

**Experiment No. 4**

**Title:** Exploring Application layer protocols

**Batch: B1 Roll No.: 1914078 Experiment No.4**

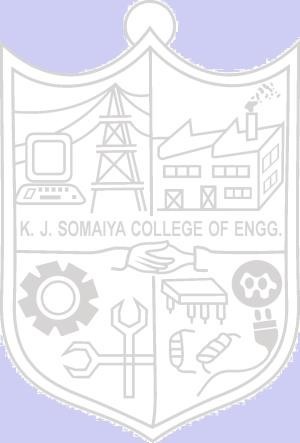
**Aim:** To explore the application layer protocols using Wireshark.

**Resources needed**: Internet, Wireshark software (downloaded from the official site)

**Theory**

**Background of Wireshark**

Wireshark is a network packet analyzer. Any network packet analyzer will try to capture network packets and will try to display that packet data as detailed as possible in human readable format. Wireshark is an open source software project, and is released under the [GNU General Public License (GP](https://www.gnu.org/licenses/gpl-2.0.html)L). We can freely use Wireshark on any number of computers, without worrying about license keys. In addition, all source code is freely available under the GPL. Because of that, it is very easy for people to add new protocols to Wireshark, either as plug-in, or built into the source code. In the past, such tools were either very expensive, proprietary. However, with the advent of Wire-shark, all that has changed. Wireshark is perhaps one of the best open source packet analyzers available today.

**What Wireshark is not**

Here are some things Wireshark does not provide:

1. Wireshark isn’t an intrusion detection system. It will not warn us when someone does strange things on our network that he/she isn’t allowed to do. However, if strange things happen, Wireshark might help you figure out what is really going on.
2. Wireshark will not manipulate things on the network, it will only "measure" things from it. Wireshark doesn’t send packets on the network or do other active things.

**Applications**

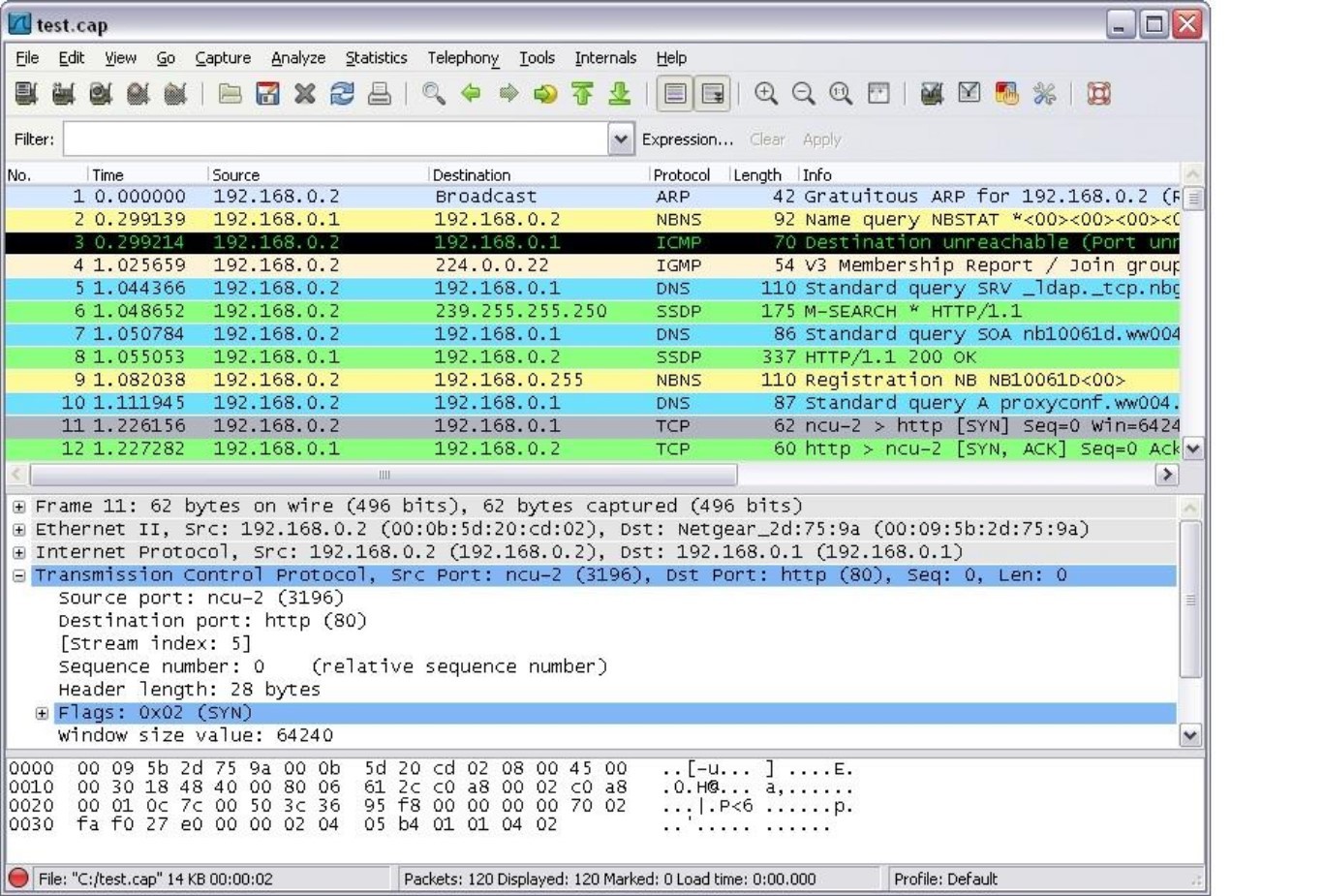
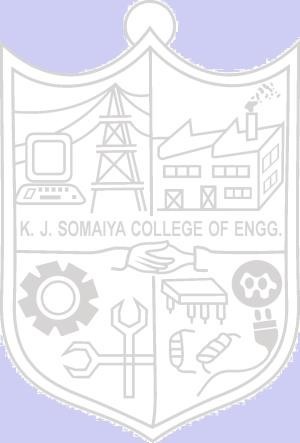
Here are some applications. Many people use Wireshark for doing following things,

* Network administrators use it to ***troubleshoot network problems.***
* Network security engineers use it to ***examine security problems (Network Forensics.)***
* Developers use it to ***debug protocol implementations.***
* People use it to ***learn network protocol* internals/analysis.**

Beside these examples Wireshark can be helpful in many other situations too.

The following are some of the features Wireshark has:

* Available for*UNIX* and *Windows operating systems.*
* *Capture* live packet data from a chosen network interface.
* *Open* files containing packet data captured with tcpdump/WinDump and a number of other packet capture programs.
* *Import* packets from text files containing hex dumps of packet data.
* Display packets with *very detailed protocol information*.
* *Save* packet data captured.



* *Export* some or all packets in a number of capture file formats.
* *Filter packets* on many criteria.
* *Search* for packets on many criteria.
* *Colorize* packet display based on filters.
* Create various *statistics*.
* …and *a lot more!*

However, to really appreciate its power we have to start using it. Here is a snapshot of Wireshark main menu.

Most important menus are : 1) Capture 2) Analyze 3) Statistics Students are expected to explore all these menus and sub-menus in details.

Wireshark can capture traffic from many different network media types including wireless LAN as well. Which media types are supported, depends on many things like the operating system we are using and the hardware support.

**Physical Interfaces support**

1. ATM - capture [ATM](https://wiki.wireshark.org/ATM) traffic
2. [Bluetooth- c](https://wiki.wireshark.org/CaptureSetup/Bluetooth)apture Bluetooth traffic .
3. Cisco HDLC links - capture on synchronous links using Cisco HDLC encapsulation. D. [Ethern](https://wiki.wireshark.org/CaptureSetup/Ethernet)et- capture on different topologies, including

switched networks.

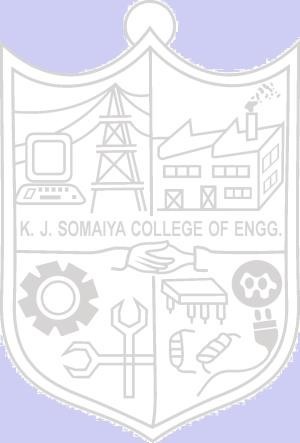
1. Framerelay – captures framerelay traffic.
2. IrDA capture IrDA traffic - currently limited to Linux.
3. PPP links - capture on dial-up lines, ISDN connections and PPP-over-Ethernet (PPPoe, e.g. ADSL)
4. Tokenring - capture on Tokenring adapters, promiscuous mode and switched networks
5. USB- capture of raw USB traffic
6. WLAN- capture on 802.11 ([WLAN, Wi-Fi) in](https://wiki.wireshark.org/WLAN)terfaces, including "monitor mode" , raw 802.11 headers and radio information

# Virtual interfaces :

1. Loopback - capture traffic from a machine to itself, including the IP address 127.0.0.1
2. Pipes - use UNIX pipes to capture from other applications (even remote!)
3. VLAN – capture VLAN traffic, including VLAN tags.

In addition to this, Wireshark can do following things.

1. Import files from many other capture programs.
2. Wireshark can open packets captured from a large number of other capture programs.



1. Export files for many other capture programs.
2. Wireshark can save packets captured in a large number of formats of other capture programs.
3. Can be used as a protocol decoder.
4. Few application layer protocols(HTTP,FTP,SMTP etc..) make use of TCP to deliver their purpose so you see tcp as their backbone.

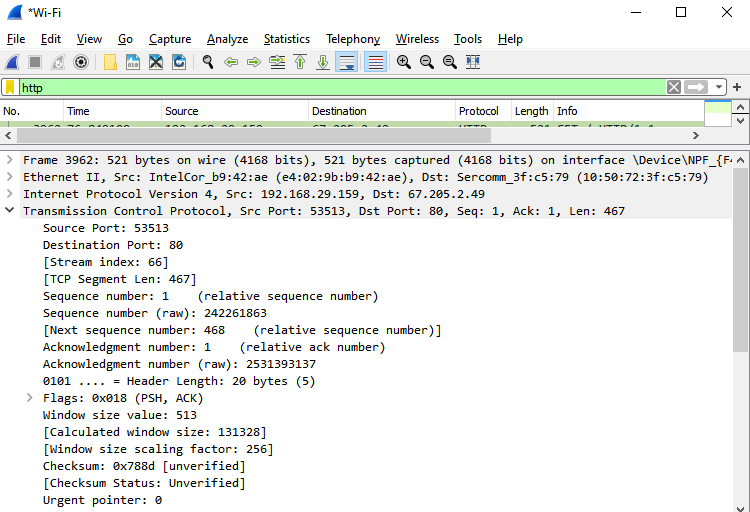
# Procedure:

1. Start the machine as an administrator.
2. Start internet.
3. Go to the official website of Wireshark . ( [www.wireshark.org) a](http://www.wireshark.org/)nd download the old stable version of Wireshark for 32 bit windows operating system.
4. After successful installation you will get the blue icon of Wireshark on the desktop.
5. Click on the icon and start the software.
6. Choose an interface and start capturing the packets.
7. Study the packet details of all the protocols of Application layer( DNS,HTTP and FTP etc.)
8. Perform the statistics for a particular protocol. ( Every student should perform for different protocol. )

10. Show the output to the teacher and get it approved.

# Activity:

**Identify one application layer protocol , study analyse and interpret the same along with snapshots**.



**Opened friv.com**

**Source Port**: 53513

**Destination Port**: 80

**Source**: 192.168.29.159

**Destination**: 67.205.2.49

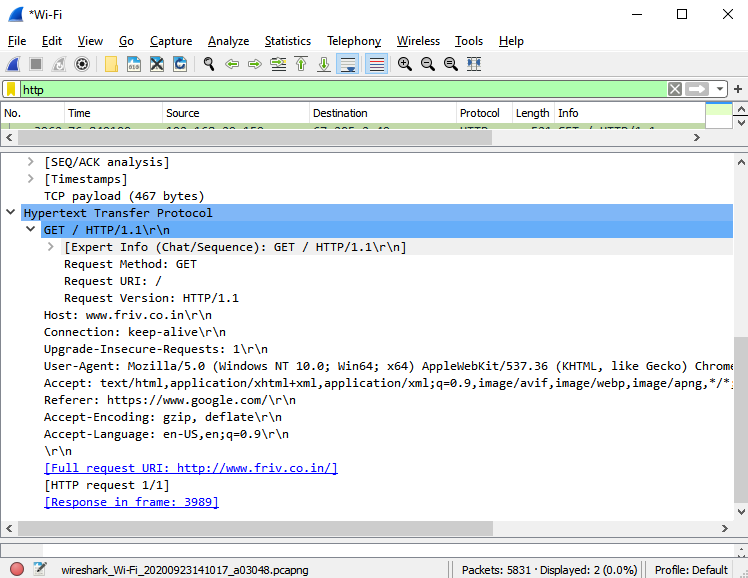
**Protocol**: Hypertext Transfer Protocol

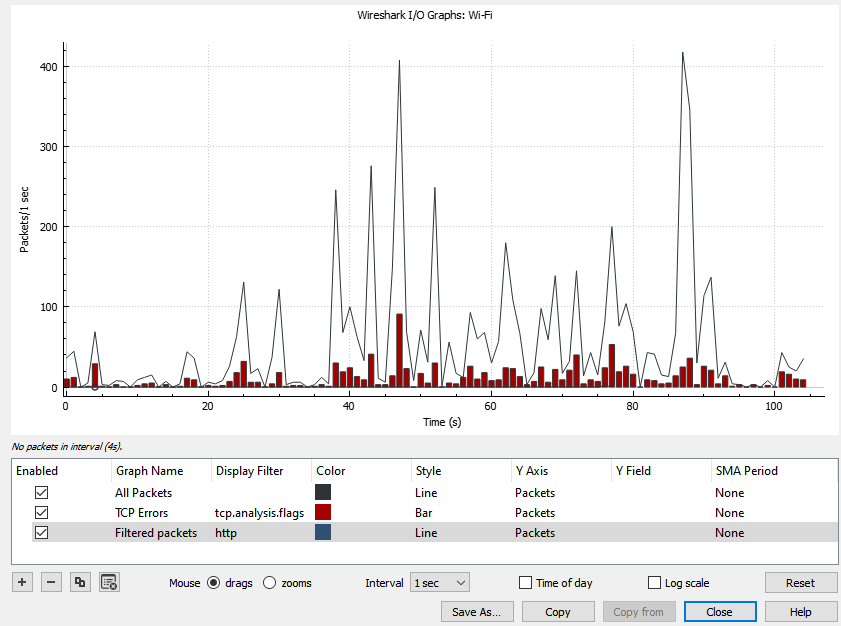
**Request**: GET

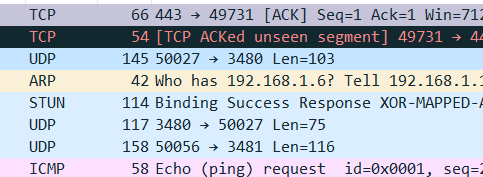
**Full request URI**: http://www.friv.co.in/

**Host**: www.friv.co.in\r\n

**Packet number**: 3989







**Questions:**

1. **What is MIME? List the MIME headers**.

Multipurpose Internet Mail Extensions (MIME) is an Internet standard that extends the format of email messages to support text in character sets other than ASCII, as well as attachments of audio, video, images, and application programs.

MIME defines five media types for discrete content: text, image , audio , video , and application

The following headers are defined in MIME:

* MIME-Version
* Content-Type
* Content-Transfer-Encoding
* Content-ID
* Content-Description
* Content-Disposition

1. **Why do we need POP3 for electronic mail?**

* Post Office Protocol version 3 (POP3) is a standard mail protocol used to receive emails from a remote server to a local email client. If you use POP3, your messages are stored on your local computer, which reduces the space your email account uses on your web server.
* POP3 allows you to download email messages on your local computer and read them even when you are offline.

1. **What are the FTP transmission modes?**

* Transmission mode refers to the mechanism of transferring of data between two devices connected over a network.
* FTP defines three different transmission modes (also called transfer modes) that specify exactly how data is sent from one device to another over an opened data channel: stream mode, block mode, and compressed mode.

1. **Name the three components of a browser.**

* A browser consists of a controller, client program, and interpreters.

1. **What is URL and what are its components?**

* URL stands for Uniform Resource Locator. A URL is nothing more than the address of a given unique resource on the Web.
* It is the mechanism used by browsers to retrieve any published resource on the web.
* Example: <http://www.example.com:80/path/to/myfile.html?key1=value1&key2=value2#SomewhereInTheDocument>
* It’s components are:
* Protocol: http
* domain name: www.example.com
* port: :80
* path to the resource : path/to/myfile.html
* parameters : key1=value1&key2=value2
* anchor: #SomewhereInTheDocument

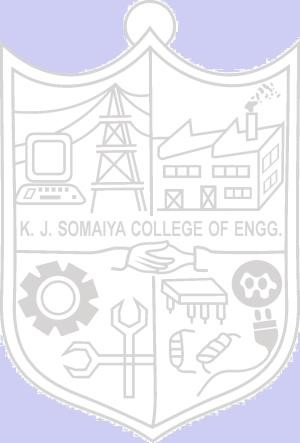
**Outcomes: To enumerate the layers of the OSI/TCP model, their functions and ptotocols (Application layer)**

**-------------------------------------------------------------------------------------------------------**

**Conclusion:** We explored the application layer protocols using Wireshark.

**-------------------------------------------------------------------------------------------------------**

**Grade: AA / AB / BB / BC / CC / CD /DD**



**Signature of faculty in-charge with date**

**References:**

**Books and Websites:**

* Behrouz A Forouzan, Data Communication and networking,Tata Mc Graw hill, India, 4th Edition
* [http://www.wireshark.org](http://www.wireshark.org/)
* Wireshark user manual

(Autonomous College Affiliated to University of Mumbai)