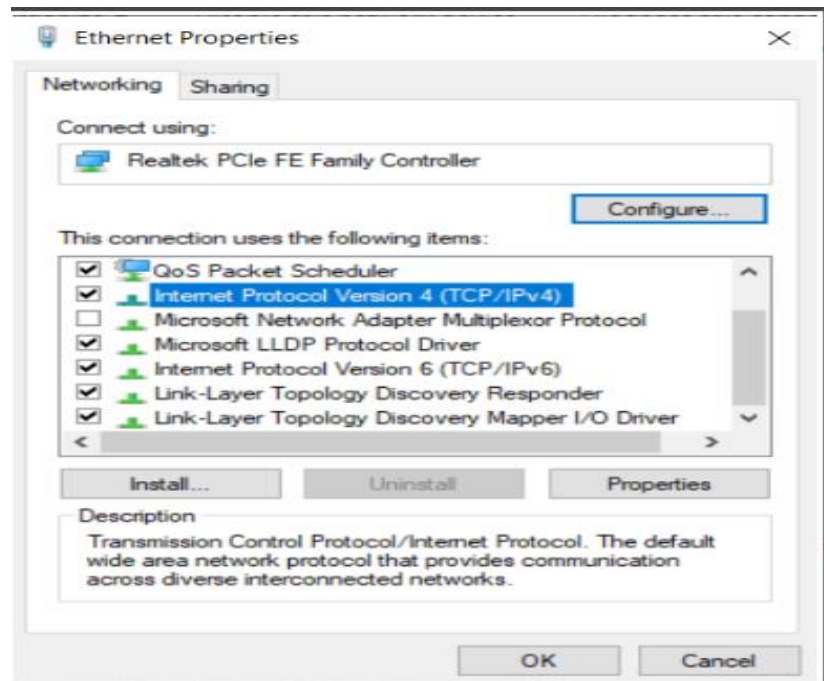


6. Click on the “Properties”.

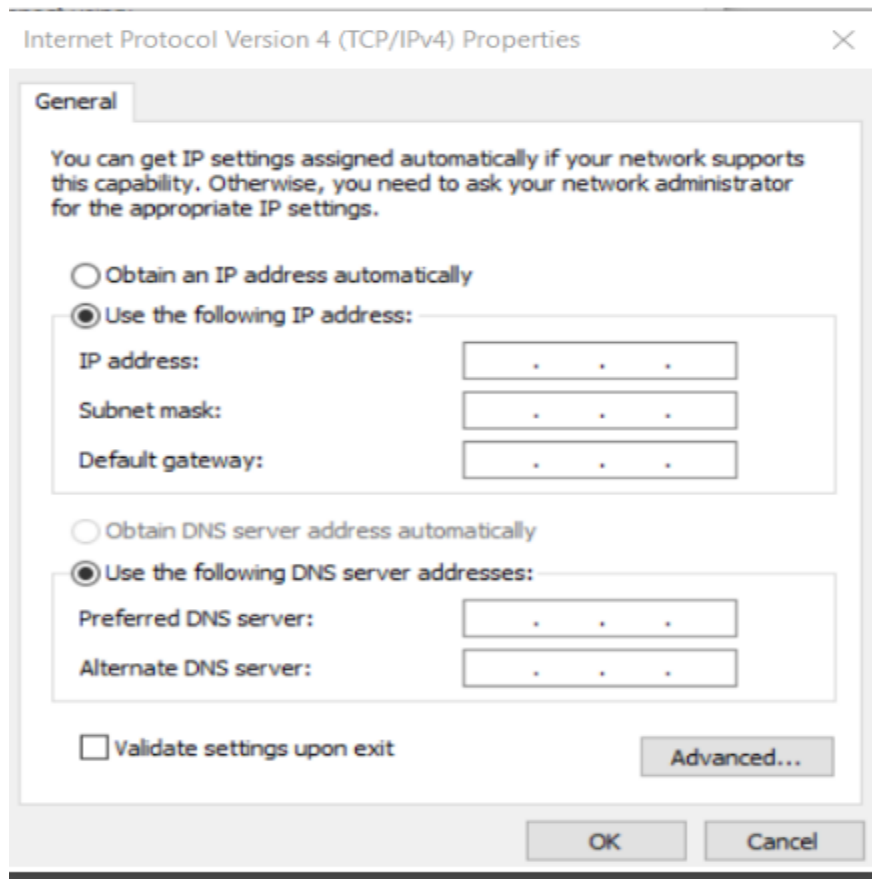


7. Select IPv4.

The IPv4 address is a 32-bit number that uniquely identifies a network interface on a machine. It is widely used even when its successor protocol, i.e. IPv6 is more efficient.



8. Go to “Properties” and Click on the “Use the following IP address” radio button for manually adding the IP address.



9. Enter the details provided by your ISP. These includes –

IP address –

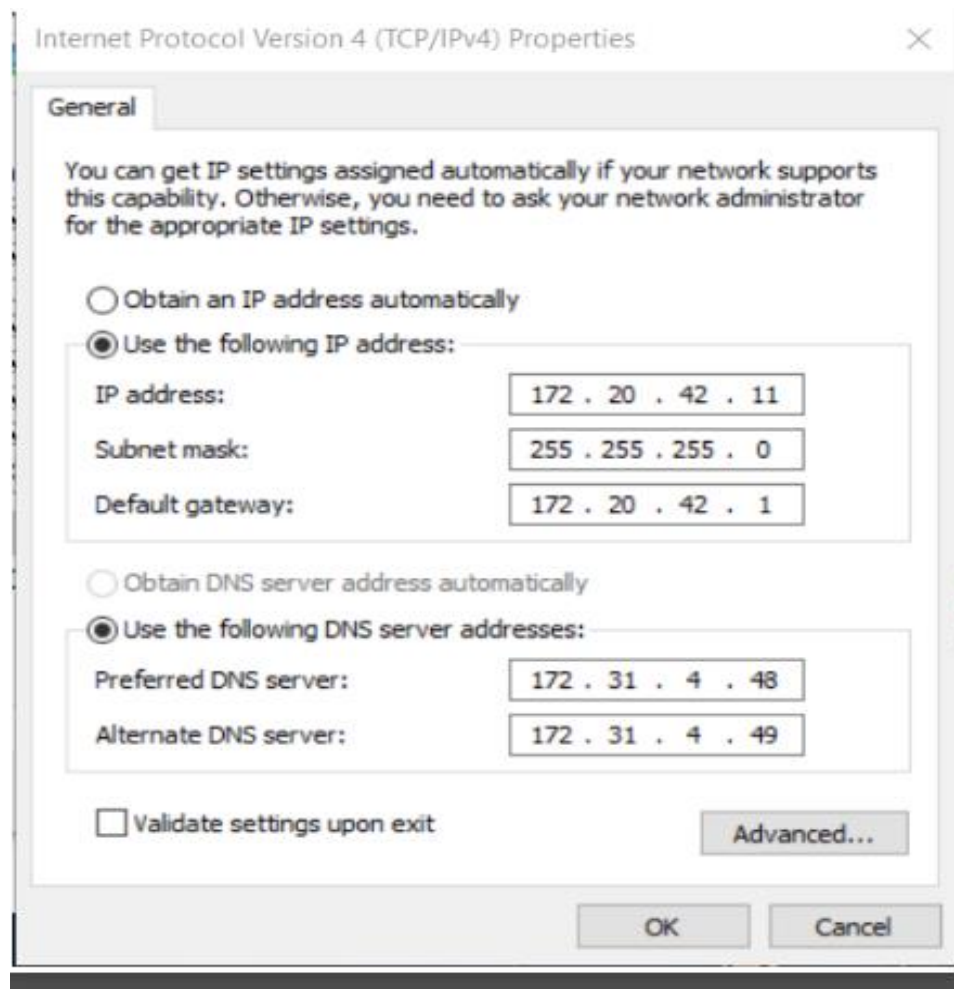
An IP address is a unique address that identifies a device on the internet or a local network. IP stands for “Internet Protocol,” which is the set of rules governing the format of data sent via the internet or local network.

Subnet mask –

A subnet mask is used to divide an IP address into two parts. One part identifies the host (computer), the other part identifies the network to which it belongs.

DNS server –

It is basically a phonebook of internet that maps IP addresses to its domain name



10. Click OK to save and apply your settings

4. Installation and introduction to Wire shark

Aim: Installation of Wire Shark Tool Software in Computer with contains windows OS

Description:

Wire shark is an open source packet Sniffer and Analysis tool. Wire shark captures packets on the LAN from each connected device to the network .Wire shark is used for monitoring traffic troubleshooting and network issues inspecting individual packets .Wire shark is used for detect the suspicious activities in the network by analyzing packets Wire shark is must note a tool for network administrators and cyber security specialist

Wire shark contain various core components such as Main toolbar, filter toolbar and status toolbar, promiscuous mode and various filters such as display and capture filters to analysis of traffic in the network connected devices

Some of the Features of Wire Shark:

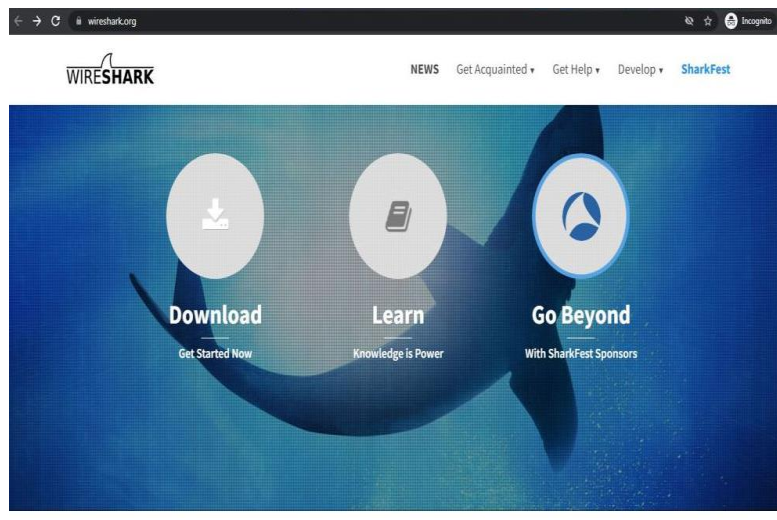
- Available for UNIX and Windows.
- Capture live packet data from a network interface.
- Display packets with very detailed protocol information.
- Save packet data captured.
- Filter packets on many criteria.
- Search for packets on many criteria.
- Colorize packet display based on filters.
- Create various statistics.

Wire shark is pre installed in Kali Linux and Parrot OS Wire shark can download for windows and Mac as well its interface and feature are same across the all in operating systems

Procedure:

Steps to follow for installation of Wire Shark:

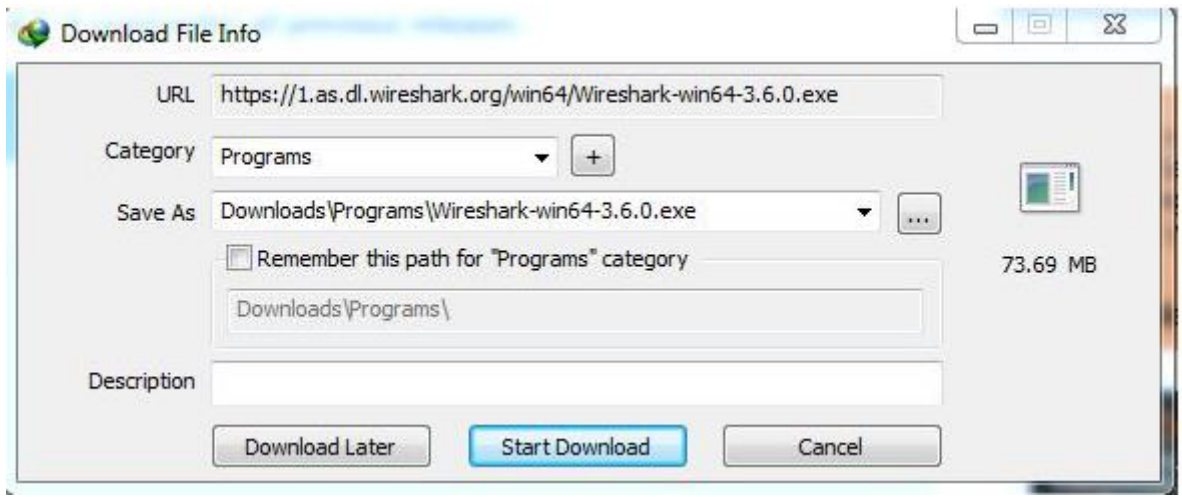
1. Open Web Browser Download wire shark from www.wireshark.org



Step 2: Click on Download, a new webpage will open with different installers of Wireshark.



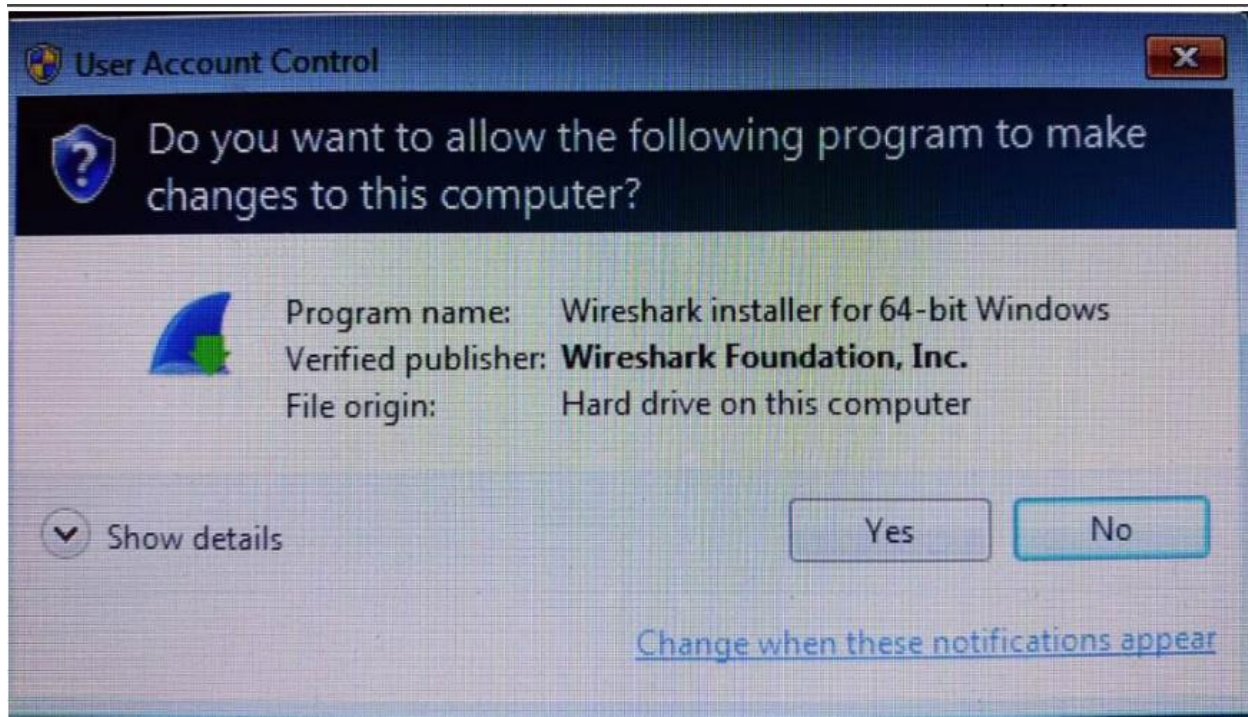
Step 3: Downloading of the executable file will start shortly. It is a small 73.69 MB file that will take some time.



Step 4: Now check for the executable file in downloads in your system and run it.



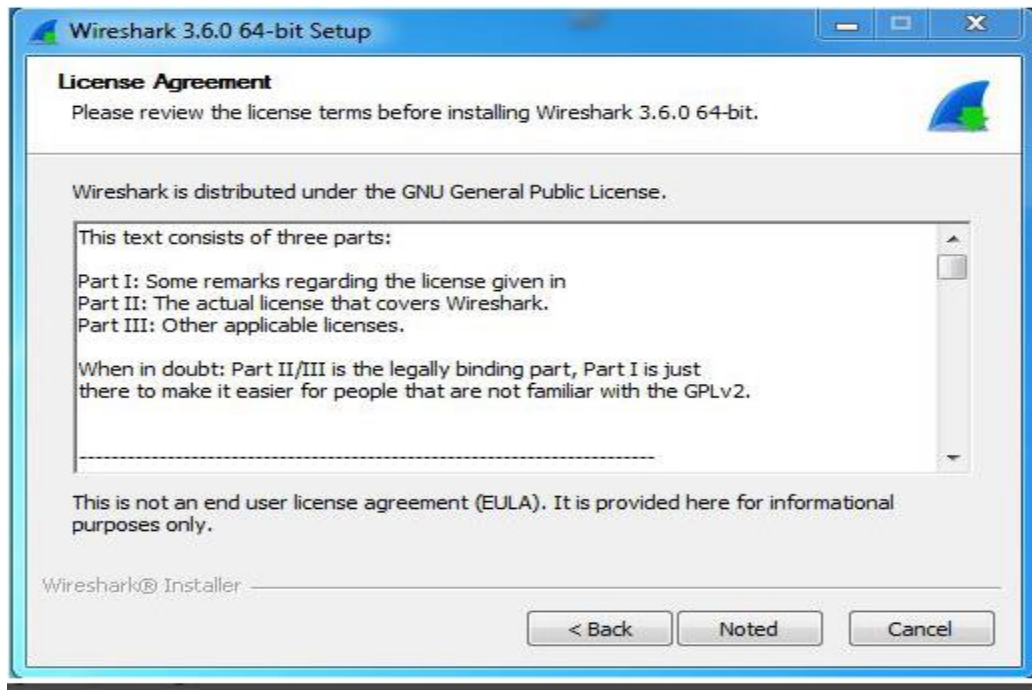
Step 5: It will prompt confirmation to make changes to your system. Click on Yes.



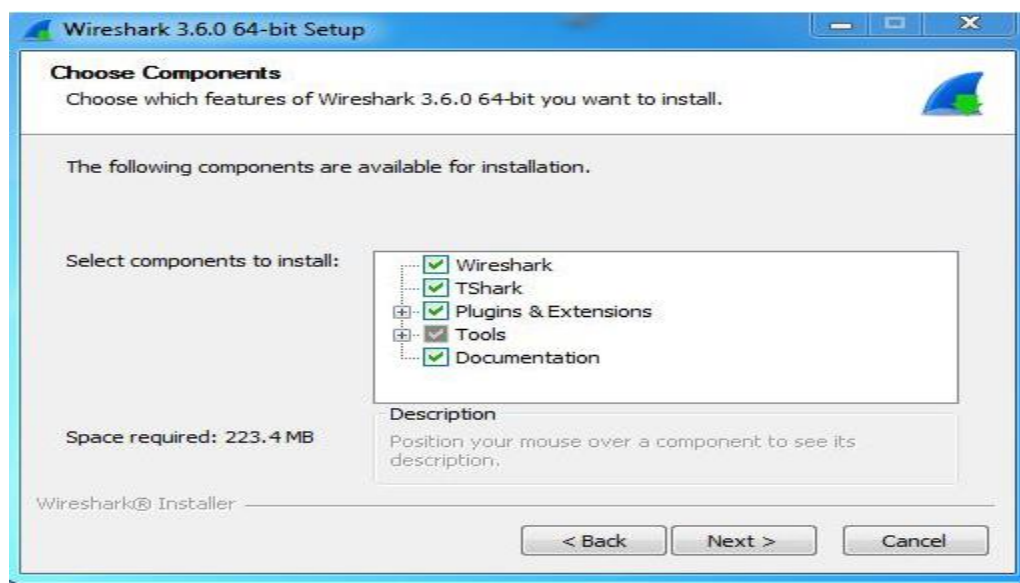
Step 6: Setup screen will appear, click on Next.



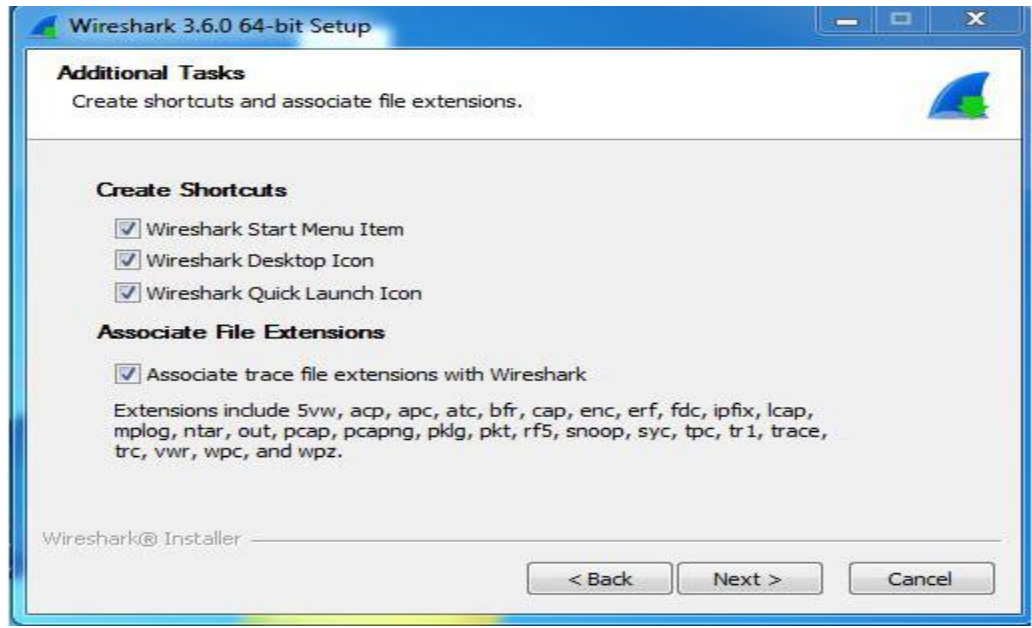
Step 7: The next screen will be of License Agreement, click on Noted.



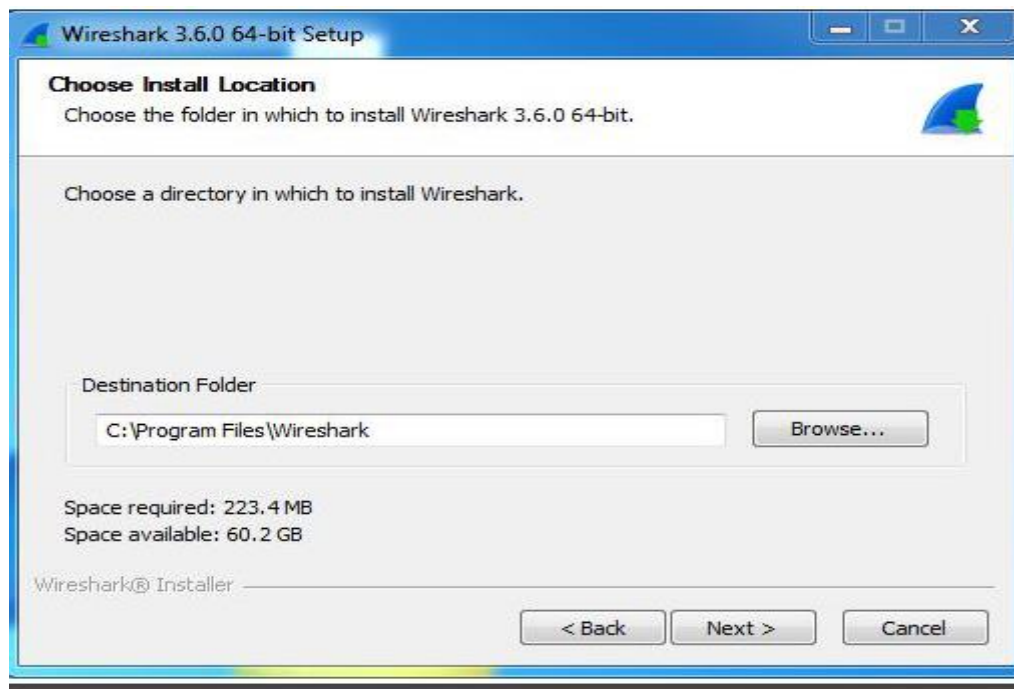
Step 8: This screen is for choosing components, all components are already marked so don't change anything just click on the Next button.



Step 9: This screen is of choosing shortcuts like start menu or desktop icon along with file extensions which can be intercepted by Wire shark, tick all boxes and click on Next button.



Step 10: The next screen will be of installing location so choose the drive which will have sufficient memory space for installation. It needed only a memory space of 223.4 MB



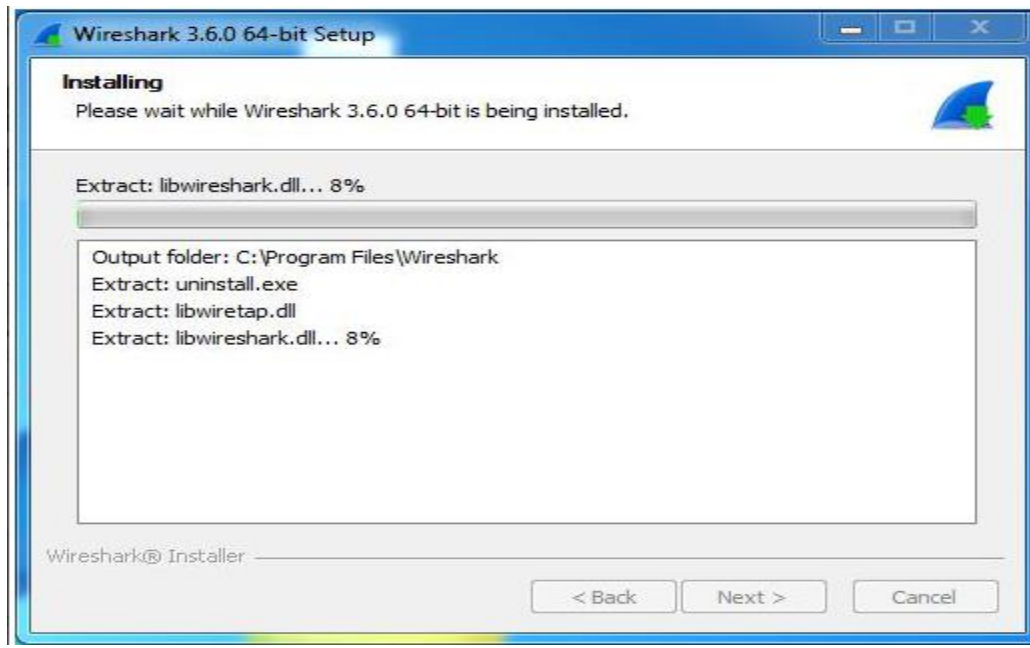
Step 11: Next screen has an option to install Npcap which is used with Wire shark to capture packets pcap means packet capture so the install option is already checked don't change anything and click the next button.



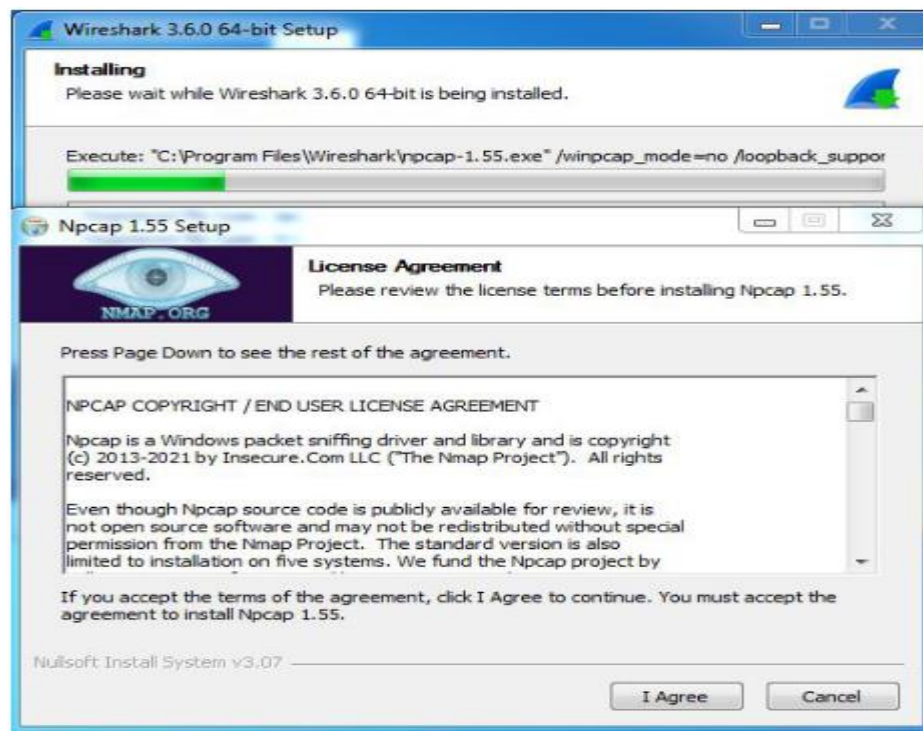
Step 12: Next screen is about USB network capturing so it is one's choice to use it or not, click on Install.



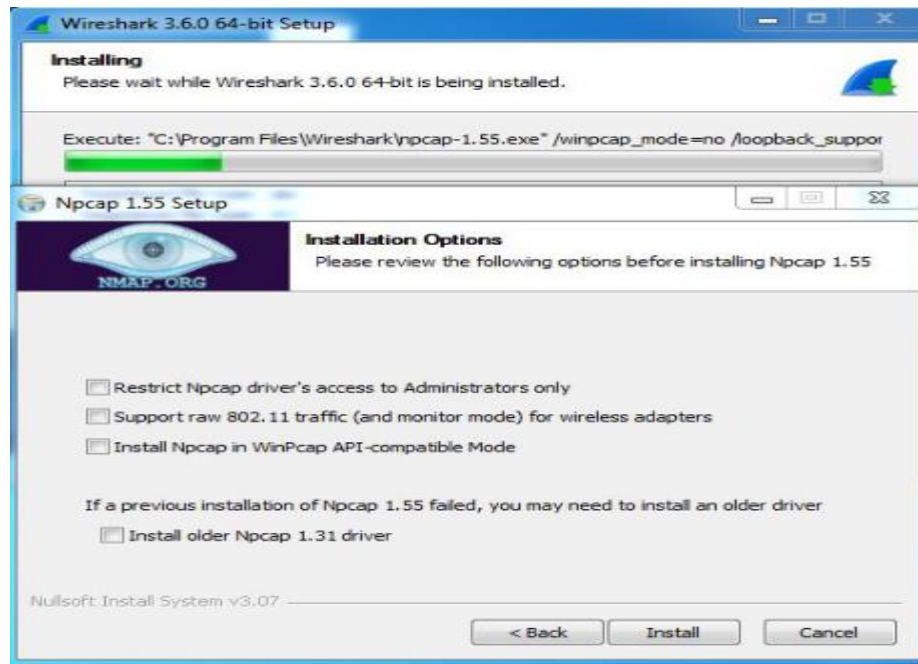
Step 13: After this installation process will start.



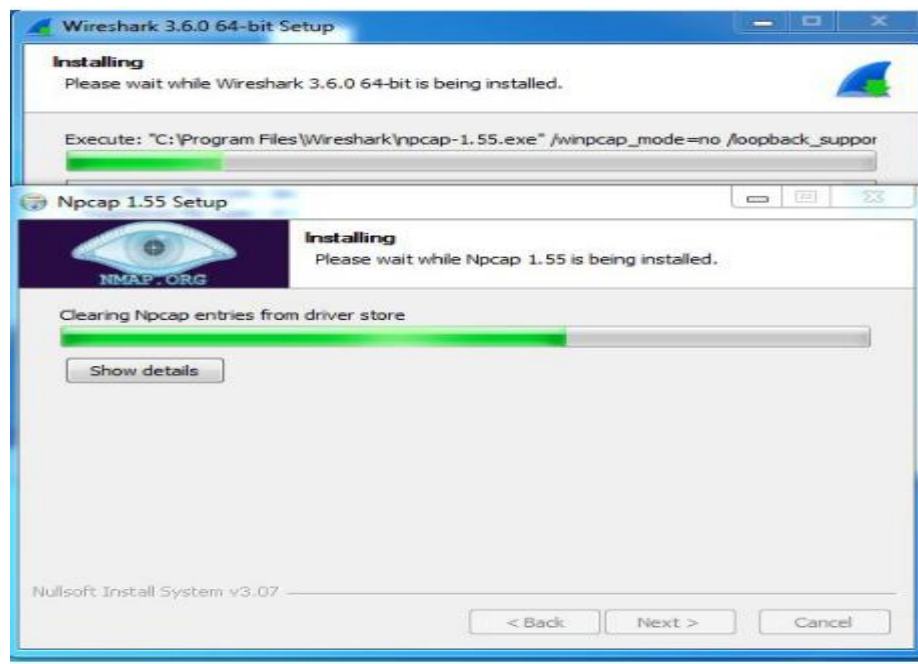
Step 14: This installation will prompt for Npcap installation as already checked so the license agreement of Npcap will appear to click on the I Agree button.



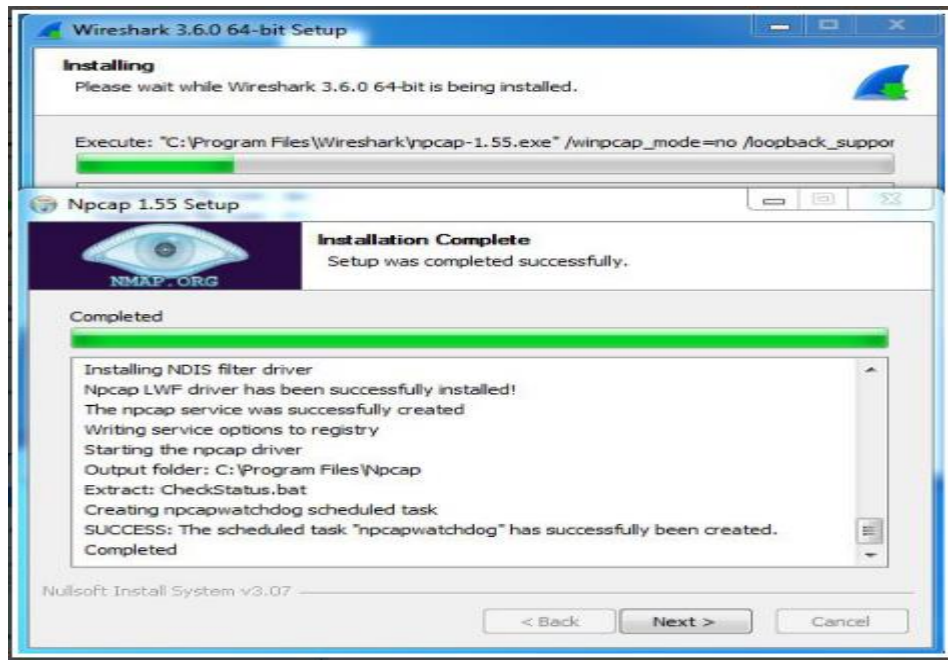
Step 15: Next screen is about different installing options of npcap, don't do anything click on Install.



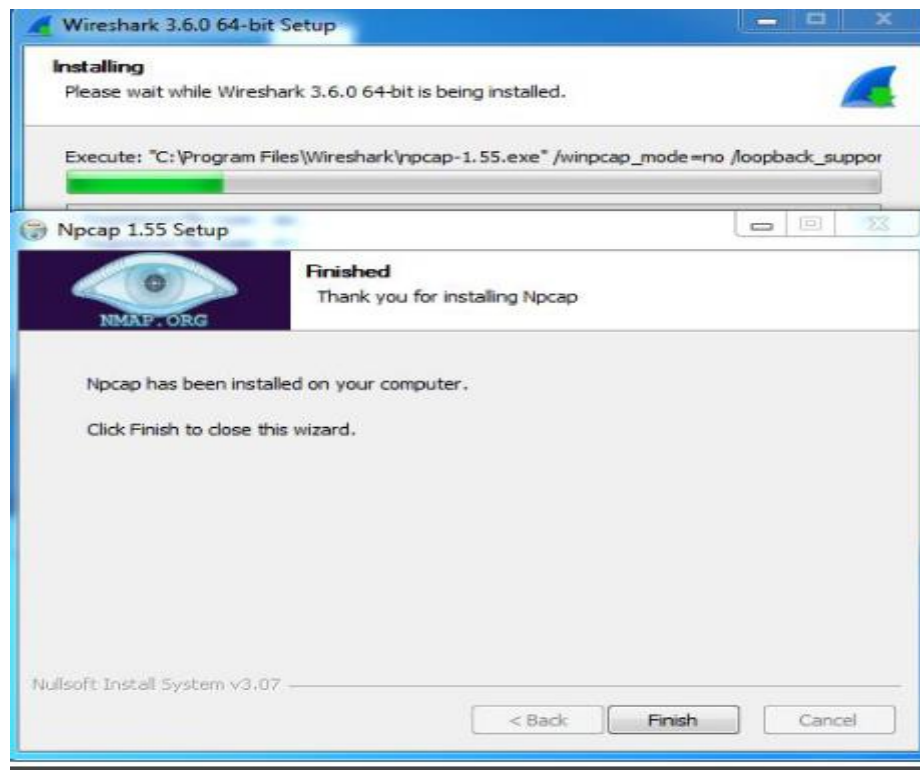
Step 16: After this installation process will start which will take only a minute.



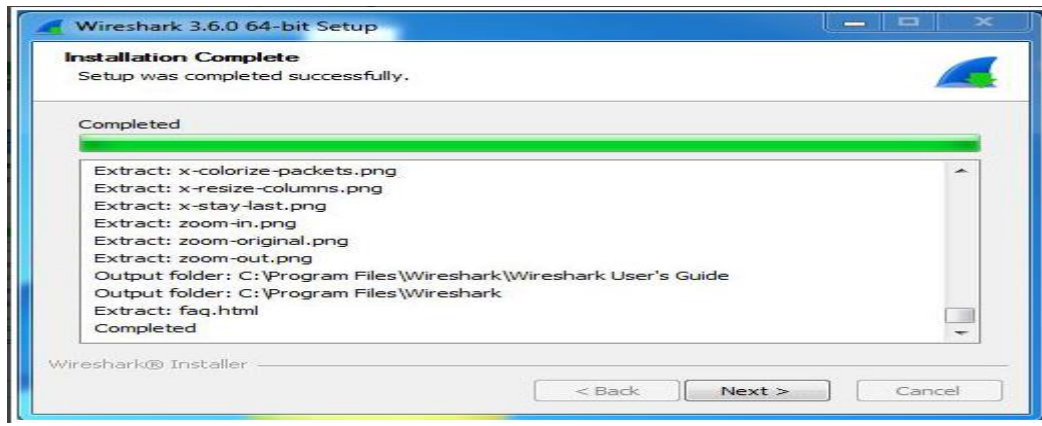
Step 17: After this installation process will complete click on the Next button.



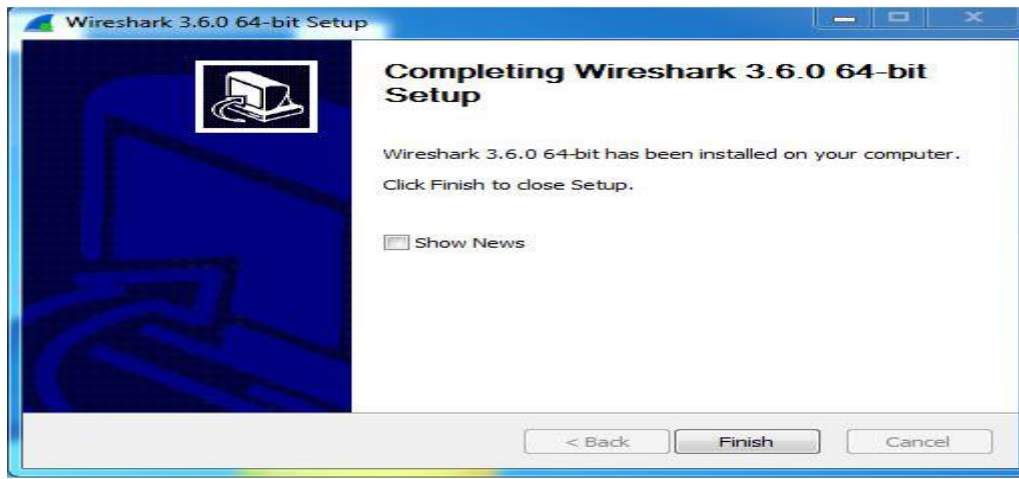
Step 18: Click on Finish after the installation process is complete.



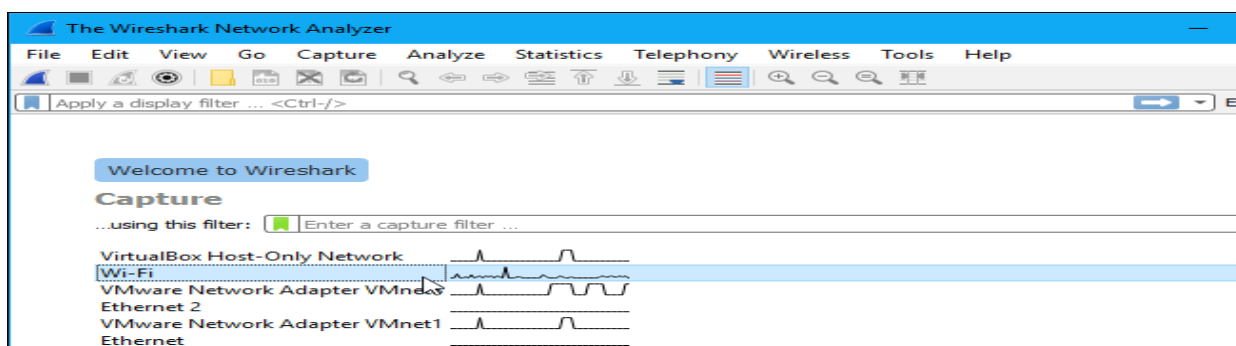
Step 19: After this installation process of Wire shark will complete click on the Next button.



Step 20: Click on Finish after the installation process of Wire shark is complete.



Step 21: Click on Wire Shark Icon and Start to work



5. To write a program to implement character count framing method.

Aim: To write a program to implement character count framing method.

Description:

This method uses a field in the header to specify the number of characters in the frame. When the data link layer at the destination sees the character count, it knows how many characters follow, and hence where the end of the frame is. The disadvantage is that if the count is garbled by a transmission error, the destination will lose synchronization and will be unable to locate the start of the next frame. So, this method is rarely used.

Program:

```
#include<string.h>
#include<ctype.h>
main()
{
    char a[100];
    int frame=0,noc=0,i,j,n;
    clrscr();
    printf("Enter the String:");
    scanf("%s",a);

    for(i=0;a[i]!='\0';i=i+n)
    {
        noc = a[i];
        n= noc-48;
        frame++;
        printf("\nCharacters in  frame %d is",frame);

        for(j=i;j<i+n;j++)
            printf("%c ",a[j]); // a[ j ]
    }
    printf("\nno of frames %d\n no of characters is %d",frame,strlen(a));
}
```

Output1:

Enter the string:5hell6h3454

Characters in frame 1 is 5 h e l l
Characters in frame 2 is 6 h 3 4 5 4
no of frames 2
no of characters is 11

Output2:

Enter the string: 54321612345

Characters in frame 1 is 5 4 3 2 1
Characters in frame 2 is 6 1 2 3 4 5
no of frames 2
no of characters is 11

6. To write a program to implement character or Byte stuffing.

Aim: To write a program to implement character stuffing.

Description:

In the second method, each frame starts with the ASCII character sequence DLE STX and ends with the sequence DLE ETX. (where DLE is Data Link Escape, STX is Start of Text and ETX is End of Text).

This method overcomes the drawbacks of the character count method. If the destination ever loses synchronization, it only has to look for DLE STX and DLE ETX characters. If however, binary data is being transmitted then there exists a possibility of the characters DLE STX and DLE ETX occurring in the data. Since this can interfere with the framing, a technique called character stuffing is used.

The sender's data link layer inserts an DLE character just before the DLE character in the data. The receiver's data link layer removes this DLE before this data is given to the network layer. However character stuffing is closely associated with 8-bit characters and this is a major hurdle in transmitting arbitrary sized characters.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
    int n,i,j;
    char str[100];
    char str1[3]="DLE",str2[3]="STX",str3[3]="ETX",str4[100]="\0",str5[100]="\0";
    clrscr();
    printf("\n enter the string:");
    gets(str);
    n=strlen(str);
    printf("\n the length of the string is:%d",n);
    n=n+1;
    printf("\n the character count of our input is :%d",n);
    n=n-1;
    strcpy(str4,str1);
    strcat(str4,str2);
    for(i=0,j=0;str[i]!='\0';i++)
    {
        if(str[i]==' ')
        {
```

```
strcat(str5,str1);
j+=3;
}
else
{
str5[j]=str[i];
j++;
}
}
strcat(str4,str5);
strcat(str4,str1);
strcat(str4,str3);

printf("\n after character stuffing:\n");
printf("%s\n",str4);
getch();
}
```

Output:

enter the string: hello buy

the length of the string will be:10

the character count of our input is :11

after character stuffing:

DLESTX hello DLE buy DLEETX

7. To write a program to implement bit stuffing.

Aim: To write a program to implement bit stuffing.

Description:

This is a kind of stuffing in which each frame begins and ends with a special bit pattern 01111110, called a flag byte. Whenever the sender's data link layer encounters five consecutive ones in the data, it automatically stuffs a 0 bit into the outgoing bit stream. When the receiver sees five consecutive incoming 1 bits it automatically de-stuffs the 0 bit. For example if the user data contain the flag pattern, 01111110 the flag is transmitted as 01111010 but stored in receiver's memory as 01111110.

Program:

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
    char s[50],flag[10];
    int i,j=0,l1,l2;
    clrscr();
    printf("enter the string \n");
    gets(s);
    l1=strlen(s);
    l2=l1/8;
    printf("length=%d\tbytes=%d",l1,l2);
    printf("\nstartingflag\t\tdata\t\t\tendingflag\n");
    strcpy(flag,"01111110");
    printf("%s",flag);
    for(i=0;s[i]!='\0';i++)
    if(s[i]=='0')
    {
        printf("%c",s[i]);
        j=0;
    }
    else
    {
        printf("%c",s[i]);
        j++;
        if(j==5)
        {
            printf("0");
            j=0;
        }
    }
}
```

```
}  
}  
printf("%s",flag);  
getch();  
}
```

Output:

enter the string
011111111011111101110
length=21 bytes=2

startingflag	data	endingflag
01111110	01111101110111110101110	01111110

8. To write a program to implement even or odd parity.

Aim: To write a program to implement even or odd parity.

Description: Even/odd parity:

Computers can sometimes make errors when they transmit data. Even/odd parity is basic method for detecting if an odd number of bits has been switched by accident.

Odd parity: The number of 1-bit must add up to an odd number.

Even parity: The number of 1-bit must add up to an even number.

The computer knows which parity it is using.

- If it uses an even parity: If the number of of 1-bit add up to an odd number then it knows there was an error:
- If it uses an odd: If the number of of 1-bit add up to an even number then it knows there was an error:

Program:

```
#include<stdio.h>
#include<string.h>
int main()
{
    char a[100];
    int i,count_1s=0,l;
    clrscr();
    printf("\nEnter the frame with 1s and 0s:");
    gets(a);
    for(i=0;a[i]!='\0';i++)
    {
        if(a[i]=='1')
            count_1s++;
    }
    if(count_1s%2==0)
    {
        printf("\n %s is Even parity",a);
        l=strlen(a);
        a[l++]='0';
        a[l]='\0';
    }
}
```



```
}  
else  
{  
printf("\n %s is Odd parity",a);  
l=strlen(a);  
a[l]='1';  
a[l+1]='\0';  
}  
printf("\n String after adding parity bit is %s",a);  
return 0;  
}
```

Output1:

Enter the frame with 1s and 0s:110111

110111 is Odd parity

String after adding parity bit is 1101111

Output2:

Enter the frame with 1s and 0s:1101111

1101111 is Even parity

String after adding parity bit is 11011110

