

Experiment 2 - Creating Amazon EC2 Instances - Creating a LAMP Instance in the AWS CLI

AIM: To create a LAMP instance in the AWS CLI.

PROCEDURE:

1. Firstly, type `sudo su` to become the root user.
2. To update all the packages in your instance type `"yum update -y"`.

```

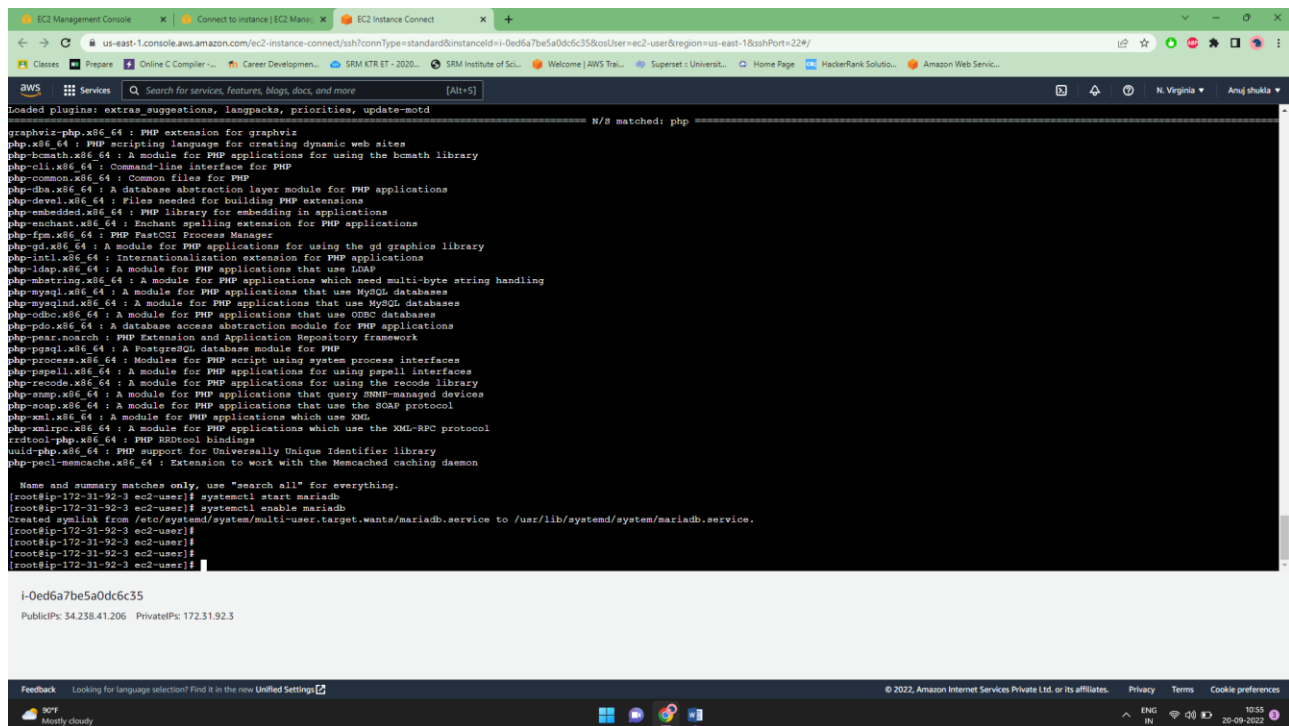
https://aws.amazon.com/amazon-linux-2/
4 package(s) needed for security, out of 11 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-92-3 ~]$ sudo su
[ec2-user@ip-172-31-92-3 ~]$ yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
Running transaction check
--> Package amazon-ssm-agent.x86_64 0:3.1.1575.0-1.amzn2 will be updated
--> Package amazon-ssm-agent.x86_64 0:3.1.1732.0-1.amzn2 will be an update
--> Package chrony.x86_64 0:4.0-3.amzn2.0.2 will be updated
--> Package chrony.x86_64 0:4.0-3.amzn2.0.2 will be an update
--> Package dhclient.x86_64 12:4.2.5-77.amzn2.1.6 will be updated
--> Package dhclient.x86_64 12:4.2.5-79.amzn2.1.1 will be an update
--> Package dhcp-common.x86_64 12:4.2.5-79.amzn2.1.1 will be an update
--> Package dhcp-common.x86_64 12:4.2.5-79.amzn2.1.1 will be an update
--> Package dhcp-libs.x86_64 12:4.2.5-79.amzn2.1.1 will be an update
--> Package dhcp-libs.x86_64 12:4.2.5-79.amzn2.1.1 will be an update
--> Package ec2-net-utils.noarch 0:1.7.0-1.amzn2 will be updated
--> Package ec2-net-utils.noarch 0:1.7.1-1.amzn2 will be an update
--> Package gnupg2.x86_64 0:2.0.22-5.amzn2.0.4 will be updated
--> Package gnupg2.x86_64 0:2.0.22-5.amzn2.0.4 will be an update
--> Package kernel.x86_64 0:5.10.135-122.509.amzn2 will be installed
--> Package kernel-tools.x86_64 0:5.10.130-118.517.amzn2 will be updated
--> Package kernel-tools.x86_64 0:5.10.135-122.509.amzn2 will be an update
--> Package microcode_ctl.x86_64 2:2.1-47.amzn2.0.12 will be updated
--> Package microcode_ctl.x86_64 2:2.1-47.amzn2.0.12 will be an update
--> Package tzdata.noarch 0:2022a-1.amzn2 will be updated
--> Package tzdata.noarch 0:2022a-1.amzn2 will be an update
--> Finished Dependency Resolution

Dependencies Resolved

Package Arch Version Repository Size
-----
i-0ed6a7be5a0dc6c35
PublicIPs: 34.238.41.206 PrivateIPs: 172.31.92.3

```

3. To install Apache server in linux, type `"yum install httpd"`.



8. Aft   enabling httpd (apache se  ve  ), go to the d  f  ct  y wh  re `cd /va  /www/html/`

9. Go to vim and type “<?php phpinfo(); ?>”.

```

php-embedded.x86_64 : PHP library for embedding in applications
php-enchant.x86_64 : Enchant spelling extension for PHP applications
php-fpm.x86_64 : PHP FastCGI Process Manager
php-gd.x86_64 : A module for PHP applications for using the gd graphics library
php-intl.x86_64 : Internationalization extension for PHP applications
php-ldap.x86_64 : A module for PHP applications that use LDAP
php-mbstring.x86_64 : A module for PHP applications which need multi-byte string handling
php-mysql.x86_64 : A module for PHP applications that use MySQL databases
php-mysqli.x86_64 : A module for PHP applications that use MySQL databases
php-odbc.x86_64 : A module for PHP applications that use ODBC databases
php-pdo.x86_64 : A database access abstraction module for PHP applications
php-pear.noarch : PHP Extension and Application Repository framework
php-pgsql.x86_64 : A PostgreSQL database module for PHP
php-process.x86_64 : Modules for PHP script using system process interfaces
php-pspell.x86_64 : A module for PHP applications for using pspell interfaces
php-recode.x86_64 : A module for PHP applications for using the recode library
php-smb.x86_64 : A module for PHP applications that query SMB-managed devices
php-soap.x86_64 : A module for PHP applications that use the SOAP protocol
php-xml.x86_64 : A module for PHP applications which use XML
php-xmlrpc.x86_64 : A module for PHP applications which use the XML-RPC protocol
rrdtool-php.x86_64 : PHP RRDtool bindings
snmp-php.x86_64 : PHP support for Universally Unique Identifier library
php-pecl-memcache.x86_64 : Extension to work with the Memcached caching daemon

Name and summary matches only, use "search all" for everything.
[root@ip-172-31-92-3 ec2-user]# systemctl start mariadb
[root@ip-172-31-92-3 ec2-user]# systemctl enable mariadb
Created symlink from /etc/systemd/system/multi-user.target.wants/mariadb.service to /usr/lib/systemd/system/mariadb.service.
[root@ip-172-31-92-3 ec2-user]#
[root@ip-172-31-92-3 ec2-user]#
[root@ip-172-31-92-3 ec2-user]# cd /var/www/html
[root@ip-172-31-92-3 ec2-user]# cd /var/www/html/
bash: cd: /var/www/html/: No such file or directory
[root@ip-172-31-92-3 ec2-user]# cd /var/www/html
[root@ip-172-31-92-3 html]# ls
[root@ip-172-31-92-3 html]# pwd
/var/www/html
[root@ip-172-31-92-3 html]# vim index.php
[root@ip-172-31-92-3 html]#

```

PublicIPs: 34.238.41.206 PrivateIPs: 172.31.92.3

Copy the public ip address of public domain name from the console and paste in the web browser.

OUTPUT:

PHP Version 5.4.16	
System	Linux ip-172-31-92-239.ap-south-1.compute.internal 5.10.130-118.517.amzn2.x86_64 #1 SMP Wed Jul 13 18:51:52 UTC 2022 x86_64
Build Date	Oct 31 2018 19:36:17
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/curl.ini, /etc/php.d/diffinfo.ini, /etc/php.d/discard.ini, /etc/php.d/mysqld.ini, /etc/php.d/mysqli.ini, /etc/php.d/mysqli_mysql.ini, /etc/php.d/pdo_mysql.ini, /etc/php.d/pdo_mysql_sqlite.ini, /etc/php.d/pdo_sqlite.ini, /etc/php.d/zip.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	APC20100525.NTS
PHP Extension Build	APC20100525.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte	disabled

RESULT:

LAMP instance was successfully created and executed in AWS CLI.