**EXPERIMENT NO:9** **DATE:**

# SETTING UP THE LAMP STACK

## INTRODUCTION

LAMP is an acronym that stands for Linux, Apache, MySQL, and PHP/Perl/Python. It is a popular open-source software stack used for web development and hosting dynamic websites and web applications.

Together, the LAMP stack provides a complete environment for developing, deploying, and hosting web applications. It is a widely adopted solution due to its open-source nature, flexibility, and robustness.

## LAMP Stack

**Linux:** It refers to the operating system, specifically a variant of the Unix operating system, that serves as the foundation for the LAMP stack. Linux provides the underlying infrastructure and resources for running web servers and other software components.

**Apache:** It is a widely used web server software that delivers web content over the internet. Apache handles HTTP requests from web browsers and serves web pages to users. It supports various features like virtual hosting, SSL/TLS encryption, and URL rewriting.

**MySQL:** It is a popular open-source relational database management system (RDBMS) that provides a robust and scalable solution for storing and managing data. MySQL is widely used for web applications, and it allows developers to create, read, update, and delete data using the SQL (Structured Query Language) language.

**PHP/Perl/Python:** These are scripting languages commonly used for web development in the LAMP stack.

## Advantages of LAMP

- **Open Source**: LAMP components are freely available, customizable, and benefit from a large community of developers for updates, bug fixes, and security patches.
- **Cost-effective**: LAMP reduces costs as it is built on open-source software, eliminating licensing fees and offering affordable solutions.
- **Community Support**: LAMP has a vast and active developer community, providing abundant resources, documentation, and support for troubleshooting.
- **Scalability:** LAMP can handle high web traffic and easily scale with features like Apache's concurrency handling and MySQL's replication and clustering.

- **Compatibility**: LAMP is compatible with various platforms, making deployment on different operating systems and hardware configurations simple.
- **Security**: LAMP emphasizes security with a robust operating system (Linux) and regular security updates for Apache, MySQL, and scripting languages.
- **Rapid Development**: LAMP offers numerous frameworks, libraries, and pre-built modules for accelerated web development.
- **Flexibility**: LAMP allows customization and supports multiple scripting languages (PHP, Perl, Python), enabling developers to choose the best tools for their needs.

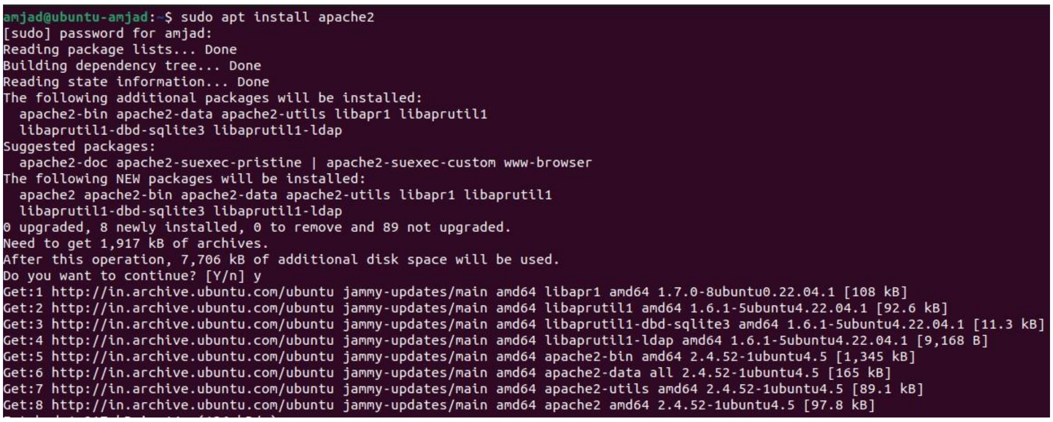
## INSTALLATION

### Step 1 — Installing Apache and Updating the Firewall

The Apache web server is a popular open-source web server that can be used along with PHP to host dynamic websites. It's well-documented and has been in wide use for much of the history of the web.

First, make sure your apt cache is updated with: *sudo apt update*

Once the cache has been updated, you can install Apache with: *sudo apt install apache2*

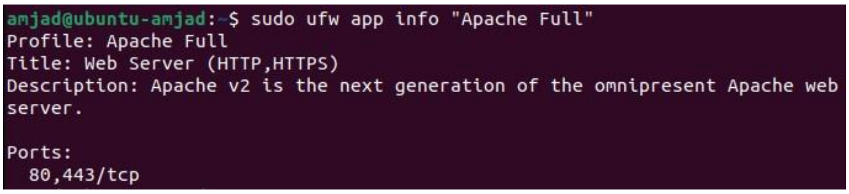


```
amjad@ubuntu-amjad:~$ sudo apt install apache2
[sudo] password for amjad:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap
0 upgraded, 8 newly installed, 0 to remove and 89 not upgraded.
Need to get 1,917 kB of archives.
After this operation, 7,706 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libapr1 amd64 1.7.0-8ubuntu0.22.04.1 [108 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1 amd64 1.6.1-5ubuntu4.22.04.1 [92.6 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.1-5ubuntu4.22.04.1 [11.3 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 libaprutil1-ldap amd64 1.6.1-5ubuntu4.22.04.1 [9,168 B]
Get:5 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-bin amd64 2.4.52-1ubuntu4.5 [1,345 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-data all 2.4.52-1ubuntu4.5 [165 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2-utils amd64 2.4.52-1ubuntu4.5 [89.1 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu jammy-updates/main amd64 apache2 amd64 2.4.52-1ubuntu4.5 [97.8 kB]
```

### Adjust the Firewall to Allow Web Traffic

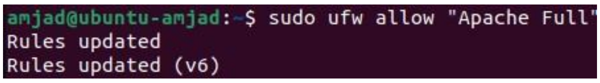
Make sure that your firewall allows HTTP and HTTPS traffic. You can check that UFW has an application profile for Apache like so: *sudo ufw app list*

If you look at the Apache Full profile details, you'll see that it enables traffic to ports 80 and
443: *sudo ufw app info "Apache Full"*



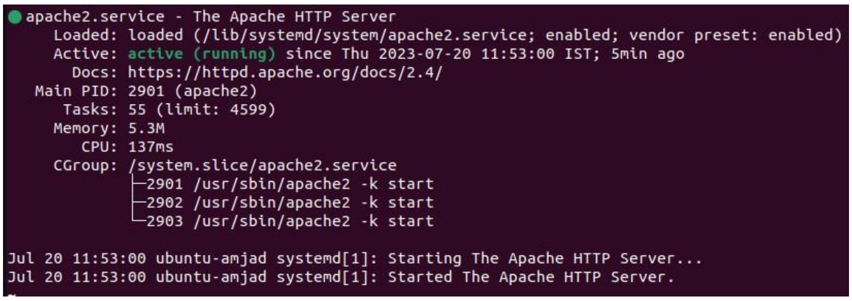
```
amjad@ubuntu-amjad:~$ sudo ufw app info "Apache Full"
Profile: Apache Full
Title: Web Server (HTTP,HTTPS)
Description: Apache v2 is the next generation of the omnipresent Apache web
server.

Ports:
  80,443/tcp
```

To allow incoming HTTP and HTTPS traffic for this server, run: *sudo ufw allow "Apache Full"*



```
amjad@ubuntu-amjad:~$ sudo ufw allow "Apache Full"
Rules updated
Rules updated (v6)
```

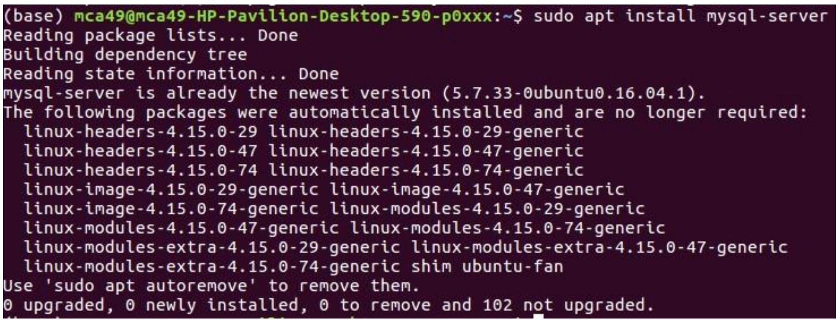
You can do a spot check right away to verify that everything went as planned by visiting your
server's public IP address in your web browser : http://your_server_ip
**OR**
Confirm that Apache is now running with the following command:  *sudo systemctl status apache2*



```
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2023-07-20 11:53:00 IST; 5min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2901 (apache2)
      Tasks: 55 (limit: 4599)
     Memory: 5.3M
        CPU: 137ms
     CGroup: /system.slice/apache2.service
             ├─2901 /usr/sbin/apache2 -k start
             ├─2902 /usr/sbin/apache2 -k start
             └─2903 /usr/sbin/apache2 -k start

Jul 20 11:53:00 ubuntu-amjad systemd[1]: Starting The Apache HTTP Server...
Jul 20 11:53:00 ubuntu-amjad systemd[1]: Started The Apache HTTP Server.
```

## Step 2 — Installing MySQL

MySQL is a database management system. Basically, it will organize and provide access to databases where your site can store information.Again, use apt to acquire and install this software: *sudo apt install mysql-server*



```
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~$ sudo apt install mysql-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
mysql-server is already the newest version (5.7.33-0ubuntu0.16.04.1).
The following packages were automatically installed and are no longer required:
  linux-headers-4.15.0-29 linux-headers-4.15.0-29-generic
  linux-headers-4.15.0-47 linux-headers-4.15.0-47-generic
  linux-headers-4.15.0-74 linux-headers-4.15.0-74-generic
  linux-image-4.15.0-29-generic linux-image-4.15.0-47-generic
  linux-image-4.15.0-74-generic linux-modules-4.15.0-29-generic
  linux-modules-4.15.0-47-generic linux-modules-4.15.0-74-generic
  linux-modules-extra-4.15.0-29-generic linux-modules-extra-4.15.0-47-generic
  linux-modules-extra-4.15.0-74-generic shim ubuntu-fan
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 102 not upgraded.
```

When the installation is complete, run a simple security script that comes pre-installed with MySQL which will remove some dangerous defaults and lock down access to your database system. Start the interactive script by running: *sudo mysql_secure_installation*

When you're finished, test if you're able to log in to the MySQL console by typing: *sudo mysql*

To exit the MySQL console, type: *exit*

## Step 3 — Installing PHP

PHP is the component of your setup that will process code to display dynamic content. It can run scripts, connect to your MySQL databases to get information, and hand the processed content over to your web server so that it can display the results to your visitors.

In addition to the php package, you’ll also need libapache2-mod-php to integrate PHP into Apache, and the php-mysql package to allow PHP to connect to MySQL databases. Run the following command to install all three packages and their dependencies: *sudo apt install php libapache2-mod-php php-mysql*

**Step 4 — Testing PHP Processing on your Web Server**

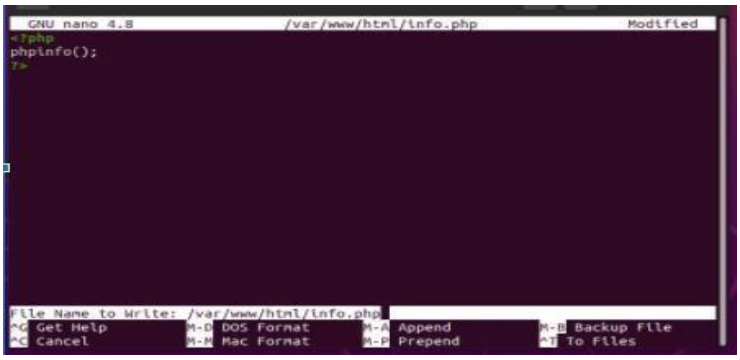
In order to test that your system is properly configured for PHP, create a PHP script called info.php. In order for Apache to find this file and serve it correctly, it must be saved to your web root directory.

Create the file at the web root you created in the previous step by running:

*$ sudo nano /var/www/your_domain/info.php*

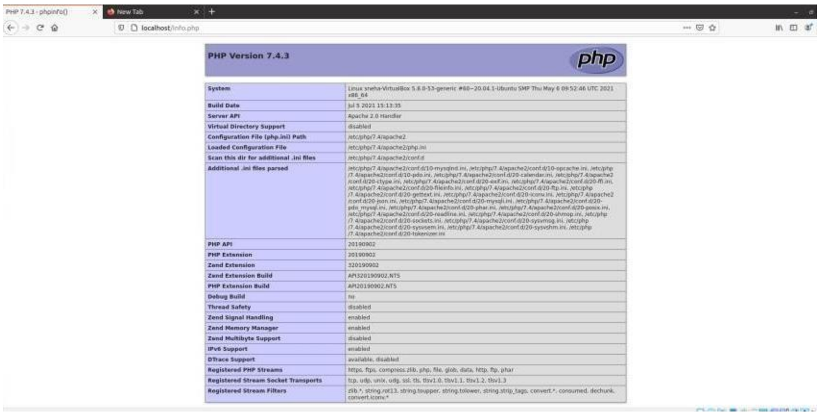
This will open a blank file. Add the following text, which is valid PHP code, inside the file:

```
<?php
phpinfo();
?>
```



```
  GNU nano 4.8                /var/www/html/info.php                    Modified
<?php
phpinfo();
?>

File Name to Write: /var/www/html/info.php
^G Get Help          M-D DOS Format       M-A Append           M-B Backup File
^C Cancel            M-M Mac Format       M-P Prepend          ^T To Files
```

When you are finished, save and close the file. Now you can test whether your web server is able to correctly display content generated by this PHP script. To try this out, visit this page in your web browser. You’ll need your server’s public IP address or domain name again. The address you will want to visit is: http://your_domain/info.php



PHP Version 7.4.3

| | |
|---|---|
| System | Linux sneha-VirtualBox 5.8.0-53-generic #60~20.04.1-Ubuntu SMP Thu May 6 09:52:46 UTC 2021 x86_64 |
| Build Date | Jul 5 2021 15:13:35 |
| Server API | Apache 2.0 Handler |
| Virtual Directory Support | disabled |
| Configuration File (php.ini) Path | /etc/php/7.4/apache2 |
| Loaded Configuration File | /etc/php/7.4/apache2/php.ini |
| Scan this dir for additional .ini files | /etc/php/7.4/apache2/conf.d |
| Additional .ini files parsed | /etc/php/7.4/apache2/conf.d/10-mysqlnd.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini |
| PHP API | 20190902 |
| PHP Extension | 20190902 |
| Zend Extension | 320190902 |
| Zend Extension Build | API320190902,NTS |
| PHP Extension Build | API20190902,NTS |
| Debug Build | no |
| Thread Safety | disabled |
| Zend Signal Handling | enabled |
| Zend Memory Manager | enabled |
| Zend Multibyte Support | disabled |
| IPv6 Support | enabled |
| DTrace Support | available, disabled |
| Registered PHP Streams | https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar |
| Registered Stream Socket Transports | tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3 |
| Registered Stream Filters | zlib.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk, convert.iconv.* |

**RESULT:**Familiarised with Lamp stack

**EXPERIMENT NO:10** **DATE:**

# LARAVEL INSTALLATION

## INTRODUCTION

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities which incorporates the basic features of PHP frameworks like CodeIgniter, Yii and other programming languages like Ruby on Rails. Laravel has a very rich set of features which will boost the speed of web development.

If you are familiar with Core PHP and Advanced PHP, Laravel will make your task easier. It saves a lot time if you are planning to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

## ADVANTAGES OF LARAVEL

Laravel offers you the following advantages, when you are designing a web application based on it −

● The web application becomes more scalable, owing to the Laravel framework. ● Considerable time is saved in designing the web application, since Laravel reuses the components from other framework in developing web application.

● It includes namespaces and interfaces, thus helps to organize and manage resources.

### Composer

Composer is a tool which includes all the dependencies and libraries. It allows a user to create a project with respect to the mentioned framework (for example, those used in Laravel installation). Third party libraries can be installed easily with help of composer. All the dependencies are noted in composer.json file which is placed in the source folder.

### Artisian

The command line interface used in Laravel is called Artisan. It includes a set of commands which assists in building a web application. These commands are incorporated from Symphony framework, resulting in add-on features in Laravel 5.1 (latest version of Laravel).

## FEATURES OF LARAVEL

Laravel offers the following key features which makes it an ideal choice for designing web applications –

- **Modularity**: Laravel provides 20 built in libraries and modules which helps in enhancement of the application. Every module is integrated with Composer dependency manager which eases updates.

- **Testability**: Laravel includes features and helpers which helps in testing through various test cases. This feature helps in maintaining the code as per the requirements.

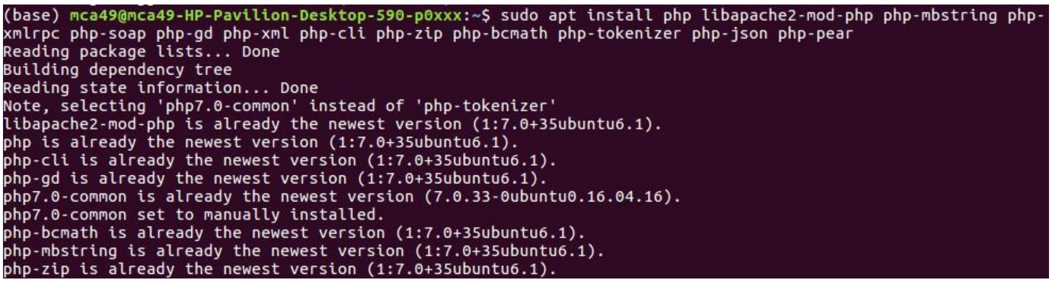
- **Routing**: Laravel provides a flexible approach to the user to define routes in the web application. Routing helps to scale the application in a better way and increases its performance.

**INSTALLATION**

1.Install Apache Web Server: *sudo apt install apache2.*

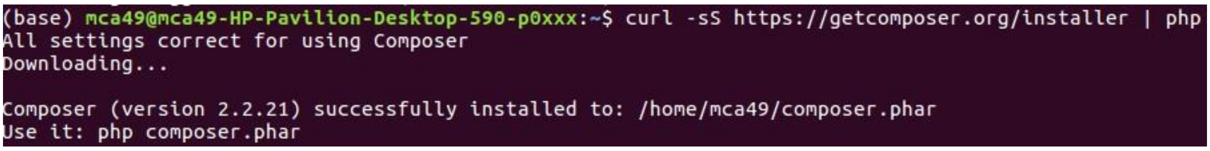
2.Install a Database Manager: *sudo apt install mariadb-server*

3.Install PHP:
*sudo apt install php libapache2-mod-php php-mbstring php-xmlrpc php-soap php-gd php-xml php-cli php-zip php-bcmath php-tokenizer php-json php-pear.*



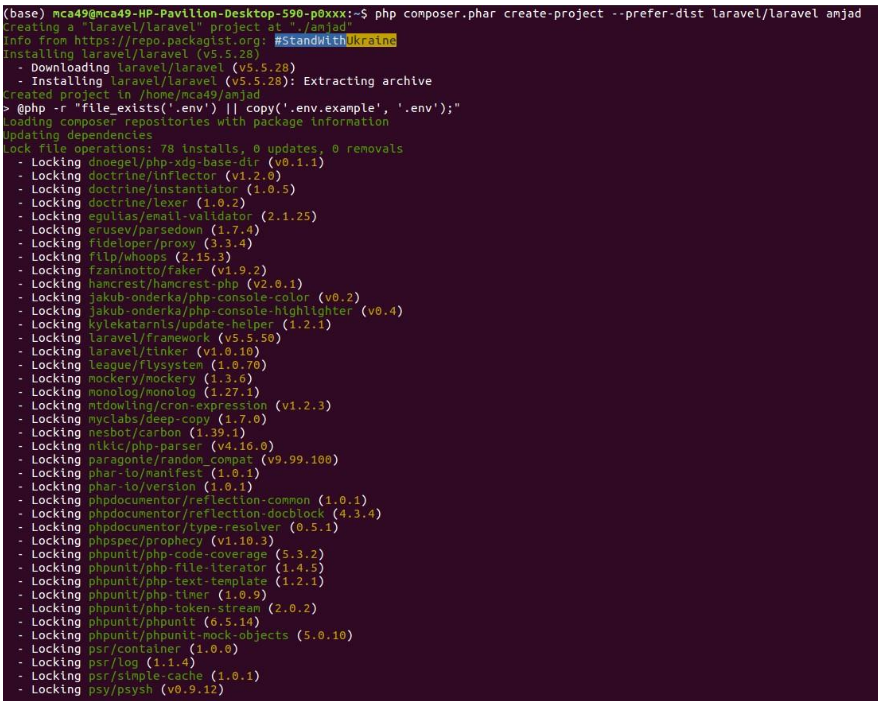
```
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~$ sudo apt install php libapache2-mod-php php-mbstring php-
xmlrpc php-soap php-gd php-xml php-cli php-zip php-bcmath php-tokenizer php-json php-pear
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'php7.0-common' instead of 'php-tokenizer'
libapache2-mod-php is already the newest version (1:7.0+35ubuntu6.1).
php is already the newest version (1:7.0+35ubuntu6.1).
php-cli is already the newest version (1:7.0+35ubuntu6.1).
php-gd is already the newest version (1:7.0+35ubuntu6.1).
php7.0-common is already the newest version (7.0.33-0ubuntu0.16.04.16).
php7.0-common set to manually installed.
php-bcmath is already the newest version (1:7.0+35ubuntu6.1).
php-mbstring is already the newest version (1:7.0+35ubuntu6.1).
php-zip is already the newest version (1:7.0+35ubuntu6.1).
```

4.Install Composer: *curl -sS https://getcomposer.org/installer | php*

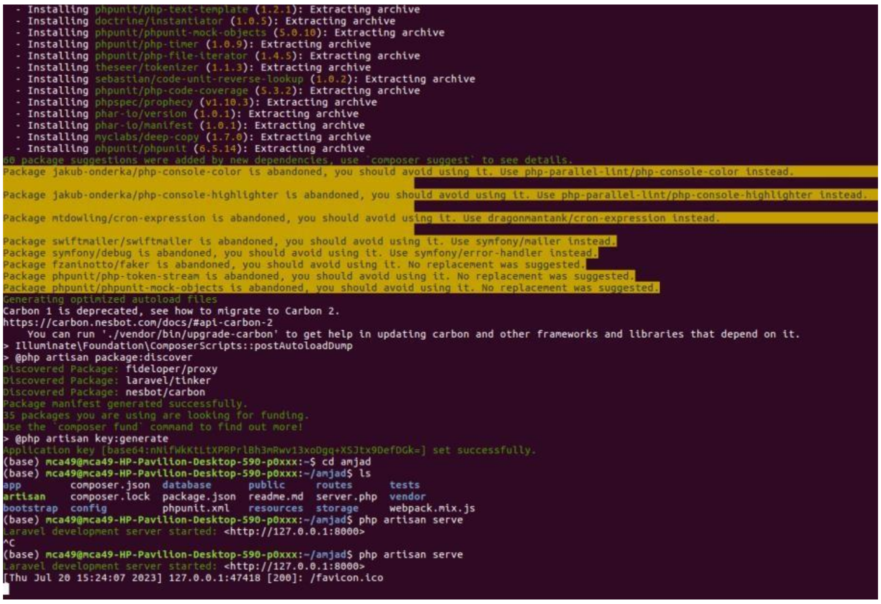


```
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 2.2.21) successfully installed to: /home/mca49/composer.phar
Use it: php composer.phar
```

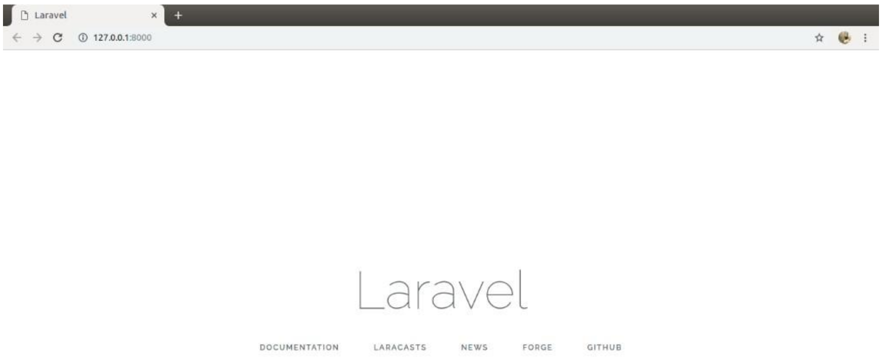
5.Install Laravel on Ubuntu Using Composer. Now we can install Laravel. To do this, run the following command: *composer create-project --prefer-dist laravel/laravel [project_name].*



```
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~$ php composer.phar create-project --prefer-dist laravel/laravel amjad
Creating a "laravel/laravel" project at "./amjad"
Info from https://repo.packagist.org: #StandWithUkraine
Installing laravel/laravel (v5.5.28)
  - Downloading laravel/laravel (v5.5.28)
  - Installing laravel/laravel (v5.5.28): Extracting archive
Created project in /home/mca49/amjad
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 78 installs, 0 updates, 0 removals
  - Locking dnoegel/php-xdg-base-dir (v0.1.1)
  - Locking doctrine/inflector (v1.2.0)
  - Locking doctrine/instantiator (1.0.5)
  - Locking doctrine/lexer (1.0.2)
  - Locking egulias/email-validator (2.1.25)
  - Locking erusev/parsedown (1.7.4)
  - Locking fideloper/proxy (3.3.4)
  - Locking filp/whoops (2.15.3)
  - Locking fzaninotto/faker (v1.9.2)
  - Locking hamcrest/hamcrest-php (v2.0.1)
  - Locking jakub-onderka/php-console-color (v0.2)
  - Locking jakub-onderka/php-console-highlighter (v0.4)
  - Locking kylekatarnls/update-helper (1.2.1)
  - Locking laravel/framework (v5.5.50)
  - Locking laravel/tinker (v1.0.10)
  - Locking league/flysystem (1.0.70)
  - Locking mockery/mockery (1.3.6)
  - Locking monolog/monolog (1.27.1)
  - Locking mtdowling/cron-expression (v1.2.3)
  - Locking myclabs/deep-copy (1.7.0)
  - Locking nesbot/carbon (1.39.1)
  - Locking nikic/php-parser (v4.16.0)
  - Locking paragonie/random_compat (v9.99.100)
  - Locking phar-io/manifest (1.0.1)
  - Locking phar-io/version (1.0.1)
  - Locking phpdocumentor/reflection-common (1.0.1)
  - Locking phpdocumentor/reflection-docblock (4.3.4)
  - Locking phpdocumentor/type-resolver (0.5.1)
  - Locking phpspec/prophecy (v1.10.3)
  - Locking phpunit/php-code-coverage (5.3.2)
  - Locking phpunit/php-file-iterator (1.4.5)
  - Locking phpunit/php-text-template (1.2.1)
  - Locking phpunit/php-timer (1.0.9)
  - Locking phpunit/php-token-stream (2.0.2)
  - Locking phpunit/phpunit (6.5.14)
  - Locking phpunit/phpunit-mock-objects (5.0.10)
  - Locking psr/container (1.0.0)
  - Locking psr/log (1.1.4)
  - Locking psr/simple-cache (1.0.1)
  - Locking psy/psysh (v0.9.12)
```



```
  - Installing phpunit/php-text-template (1.2.1): Extracting archive
  - Installing doctrine/instantiator (1.0.5): Extracting archive
  - Installing phpunit/phpunit-mock-objects (5.0.10): Extracting archive
  - Installing phpunit/php-timer (1.0.9): Extracting archive
  - Installing phpunit/php-file-iterator (1.4.5): Extracting archive
  - Installing theseer/tokenizer (1.1.3): Extracting archive
  - Installing sebastian/code-unit-reverse-lookup (1.0.2): Extracting archive
  - Installing phpunit/php-code-coverage (5.3.2): Extracting archive
  - Installing phpspec/prophecy (v1.10.3): Extracting archive
  - Installing phar-io/version (1.0.1): Extracting archive
  - Installing phar-io/manifest (1.0.1): Extracting archive
  - Installing myclabs/deep-copy (1.7.0): Extracting archive
  - Installing phpunit/phpunit (6.5.14): Extracting archive
60 package suggestions were added by new dependencies, use `composer suggest` to see details.
Package jakub-onderka/php-console-color is abandoned, you should avoid using it. Use php-parallel-lint/php-console-color instead.

Package jakub-onderka/php-console-highlighter is abandoned, you should avoid using it. Use php-parallel-lint/php-console-highlighter instead.

Package mtdowling/cron-expression is abandoned, you should avoid using it. Use dragonmantank/cron-expression instead.

Package swiftmailer/swiftmailer is abandoned, you should avoid using it. Use symfony/mailer instead.
Package symfony/debug is abandoned, you should avoid using it. Use symfony/error-handler instead.
Package fzaninotto/faker is abandoned, you should avoid using it. No replacement was suggested.
Package phpunit/php-token-stream is abandoned, you should avoid using it. No replacement was suggested.
Package phpunit/phpunit-mock-objects is abandoned, you should avoid using it. No replacement was suggested.
Generating optimized autoload files
Carbon 1 is deprecated, see how to migrate to Carbon 2.
https://carbon.nesbot.com/docs/#api-carbon-2
    You can run './vendor/bin/upgrade-carbon' to get help in updating carbon and other frameworks and libraries that depend on it.
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover
Discovered Package: fideloper/proxy
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Package manifest generated successfully.
35 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan key:generate
Application key [base64:nNifWkKtLtXPRPrlBh3mRwv13xoDgq+XSJtx9DefDGk=] set successfully.
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~$ cd amjad
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~/amjad$ ls
app        composer.json  database      public     routes      tests
artisan    composer.lock  package.json  readme.md  server.php  vendor
bootstrap  config         phpunit.xml   resources  storage     webpack.mix.js
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~/amjad$ php artisan serve
Laravel development server started: <http://127.0.0.1:8000>
^C
(base) mca49@mca49-HP-Pavilion-Desktop-590-p0xxx:~/amjad$ php artisan serve
Laravel development server started: <http://127.0.0.1:8000>
[Thu Jul 20 15:24:07 2023] 127.0.0.1:47418 [200]: /favicon.ico
```



**RESULT:**Familiarised with Laravel

## EXPERIMENT NO:7 DATE:

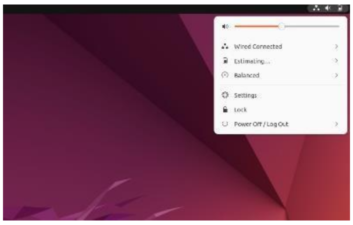
# NETWORKING

### STATIC IP

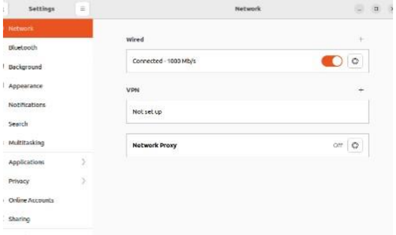
- A static IP address is a fixed IP address that is manually assigned to a device by the network administrator.
- Once assigned, the device will always have the same IP address, which makes it easy to manage and configure.
- Static IP addresses are commonly used for servers, printers, and other network devices that require a fixed IP address for remote access and management.
- The disadvantage of using a static IP address is that it can be more difficult to manage in large networks and can lead to IP address conflicts if not properly managed.

### Setting STATIC IP

1. Click on the top right network icon and select settings of the network interface you wish to configure to use a static IP address on Ubuntu.



2.Click on the settings icon to start IP address configuration.

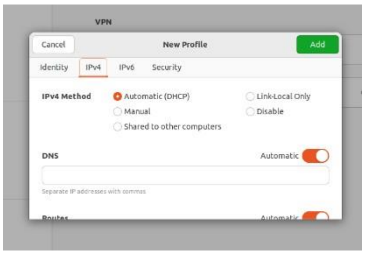


Connected - 1000 Mb/s

VPN: Not set up

Network Proxy: Off

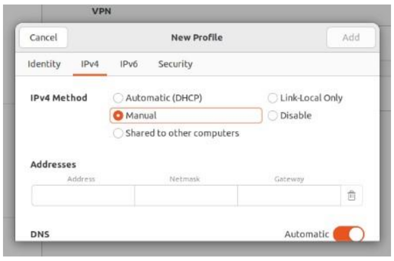
3. Select IPv4 tab.



IPv4 Method: ● Automatic (DHCP) ○ Manual ○ Shared to other computers ○ Link-Local Only ○ Disable

DNS: Automatic

4.Select manual and enter your desired IP address, netmask, gateway and DNS settings.



IPv4 Method: ○ Automatic (DHCP) ● Manual ○ Shared to other computers ○ Link-Local Only ○ Disable

DNS: Automatic ●

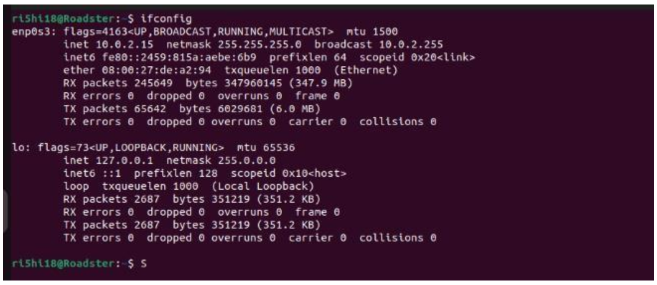
5.Once ready click Apply button.
6.Turn OFF and ON switch to apply your new network static IP configuration settings.
7.Click on the network settings icon once again to confirm your new static IP address settings.

## DYNAMIC IP

- A dynamic IP address is an IP address that is automatically assigned to a device by a DHCP server when it connects to the network.
- The IP address is assigned from a pool of available IP addresses, and the device will receive a different IP address each time it connects to the network.
- Dynamic IP addressing is commonly used in home and small office networks, as well as in large networks where managing static IP addresses would be impractical.
- The advantage of using dynamic IP addressing is that it allows for more efficient use of IP addresses, as well as easier management of IP addresses in large networks. However, it can be more difficult to configure remote access to devices with dynamic IP addresses.

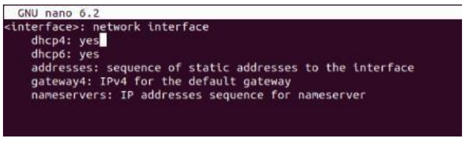
## Setting DYNAMIC IP

1. type the command ‘sudo nano /etc/netplan/01-network-manager-all.yaml ’ in the ubuntu terminal.



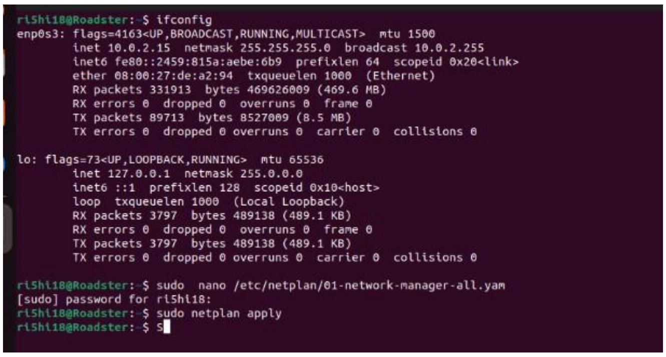
```
ri5hi18@Roadster:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::2459:815a:aebe:6b9  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:de:a2:94  txqueuelen 1000  (Ethernet)
        RX packets 245649  bytes 347960145 (347.9 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 65642  bytes 6029681 (6.0 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2687  bytes 351219 (351.2 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2687  bytes 351219 (351.2 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ri5hi18@Roadster:~$ S
```

2. Now find the name of the network interface you want to configure and insert the following lines:
1. dhcp4: yes
2. dhcp6: yes



```
  GNU nano 6.2
<interface>: network interface
    dhcp4: yes
    dhcp6: yes
    addresses: sequence of static addresses to the interface
    gateway4: IPv4 for the default gateway
    nameservers: IP addresses sequence for nameserver
```

3. Apply the changes with the command ‘ sudo netplan apply’ command.



```
ri5hi18@Roadster:~$ ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::2459:815a:aebe:6b9  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:de:a2:94  txqueuelen 1000  (Ethernet)
        RX packets 331913  bytes 469626009 (469.6 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 89713  bytes 8527009 (8.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 3797  bytes 489138 (489.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3797  bytes 489138 (489.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ri5hi18@Roadster:~$ sudo  nano /etc/netplan/01-network-manager-all.yam
[sudo] password for ri5hi18:
ri5hi18@Roadster:~$ sudo netplan apply
ri5hi18@Roadster:~$ S
```

**SUBNETS:**

The process of dividing a network into smaller network sections is called subnetting. This can be useful for many different purposes and helps isolate groups of hosts from each other to deal with them more easily. By default, each network has only one subnet, which contains all of the host addresses defined within. A netmask is basically a specification of the amount of address bits that are used for the network portion. A subnet mask is another netmask within used to further divide the network.

Each bit of the address that is considered significant for describing the network should be represented as a “1” in the netmask. For instance, the address we discussed above, 192.168.0.15 can be expressed like this, in binary:

1100 0000 - 1010 1000 - 0000 0000 - 0000 1111

As we described above, the network portion for class C addresses is the first 3 octets, or the first 24 bits. Since these are the significant bits that we want to preserve, the netmask would be:

1111 1111 - 1111 1111 - 1111 1111 - 0000 0000

This can be written in the normal IPv4 format as 255.255.255.0. Any bit that is a “0” in the binary representation of the netmask is considered part of the host portion of the address and can be variable. The bits that are “1” are static, however, for the network or subnetwork that is being discussed. We determine the network portion of the address by applying a bitwise AND operation to between the address and the netmask. A bitwise AND operation will save the networking portion of the address and discard the host portion. The result of this on our above example that represents our network is:

1100 0000 - 1010 1000 - 0000 0000 - 0000 0000

This can be expressed as 192.168.0.0. The host specification is then the difference between these original value and the host portion. In our case, the host is 0000 1111 or 15. The idea of subnetting is to take a portion of the host space of an address, and use it as an additional networking specification to divide the address space again. For instance, a netmask of 255.255.255.0 as we saw above leaves us with 254 hosts in the network (you cannot end in 0 or 255 because these are reserved).

So, continuing with our example, the networking portion is: 1100 0000 – 1010
1000 - 0000 0000

The host portion is:

0000 1111

We can use the first bit of our host to designate a subnetwork. We can do this by adjusting the subnet mask from this:

1111 1111 - 1111 1111 - 1111 1111 - 0000 0000

To this:

1111 1111 - 1111 1111 - 1111 1111 - 1000 0000

In traditional IPv4 notation, this would be expressed as 192.168.0.128. What we have done here is to designate the first bit of the last octet as significant in addressing the network. This effectively produces two subnetworks. The first subnetwork is from 192.168.0.1 to 192.168.0.127. The second subnetwork contains the hosts 192.168.0.129 to 192.168.0.255.

**CIDR NOTATION:**

A system called Classless Inter-Domain Routing, or CIDR, was developed as an alternative to traditional subnetting. For example, we could express the idea that the IP address 192.168.0.15 is associated with the netmask 255.255.255.0 by using the CIDR notation of 192.168.0.15/24. This means that the first 24 bits of the IP address given are considered significant for the network routing.

This allows us some interesting possibilities. We can use these to reference “supernets”. In this case, we mean a more inclusive address range that is not possible with a traditional subnet mask. For instance, in a class C network, like above, we could not combine the addresses from the networks 192.168.0.0 and 192.168.1.0 because the netmask for class C addresses is
255.255.255.0. However, using CIDR notation, we can combine these blocks by referencing this chunk as 192.168.0.0/23. This specifies that there are 23 bits used for the network

portion that we are referring to. So the first network (192.168.0.0) could be represented like this in binary:

1100 0000 - 1010 1000 - 0000 0000 - 0000 0000

While the second network (192.168.1.0) would be like this:

1100 0000 - 1010 1000 - 0000 0001 - 0000 0000

The CIDR address we specified indicates that the first 23 bits are used for the network block we are referencing. This is equivalent to a netmask of 255.255.254.0, or:

1111 1111 - 1111 1111 - 1111 1110 - 0000 0000

As you can see, with this block the 24th bit can be either 0 or 1 and it will still match, because the network block only cares about the first 23 digits. CIDR allows us more control over addressing continuous blocks of IP addresses. This is much more useful than the subnetting we talked about originally.

## CONCEPT OF SUBNET MASK

The subnet mask is used by the TCP/IP protocol to determine whether a host is on the local subnet or on a remote network. In TCP/IP, the parts of the IP address that are used as the network and host addresses aren't fixed. Unless you have more information, the network and host addresses above can't be determined. This information is supplied in another 32-bit number called a subnet mask. The subnet mask is 255.255.255.0 in this example. It isn't obvious what this number means unless you know 255 in binary notation equals 11111111. So, the subnet mask is 11111111.11111111.11111111.00000000.

Lining up the IP address and the subnet mask together, the network, and host portions of the address can be separated:

11000000.10101000.01111011.10000100 - IP address (192.168.123.132)

11111111.11111111.11111111.00000000 - Subnet mask (255.255.255.0)

The first 24 bits (the number of ones in the subnet mask) are identified as the network address. The last 8 bits (the number of remaining zeros in the subnet mask) are identified as the host address. It gives you the following addresses:

11000000.10101000.01111011.00000000 - Network address (192.168.123.0)

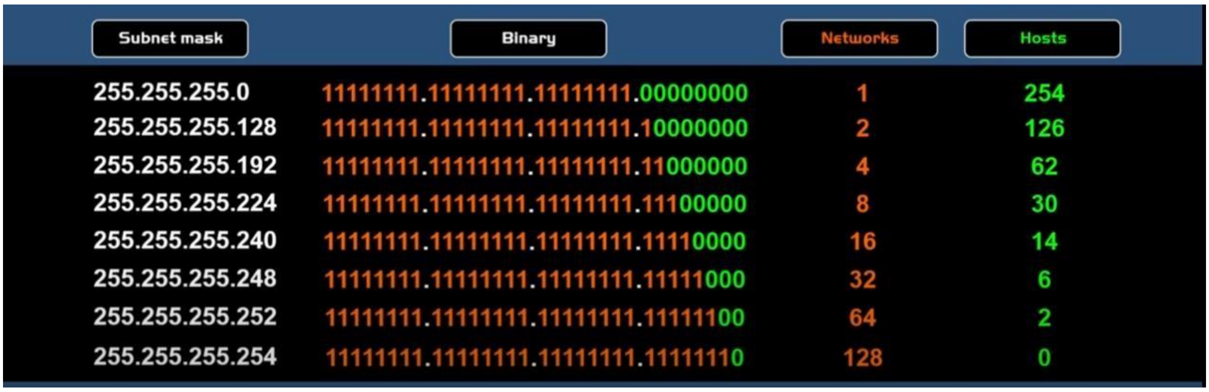
00000000.00000000.00000000.10000100 - Host address (000.000.000.132)

So now you know, for this example using a 255.255.255.0 subnet mask, that the network ID is 192.168.123.0, and the host address is 0.0.0.132. When a packet arrives on the 192.168.123.0 subnet (from the local subnet or a remote network), and it has a destination address of 192.168.123.132, your computer will receive it from the network and process it.Almost all decimal subnet masks convert to binary numbers that are all ones on the left and all zeros on the right.

Some other common subnet masks are:

Decimal Binary 255.255.255.192  1111111.11111111.1111111.11000000
255.255.255.224     1111111.11111111.1111111.11100000

Internet RFC 1878 (available from InterNIC-Public Information Regarding Internet Domain Name Registration Services) describes the valid subnets and subnet masks that can be used on TCP/IP networks.



| Subnet mask | Binary | Networks | Hosts |
|---|---|---|---|
| 255.255.255.0 | 11111111.11111111.11111111.00000000 | 1 | 254 |
| 255.255.255.128 | 11111111.11111111.11111111.10000000 | 2 | 126 |
| 255.255.255.192 | 11111111.11111111.11111111.11000000 | 4 | 62 |
| 255.255.255.224 | 11111111.11111111.11111111.11100000 | 8 | 30 |
| 255.255.255.240 | 11111111.11111111.11111111.11110000 | 16 | 14 |
| 255.255.255.248 | 11111111.11111111.11111111.11111000 | 32 | 6 |
| 255.255.255.252 | 11111111.11111111.11111111.11111100 | 64 | 2 |
| 255.255.255.254 | 11111111.11111111.11111111.11111110 | 128 | 0 |

**SUBNET USES:**

- Efficient use of IP addresses:
  Since it divides a large network into smaller subnets,this reduces the number of hosts on each subnet, which in turn reduces network traffic and increases network efficiency.
- Network segmentation:
  Subnet masks can be used to segment a network into smaller, more manageable subnets. This can improve network performance, as traffic is limited to each subnet.
- Improved security:
  Subnet masks can be used to improve network security by separating sensitive data and applications from less secure areas of the network. This makes it harder for unauthorized users to access sensitive information.
- Routing:

Subnet masks are used by routers to determine the network address of a destination IP address. This allows routers to forward packets to the correct subnet, improving network performance and efficiency.

## SETTING UP A FIREWALL FOR LAN

### UFW - Uncomplicated Firewall

Uncomplicated Firewall (UFW) is a user-friendly command-line tool for managing firewall rules on Ubuntu and Debian-based Linux systems. It is designed to simplify the process of configuring a firewall by providing a simple and easy-to-use interface.

With UFW, you can create rules to allow or deny incoming and outgoing traffic based on various criteria such as IP addresses, ports, and protocols. It also supports configuring rules for specific network interfaces.

### Setting up of firewall using UFW

Before we start, we need to ensure that UFC is properly installed in our system
To check we use the command : sudo apt-get update followed by: *sudo apt-get install ufw*

Status : check the UFW status by using the command : *sudo ufw status*

Enable : sudo ufw enable

Configure UFW rules : Now you can configure UFW rules to allow or deny incoming and outgoing traffic.

For Example : To allow incoming SSH connections, run the following command : *sudo ufw allow ssh*

Alternatively, you could you can also specify the port number or the protocol. For example, to allow incoming HTTP traffic, run the following command: *sudo ufw allow 80/tcp.*

Status:

To disable the UFW firewall on your Ubuntu system, you can use the following command in the terminal: *sudo ufw disable*

**RESULT:**Familiarised with Networking

## EXPERIMENT NO:8 DATE:

## NETWORK AND PACKET ANALYSIS

**AIM:**Analyzing network packet stream using tcpdump and wireshark

**WIRESHARK**

- A network packet protocol analyzer
- A network packet analyzer will try to capture network packets and try to display that packet data in as detail as possible.
- One of the best open-source packet analyzers available today for UNIX and Windows.

**Some intended purposes:**

- Network administrators use it to troubleshoot network problems
- Network security engineers use it to examine security problems
- QA engineers use it to verify network applications
- Developers use it to debug protocol implementations
- People use it to learn network protocol internals

**Features:**

- ✦ Available for UNIX and Windows.
- ✦ Capture live packet data from a network interface.
- ✦ Open files containing packet data captured with tcpdump/WinDump, Wireshark, and many 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.

**INSTALLATION**

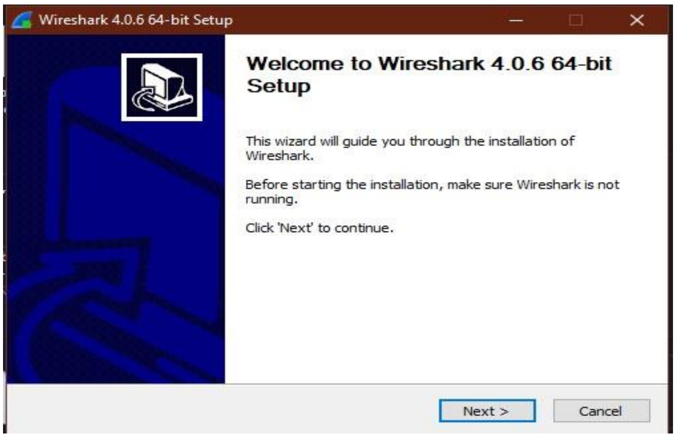
**1.Download Wireshark**

Visit the official website and download Wireshark for 64-bit Windows systems.

Download Wireshark Now →
The world's most popular network protocol analyzer
Get started with Wireshark today and see why it is the standard across many commercial and non-profit enterprises.
Get started
Wireshark 4.0 Overview

## 2.Install Wireshark

2.1 Then double click on local downloaded installer to start the installation. It will first show you below setup wizard asking to make sure Wireshark is not running. Click Next to Continue.



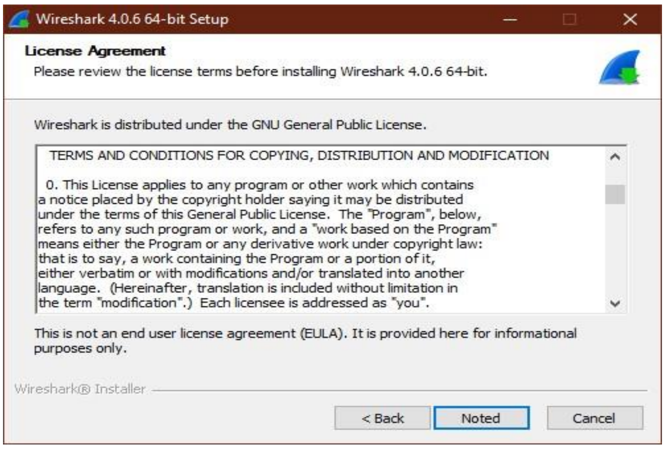
Welcome to Wireshark 4.0.6 64-bit Setup

This wizard will guide you through the installation of Wireshark.

Before starting the installation, make sure Wireshark is not running.

Click 'Next' to continue.

2.2 You will see below License Agreement. Please go through it and review all the License terms under this agreement before installing Wireshark. Click Noted to continue.



License Agreement

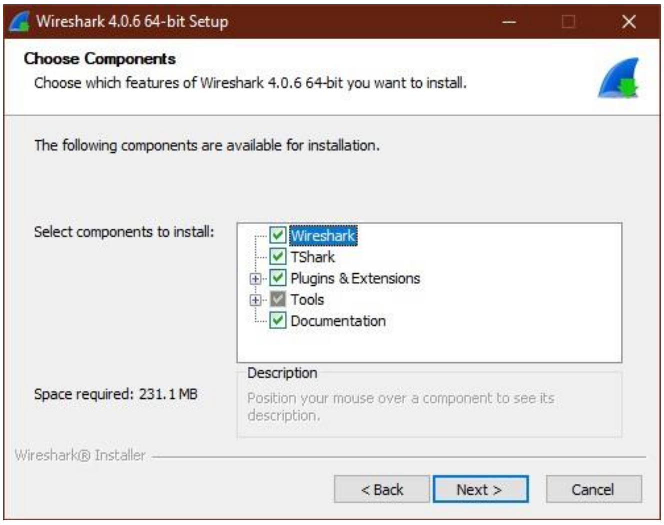
Please review the license terms before installing Wireshark 4.0.6 64-bit.

Wireshark is distributed under the GNU General Public License.

```
 TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

  0. This License applies to any program or other work which contains
a notice placed by the copyright holder saying it may be distributed
under the terms of this General Public License.  The "Program", below,
refers to any such program or work, and a "work based on the Program"
means either the Program or any derivative work under copyright law:
that is to say, a work containing the Program or a portion of it,
either verbatim or with modifications and/or translated into another
language.  (Hereinafter, translation is included without limitation in
the term "modification".)  Each licensee is addressed as "you".
```

This is not an end user license agreement (EULA). It is provided here for informational purposes only.

2.3 You can select all the Wireshark features to install. Below are the main features available to install. You can select all the required features and then click on Next to continue.



**Choose Components**

Choose which features of Wireshark 4.0.6 64-bit you want to install.

The following components are available for installation.

Select components to install:

- ☑ Wireshark
- ☑ TShark
- ☑ Plugins & Extensions
- ☑ Tools
- ☑ Documentation

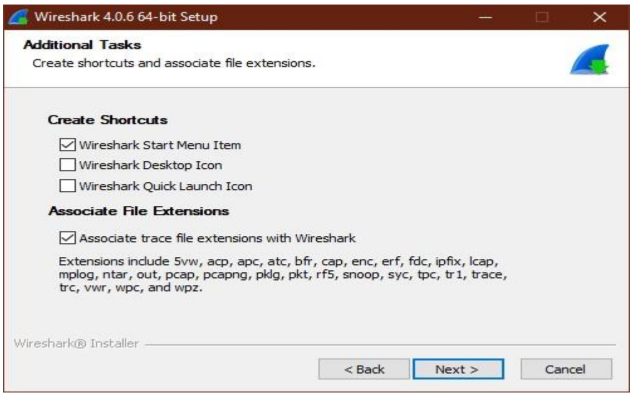
Space required: 231.1 MB

Description

Position your mouse over a component to see its description.

Wireshark® Installer

2.4 In additional tasks, you can choose to create shortcuts and associate file extensions from below. Once selected, Click Next to continue.



**Additional Tasks**

Create shortcuts and associate file extensions.

**Create Shortcuts**

☑ Wireshark Start Menu Item
☐ Wireshark Desktop Icon
☐ Wireshark Quick Launch Icon

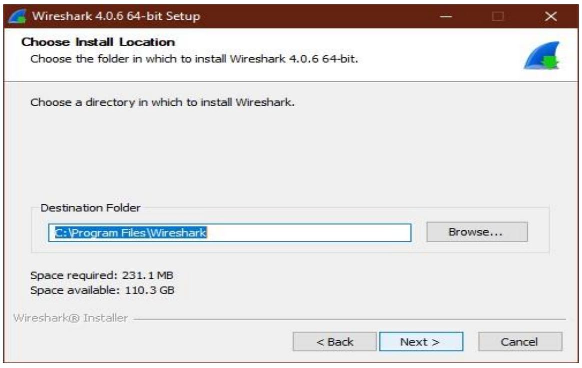
**Associate File Extensions**

☑ Associate trace file extensions with Wireshark

Extensions include 5vw, acp, apc, atc, bfr, cap, enc, erf, fdc, ipfix, lcap, mplog, ntar, out, pcap, pcapng, pklg, pkt, rf5, snoop, syc, tpc, tr1, trace, trc, vwr, wpc, and wpz.

Wireshark® Installer

2.5 You need to choose the destination folder by browsing to the location where you need to install wireshark. By default, it will install under C:\Program Files\Wireshark folder as shown below. Once chosen, Click on Next to proceed.



**Choose Install Location**

Choose the folder in which to install Wireshark 4.0.6 64-bit.

Choose a directory in which to install Wireshark.

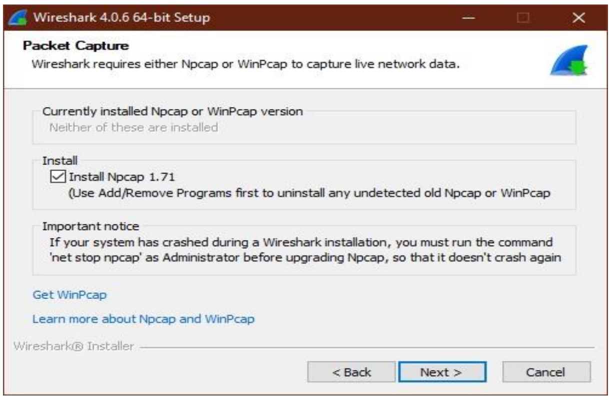
Destination Folder

C:\Program Files\Wireshark

Space required: 231.1 MB
Space available: 110.3 GB

Wireshark® Installer

2.6 To capture live network data, Wireshark requires either Npcap or WinPcap to be installed or else by default it will install Npcap in your System. If you would like to install this program then just click on Next. Otherwise, you need to unselect and then click on Next.


Wireshark 4.0.6 64-bit Setup

**Packet Capture**
Wireshark requires either Npcap or WinPcap to capture live network data.

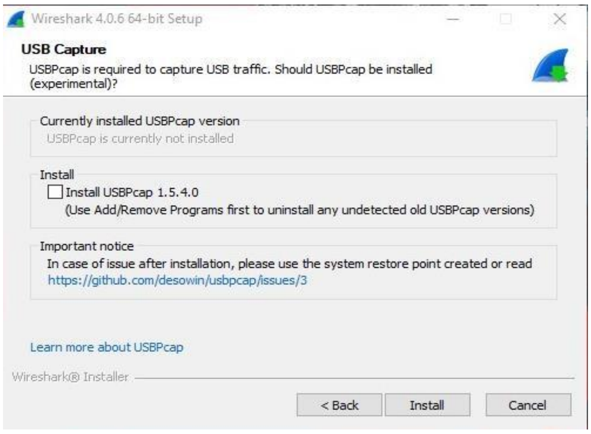
Currently installed Npcap or WinPcap version
Neither of these are installed

Install
☑ Install Npcap 1.71
(Use Add/Remove Programs first to uninstall any undetected old Npcap or WinPcap

Important notice
If your system has crashed during a Wireshark installation, you must run the command 'net stop npcap' as Administrator before upgrading Npcap, so that it doesn't crash again

Get WinPcap
Learn more about Npcap and WinPcap

2.7 Similarly, for capturing USB traffic, wireshark needs to install USBPcap tool in your System. It won't be selected by default, so you need to select it manually in case you want to install this tool. Then Click on Install.


Wireshark 4.0.6 64-bit Setup

**USB Capture**
USBPcap is required to capture USB traffic. Should USBPcap be installed (experimental)?

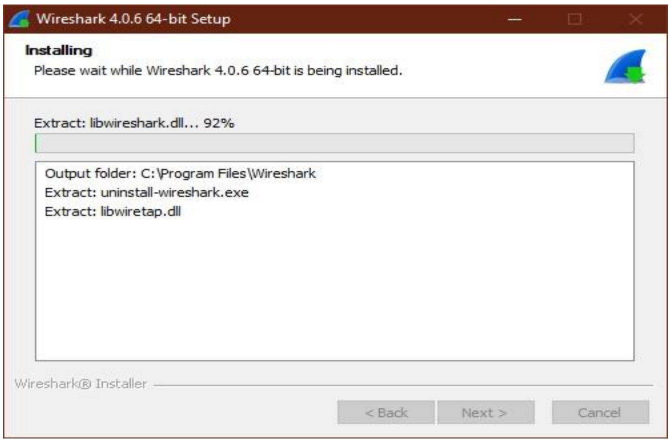
Currently installed USBPcap version
USBPcap is currently not installed

Install
□ Install USBPcap 1.5.4.0
(Use Add/Remove Programs first to uninstall any undetected old USBPcap versions)

Important notice
In case of issue after installation, please use the system restore point created or read https://github.com/desowin/usbpcap/issues/3

Learn more about USBPcap

2.8 You can see that Wireshark installation will be started as shown below.

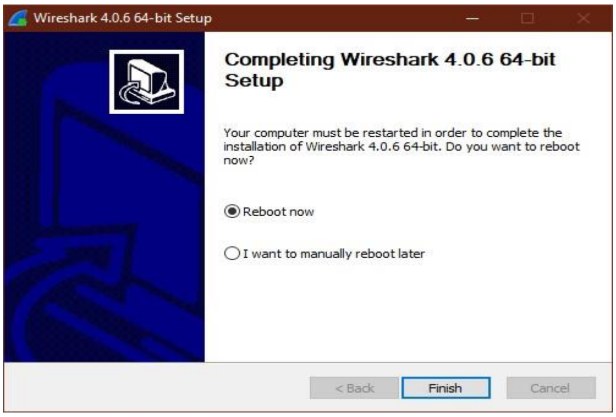

Wireshark 4.0.6 64-bit Setup

**Installing**
Please wait while Wireshark 4.0.6 64-bit is being installed.

Extract: libwireshark.dll... 92%

```
Output folder: C:\Program Files\Wireshark
Extract: uninstall-wireshark.exe
Extract: libwiretap.dll
```

2.9 Finally, you need to click on Finish to exit the wireshark setup wizard.



Completing Wireshark 4.0.6 64-bit Setup

Your computer must be restarted in order to complete the installation of Wireshark 4.0.6 64-bit. Do you want to reboot now?

● Reboot now

○ I want to manually reboot later

## TCP DUMP

- Tcpdump is a command-line packet analysis tool. Much like Wireshark, you can use Tcpdump to capture and analyze packets, troubleshoot connection issues, and look for potential security issues on a network.
- Tcpdump is a portable command-line utility that can be used even when a GUI is unavailable and when Wireshark is not installed.

## Some intended purpose

- It can read packets from a network interface card or from a previously created saved packet file. tcpdump can write packets to standard output or a file.
- It is also possible to use tcpdump for the specific purpose of intercepting and displaying the communications of another user or computer.

## Disadvantages

- Disadvantages of TCPdump are that the packets with an invalid checksum are ignored. So it is not very helpful in handling network packets where checksums are invalid. It needs specialized hardware. It only shows data that was meant to be sent but not its inner details.
- It doesn’t have the capability to inform about any fake IP address in the packet.

- Disadvantages of TCPdump are that the packets with an invalid checksum are ignored. So it is not very helpful in handling network packets where checksums are invalid. It needs specialized hardware. It only shows data that was meant to be sent but not its inner details.
- It doesn’t have the capability to inform about any fake IP address in the packet.

## INSTALLATION

1.Update the system: sudo apt-get update

2.Install TCPdump in the system: *sudo apt-get install tcpdump*
3.Check TCPdump version: *sudo tcpdump –version*

**RESULT:**Familiarised network packet stream using tcpdump and wireshark