## **Installation of LAMP(Apache) stack a**nd Magento2.4.5

### **Run Update Command**

**sudo apt update -y**

### **Install Apache2**

**sudo apt install apache2 -y**

### **Start apache2 and check status**

**sudo systemctl start apache2**

**sudo systemctl status apache2**

### **Open Browser and Type**

**http://server\_name**

**OR**

**http://your-domain.com**

### **Add php Repository**

**sudo apt-get install software-properties-common**

**sudo add-apt-repository ppa:ondrej/php**

**sudo apt-get update**

### **Install php8.1**

sudo apt-get install php8.1 libapache2-mod-php8.1 php8.1-{bcmath,gettext,mysql,curl,gd,mbstring,xml,xmlrpc,soap,intl,zip} -y

### Check php

sudo nano /var/www/html/info.php

<?php

phpinfo();

?>

### Change php.ini

### sudo nano /etc/php/8.1/fpm/php.ini

max\_execution\_time = 1800

memory\_limit = 1G

zlib.output\_compression = 128M

### sudo nano /etc/php/8.1/cli/php.ini

max\_execution\_time = 1800

memory\_limit = 1G

zlib.output\_compression = 128M

#open Browser :- http://localhost/info.php

### **Install mysql-server**

sudo apt install mysql-server

#wget https://dev.mysql.com/get/mysql-apt-config\_0.8.12-1\_all.deb

#dpkg -i mysql-apt-config\_0.8.12-1\_all.deb

##Select Ubuntu Bionic and MySQL-Server- 8.0

### #Update the MySQL packages repository.

#apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 467B942D3A79BD29

#apt-get update

### #Verify the Ubuntu policy for the MySQL package installation.

#apt-cache policy mysql-server

#sudo apt install mysql-server

### **Start MySQL and check status**

**sudo systemctl start mysql**

**sudo systemctl status mysql**

### Create Magento password user and database

mysql -u root -p

Password: enter your root password

create user ‘magento’@’localhost’ identified with mysql\_native\_password by ‘password’;

create database magento;

Grant all privileges on magento.\* to ‘magento’@’localhost’ with grant option;

Flush privileges;

### Apache Mod Enable

sudo apt install libapache2-mod-security2 -y

sudo a2enmod deflate

sudo a2enmod security2

sudo a2enmod rewrite

sudo a2enmod headers

sudo a2enmod expires

sudo a2enmod ssl

sudo systemctl restart apache2

### Make changes same as below in /etc/apache2/apache2.conf

<Directory /var/www/>

Options Indexes FollowSymLinks

AllowOverride all

Require all granted

</Directory>

### Download elastic Search

wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.17.0-amd64.deb

chmod +x elasticsearch-7.17.0-amd64.deb

sudo dpkg -i elasticsearch-7.17.0-amd64.deb  
sudo systemctl daemon-reload

sudo systemctl enable elasticsearch.service

sudo systemctl start elasticsearch.service

### Download Composer

wget https://getcomposer.org/download/2.2.18/composer.phar

mv composer.phar composer

mv composer /usr/local/bin

cd /usr/local/bin

sudo chmod -R 775 composer

Or

curl -sS https://getcomposer.org/installer | sudo php -- --install-dir=/usr/local/bin --filename=composer --version=2.1.10

composer

### Download magento2.4.5

wget https://github.com/magento/magento2/archive/refs/tags/2.4.5.tar.gz

tar -xf 2.4.5.tar.gz

mv magento2-2.4.5 magento245

sudo mv magento245 /var/www/html/

cd /var/www/html/magento245

composer install

find var generated vendor pub/static pub/media app/etc -type f -exec chmod g+w {} +

find var generated vendor pub/static pub/media app/etc -type d -exec chmod g+ws {} +

chown -R :www-data .

#find var generated vendor pub/static pub/media app/etc -type f -exec chmod g+w {} +

#find var generated vendor pub/static pub/media app/etc -type d -exec chmod g+ws {} +

chmod u+x bin/magento

### Get Admin URL

php bin/magento info:adminuri

<http://ip_or_domain/admin_url>

### Change Admin URL

sudo nano /var/www/html/magento245/app/etc/env.php

#Find Below line and make changes

'frontName' => 'your\_admin\_url'

### OpenSSL Configuration

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/magento2lemp.key -out /etc/ssl/certs/magento2lemp.crt

### Output

Country Name (2 letter code) [AU]:IN

State or Province Name (full name) [Some-State]:Uttar Pradesh

Locality Name (eg, city) []:Lucknow

Organization Name (eg, company) [Internet Widgits Pty Ltd]:Cedcoss

Organizational Unit Name (eg, section) []:AWS

Common Name (e.g. server FQDN or YOUR\_DOMAIN name) []:192.168.2.182

Email Address []:admin@localhost.com

### Edit ssl configuration file

sudo nano /etc/apache2/conf-available/ssl-params.conf

SSLCipherSuite EECDH+AESGCM:EDH+AESGCM

# Requires Apache 2.4.36 & OpenSSL 1.1.1

SSLProtocol -all +TLSv1.3 +TLSv1.2

SSLOpenSSLConfCmd Curves X25519:secp521r1:secp384r1:prime256v1

# Older versions

# SSLProtocol All -SSLv2 -SSLv3 -TLSv1 -TLSv1.1

SSLHonorCipherOrder On

#Header always set Strict-Transport-Security "max-age=63072000; includeSubDomains; preload"

Header always set X-Frame-Options DENY

Header always set X-Content-Type-Options nosniff

# Requires Apache >= 2.4

SSLCompression off

SSLUseStapling on

SSLStaplingCache "shmcb:logs/stapling-cache(150000)"

# Requires Apache >= 2.4.11

SSLSessionTickets Off

### Make backup file for default ssl

sudo cp /etc/apache2/sites-available/default-ssl.conf /etc/apache2/sites-available/default-ssl.conf.bak\

sudo nano /etc/apache2/sites-available/default-ssl.conf

<IfModule mod\_ssl.c>

<VirtualHost \_default\_:443>

ServerAdmin your\_email@domain.com #your domain

ServerName Server\_name\_or\_domian\_name

DocumentRoot /var/www/html #Change Your Document root

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

SSLEngine on

SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem

SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

<FilesMatch "\.(cgi|shtml|phtml|php)$">

SSLOptions +StdEnvVars

</FilesMatch>

<Directory /usr/lib/cgi-bin>

SSLOptions +StdEnvVars

</Directory>

</VirtualHost>

</IfModule>

sudo nano /etc/apache2/sites-available/000-default.conf

<VirtualHost \*:80>

. . .

Redirect "/" "https://your\_domain\_or\_IP/"

. . .

</VirtualHost>

### UFW

sudo ufw app list

sudo ufw status

sudo ufw allow 'Apache Full'

sudo ufw allow ‘Apache Secure’

sudo ufw allow ‘OpenSSH’

sudo ufw delete allow 'Apache'

sudo ufw allow ssh

sudo a2enmod ssl

sudo a2enmod headers

sudo a2ensite default-ssl

sudo a2enconf ssl-params

sudo apache2ctl configtest

sudo systemctl restart apache2

sudo nano /etc/apache2/sites-available/000-default.conf

<VirtualHost \*:80>

. . .

Redirect permanent "/" "https://your\_domain\_or\_IP/"

. . .

</VirtualHost>

sudo apache2ctl configtest

sudo systemctl restart apache2

### Install Magento2

php bin/magento setup:install --base-url=http://192.168.0.111/magento246/pub --db-host=localhost --db-name=magento --db-user=magento --db-password=password --admin-firstname=admin --admin-lastname=admin --admin-email=admin@gmail.com --admin-user=admin --admin-password=admin@123 --language=en\_US --currency=USD --timezone=America/Chicago --use-rewrites=1

### If CSS Error Occurred

sudo chown -R www-data:www-data \*

php bin/magento indexer:reindex

php bin/magento setup:upgrade

php bin/magento setup:static-content:deploy -f

php bin/magento cache:flush

php bin/magento cache:c

### Setup Di-Compile Command

php bin/magento s:d:c

### If Interal Error 500 Occurs

sudo chmod -R 777 var/ pub/ generated/

### For Magento2.4.5-P1

bin/magento module:disable Magento\_TwoFactorAuth

php bin/magento admin:user:unlock admin

============================================================================

## Configure Varnish in Magento2.4.5 with SSL

sudo apt update

### Register Varnish Cache 7 repo on Ubuntu 22.04|20.04|18.04

sudo apt install debian-archive-keyring curl gnupg apt-transport-https -y

curl -fsSL https://packagecloud.io/varnishcache/varnish70/gpgkey|sudo gpg --dearmor -o /etc/apt/trusted.gpg.d/varnish.gpg

### Ubuntu 20.04 / 18.04:

. /etc/os-release

sudo tee /etc/apt/sources.list.d/varnishcache\_varnish70.list > /dev/null <<-EOF

deb https://packagecloud.io/varnishcache/varnish70/$ID/ $VERSION\_CODENAME main

deb-src https://packagecloud.io/varnishcache/varnish70/$ID/ $VERSION\_CODENAME main

EOF

### Ubuntu 22.04:

sudo tee /etc/apt/sources.list.d/varnishcache\_varnish70.list > /dev/null <<-EOF

deb https://packagecloud.io/varnishcache/varnish70/ubuntu/ focal main

deb-src https://packagecloud.io/varnishcache/varnish70/ubuntu/ focal main

EOF

sudo apt update

### Install Varnish Cache 7 on Ubuntu 22.04|20.04|18.04

sudo apt install varnish

### Create Systemd service file for Varnish Cache 7

sudo cp /lib/systemd/system/varnish.service /etc/systemd/system/

### Configure the listening port and cache size

[Unit]

Description=Varnish Cache, a high-performance HTTP accelerator

After=network-online.target nss-lookup.target

[Service]

Type=forking

KillMode=process

# Maximum number of open files (for ulimit -n)

LimitNOFILE=131072

# Locked shared memory - should suffice to lock the shared memory log

# (varnishd -l argument)

# Default log size is 80MB vsl + 1M vsm + header -> 82MB

# unit is bytes

LimitMEMLOCK=85983232

# Enable this to avoid "fork failed" on reload.

TasksMax=infinity

# Maximum size of the corefile.

LimitCORE=infinity

ExecStart=/usr/sbin/varnishd \

-a :80 \

-a localhost:8443,PROXY \

-p feature=+http2 \

-f /etc/varnish/default.vcl \

-s malloc,256m

ExecReload=/usr/sbin/varnishreload

[Install]

WantedBy=multi-user.target

### Reload Daemon

sudo systemctl daemon-reload

### Restart Varnish

sudo systemctl restart varnish

## Configure the Apache2 to work with Varnish

### sudo nano /etc/apache2/sites-available/magento2.conf

<VirtualHost \*:8080>

ServerName magento2.com

ServerAlias www.magento2.com

ServerAdmin webmaster@localhost

DocumentRoot /var/www/html/magento245

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

</VirtualHost>

### sudo nano /etc/apache2/ports.conf

listen 8080

<IfModule ssl\_module>

Listen 443

</IfModule>

<IfModule mod\_gnutls.c>

Listen 443

</IfModule>

### sudo nano /etc/apache2/sites-available/default-ssl.conf

<IfModule mod\_ssl.c>

<VirtualHost \*:443>

ServerAdmin webmaster@localhost

ServerName [www.magento2.com](http://www.magento2.com)

DocumentRoot /var/www/html/magento245/

ProxyPreserveHost On

ProxyPass / http://127.0.0.1:80/

RequestHeader set X-Forwarded-Port "443"

RequestHeader set X-Forwarded-Proto "https"

ErrorLog ${APACHE\_LOG\_DIR}/error.log

CustomLog ${APACHE\_LOG\_DIR}/access.log combined

SSLEngine on

SSLCertificateFile /etc/ssl/certs/magento2lamp.crt

SSLCertificateKeyFile /etc/ssl/private/magento2lamp.key

<FilesMatch "\.(cgi|shtml|phtml|php)$">

SSLOptions +StdEnvVars

</FilesMatch>

<Directory /usr/lib/cgi-bin>

SSLOptions +StdEnvVars

</Directory>

</VirtualHost>

</IfModule>

### sudo systemctl restart apache2

## Configure the VCL backend

### sudo nano /etc/varnish/default.vcl

vcl 4.1;

backend default {

.host = "127.0.0.1";

.port = "8080";

}

sub vcl\_recv {

}

sub vcl\_backend\_response {

}

sub vcl\_deliver {

}

### sudo systemctl restart apache2 varnish

## Verify Varnish Cache 7 Server

### curl -I domain\_or\_ip

magento@cedcossmagento:~$ curl -I magento2.com

HTTP/1.1 302 Found

Date: Wed, 28 Sep 2022 13:14:03 GMT

Server: Apache/2.4.52 (Ubuntu)

X-Frame-Options: DENY

X-Content-Type-Options: nosniff

Set-Cookie: PHPSESSID=mqr4odvf0hq6i423h8g1a5p6it; expires=Wed, 28-Sep-2022 14:14:03 GMT; Max-Age=3600; path=/; domain=magento2.com; HttpOnly; SameSite=Lax

Expires: Tue, 28 Sep 2021 13:14:03 GMT

Cache-Control: max-age=0, must-revalidate, no-cache, no-store

Pragma: no-cache

Location: <https://www.magento2.com/>

X-Magento-Cache-Control: max-age=0, must-revalidate, no-cache, no-store

X-Magento-Cache-Debug: MISS

Content-Security-Policy-Report-Only: font-src data: 'self' 'unsafe-inline'; form-action geostag.cardinalcommerce.com geo.cardinalcommerce.com 1eafstag.cardinalcommerce.com 1eaf.cardinalcommerce.com centinelapistag.cardinalcommerce.com centinelapi.cardinalcommerce.com pilot-payflowlink.paypal.com www.paypal.com www.sandbox.paypal.com 'self' 'unsafe-inline'; frame-ancestors 'self'; frame-src fast.amc.demdex.net \*.adobe.com bid.g.doubleclick.net geostag.cardinalcommerce.com geo.cardinalcommerce.com 1eafstag.cardinalcommerce.com 1eaf.cardinalcommerce.com centinelapistag.cardinalcommerce.com centinelapi.cardinalcommerce.com www.paypal.com www.sandbox.paypal.com pilot-payflowlink.paypal.com player.vimeo.com \*.youtube.com 'self' 'unsafe-inline'; img-src assets.adobedtm.com amcglobal.sc.omtrdc.net dpm.demdex.net cm.everesttech.net \*.adobe.com widgets.magentocommerce.com data: www.googleadservices.com www.google-analytics.com googleads.g.doubleclick.net www.google.com bid.g.doubleclick.net analytics.google.com www.googletagmanager.com t.paypal.com www.paypal.com www.paypalobjects.com fpdbs.paypal.com fpdbs.sandbox.paypal.com \*.vimeocdn.com i.ytimg.com \*.youtube.com validator.swagger.io data: 'self' 'unsafe-inline'; script-src assets.adobedtm.com \*.adobe.com www.googleadservices.com www.google-analytics.com googleads.g.doubleclick.net analytics.google.com www.googletagmanager.com geostag.cardinalcommerce.com 1eafstag.cardinalcommerce.com geoapi.cardinalcommerce.com 1eafapi.cardinalcommerce.com songbird.cardinalcommerce.com includestest.ccdc02.com www.paypal.com www.sandbox.paypal.com www.paypalobjects.com t.paypal.com s.ytimg.com www.googleapis.com vimeo.com www.vimeo.com \*.vimeocdn.com \*.youtube.com 'self' 'unsafe-inline' 'unsafe-eval'; style-src \*.adobe.com 'self' 'unsafe-inline'; object-src 'self' 'unsafe-inline'; media-src \*.adobe.com 'self' 'unsafe-inline'; manifest-src 'self' 'unsafe-inline'; connect-src dpm.demdex.net amcglobal.sc.omtrdc.net www.google-analytics.com www.googleadservices.com analytics.google.com www.googletagmanager.com geostag.cardinalcommerce.com geo.cardinalcommerce.com 1eafstag.cardinalcommerce.com 1eaf.cardinalcommerce.com centinelapistag.cardinalcommerce.com centinelapi.cardinalcommerce.com www.sandbox.paypal.com www.paypalobjects.com www.paypal.com pilot-payflowlink.paypal.com 'self' 'unsafe-inline'; child-src http: https: blob: 'self' 'unsafe-inline'; default-src 'self' 'unsafe-inline' 'unsafe-eval'; base-uri 'self' 'unsafe-inline';

X-Content-Type-Options: nosniff

X-XSS-Protection: 1; mode=block

X-Frame-Options: SAMEORIGIN

X-UA-Compatible: IE=edge

Content-Length: 0

Content-Type: text/html; charset=UTF-8

X-Varnish: 131222

Age: 0

Via: 1.1 varnish (Varnish/7.0)

Connection: keep-alive

## Paise waala SSL kaise lagate hain (ye bhi free wala hai)

First Open Certbot.org website

Now take the SSH of your Server

Now Install snapd

sudo apt install snapd

sudo snap install core

sudo snap refresh core

sudo snap install --classic certbot

sudo ln -s /snap/bin/certbot /usr/bin/certbot

sudo chmod -R go-w /var/lib/snapd #if snapd status show write permission issue

sudo certbot --apache

sudo systemctl reload apache2.service

php bin/magento setup:store-config:set --use-secure-admin=1

mysql -u root -p

select \* from core\_config\_data where path like '%base%url%';

php bin/magento setup:store-config:set --base-url-secure= "https://demo2-php81.cedcommerce.com/tiktok/pub/"

php bin/magento setup:store-config:set --base-url-secure= "http://localhost/ricardo/pub/"