

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**”.

```

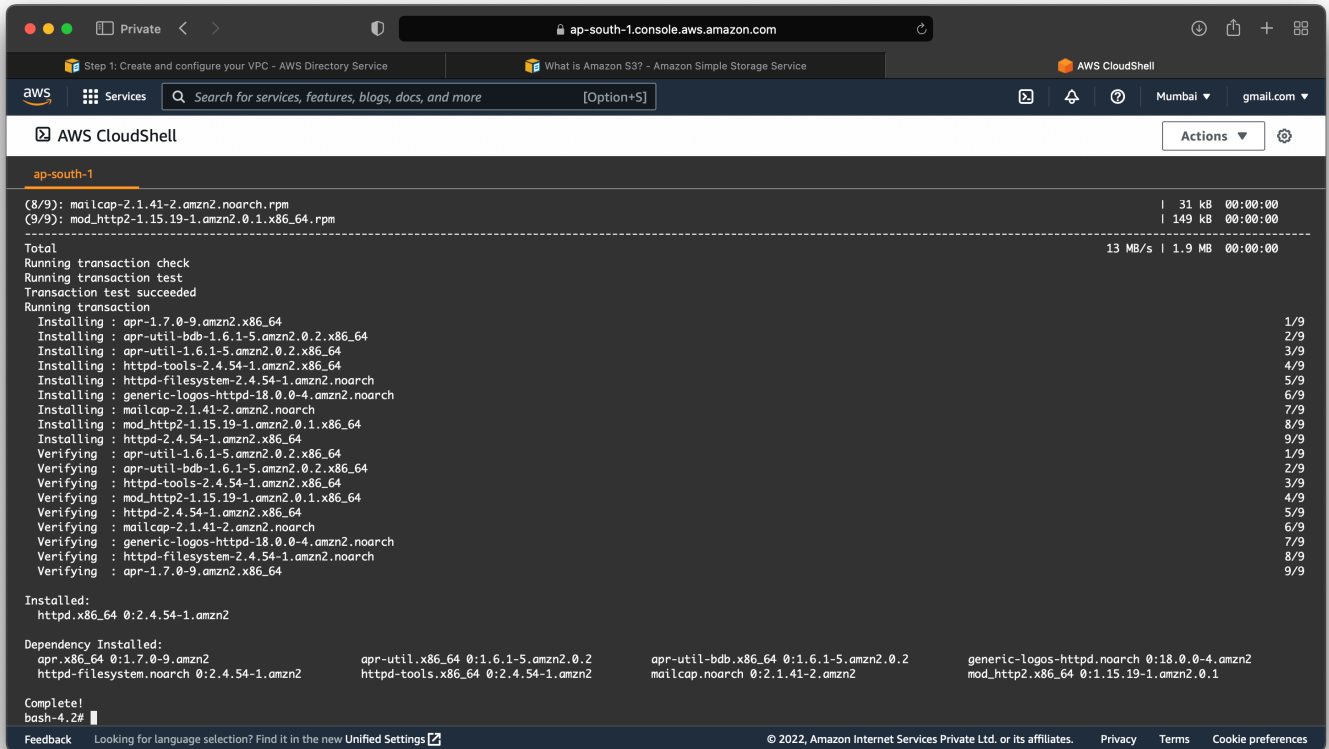
Preparing your terminal...
[cloudshell-user@ip-10-0-174-185 ~]$ Try these commands to get started:
aws help or aws <command> help or aws <command> --cli-auto-prompt
[cloudshell-user@ip-10-0-174-185 ~]$ sudo su
bash-4.2# yum update -y
Loaded plugins: ovl, priorities
Resolving Dependencies
--> Running transaction check
--> Package glibc.x86_64 0:2.26-59.amzn2 will be updated
--> Package glibc.x86_64 0:2.26-60.amzn2 will be an update
--> Package glibc-common.x86_64 0:2.26-59.amzn2 will be updated
--> Package glibc-common.x86_64 0:2.26-60.amzn2 will be an update
--> Package glibc-langpack-en.x86_64 0:2.26-59.amzn2 will be updated
--> Package glibc-langpack-en.x86_64 0:2.26-60.amzn2 will be an update
--> Package glibc-minimal-langpack.x86_64 0:2.26-59.amzn2 will be updated
--> Package glibc-minimal-langpack.x86_64 0:2.26-60.amzn2 will be an update
--> Package libcrypt.x86_64 0:2.26-59.amzn2 will be updated
--> Package libcrypt.x86_64 0:2.26-60.amzn2 will be an update
--> Package tzdata.noarch 0:2022c-1.amzn2 will be updated
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                                Arch              Version            Repository          Size
=====
Updating:
glibc                                  x86_64            2.26-60.amzn2      amzn2-core          3.4 M
glibc-common                          x86_64            2.26-60.amzn2      amzn2-core          773 k
glibc-langpack-en                     x86_64            2.26-60.amzn2      amzn2-core          289 k
glibc-minimal-langpack                x86_64            2.26-60.amzn2      amzn2-core          32 k
libcrypt                              x86_64            2.26-60.amzn2      amzn2-core          52 k
tzdata                                noarch            2022c-1.amzn2      amzn2-core          481 k
=====

```

3. To install Apache server in linux, type “**yum install httpd**”.



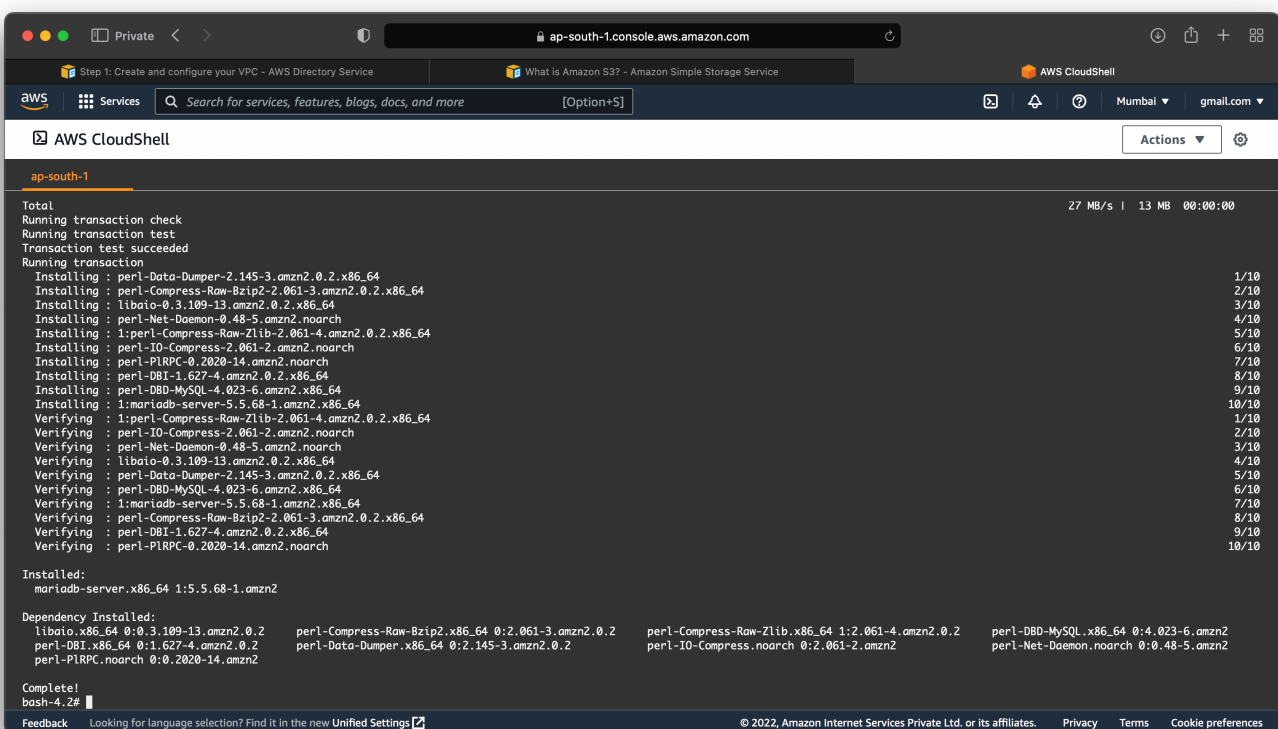
```
(8/9): mailcap-2.1.41-2.amzn2.noarch.rpm | 31 kB 00:00:00
(9/9): mod_http2-1.15.19-1.amzn2.0.1.x86_64.rpm | 149 kB 00:00:00
-----
Total | 13 MB/s | 1.9 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : apr-1.7.0-9.amzn2.x86_64 1/9
  Installing : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
  Installing : apr-util-1.6.1-5.amzn2.0.2.x86_64 3/9
  Installing : httpd-tools-2.4.54-1.amzn2.x86_64 4/9
  Installing : httpd-filesystem-2.4.54-1.amzn2.noarch 5/9
  Installing : generic-logos-httpd-18.0.0-4.amzn2.noarch 6/9
  Installing : mailcap-2.1.41-2.amzn2.noarch 7/9
  Installing : mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
  Installing : httpd-2.4.54-1.amzn2.x86_64 9/9
  Verifying : apr-util-1.6.1-5.amzn2.0.2.x86_64 1/9
  Verifying : apr-util-bdb-1.6.1-5.amzn2.0.2.x86_64 2/9
  Verifying : httpd-tools-2.4.54-1.amzn2.x86_64 3/9
  Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 4/9
  Verifying : httpd-2.4.54-1.amzn2.x86_64 5/9
  Verifying : mailcap-2.1.41-2.amzn2.noarch 6/9
  Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 7/9
  Verifying : httpd-filesystem-2.4.54-1.amzn2.noarch 8/9
  Verifying : apr-1.7.0-9.amzn2.x86_64 9/9

Installed:
  httpd.x86_64 0:2.4.54-1.amzn2

Dependency Installed:
  apr.x86_64 0:1.7.0-9.amzn2          apr-util.x86_64 0:1.6.1-5.amzn2.0.2          apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2          generic-logos-httpd.noarch 0:18.0.0-4.amzn2
  httpd-filesystem.noarch 0:2.4.54-1.amzn2          httpd-tools.x86_64 0:2.4.54-1.amzn2          mailcap.noarch 0:2.1.41-2.amzn2          mod_http2.x86_64 0:1.15.19-1.amzn2.0.1

Complete!
bash-4.2#
```

4. To install mysql or mariadb type “**yum install mariadb mariadb-server**”.



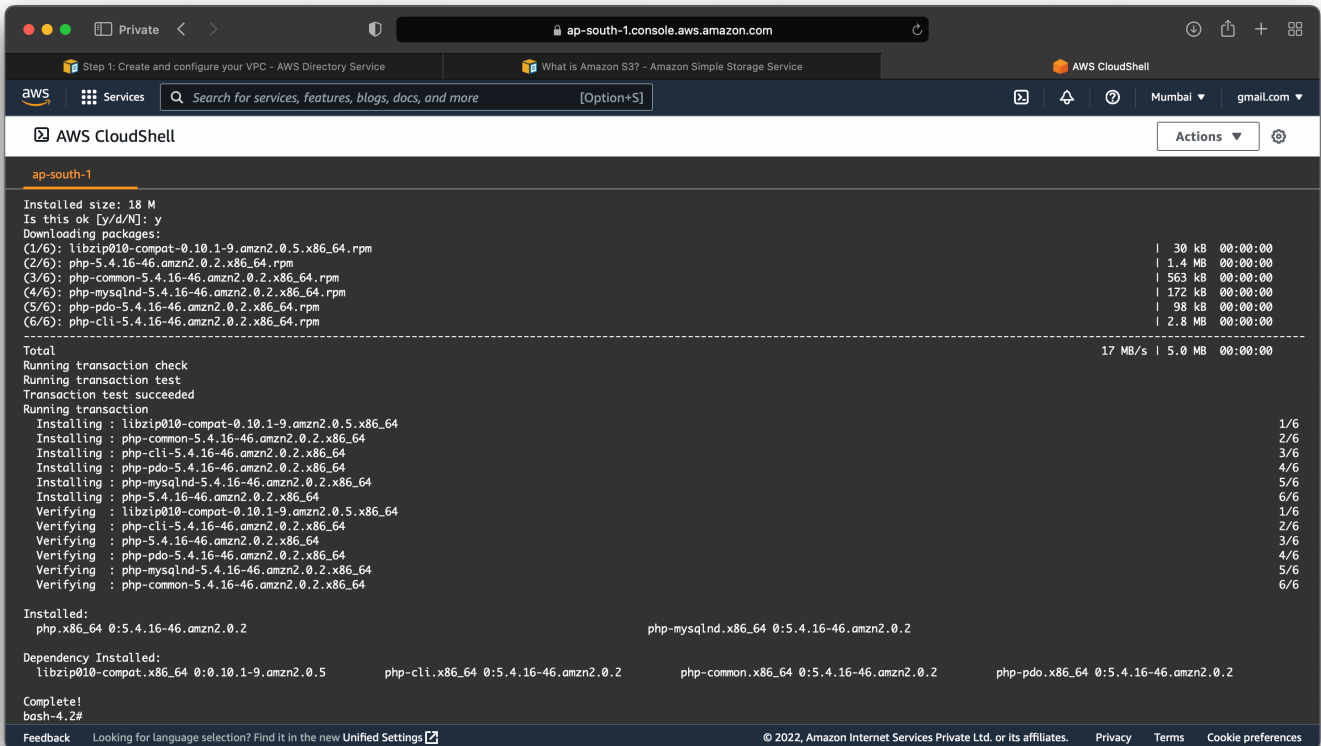
```
Total | 27 MB/s | 13 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : perl-Data-Dumper-2.145-3.amzn2.0.2.x86_64 1/10
  Installing : perl-Compress-Raw-Bzip2-2.061-3.amzn2.0.2.x86_64 2/10
  Installing : libaio-0.3.109-13.amzn2.0.2.x86_64 3/10
  Installing : perl-Net-Daemon-0.48-5.amzn2.noarch 4/10
  Installing : 1:perl-Compress-Raw-Zlib-2.061-4.amzn2.0.2.x86_64 5/10
  Installing : perl-IO-Compress-2.061-2.amzn2.noarch 6/10
  Installing : perl-PIRPC-0.2020-14.amzn2.noarch 7/10
  Installing : perl-DBI-1.627-4.amzn2.0.2.x86_64 8/10
  Installing : perl-DBD-MySQL-4.023-6.amzn2.x86_64 9/10
  Installing : 1:mariadb-server-5.5.68-1.amzn2.x86_64 10/10
  Verifying : 1:perl-Compress-Raw-Zlib-2.061-4.amzn2.0.2.x86_64 1/10
  Verifying : perl-IO-Compress-2.061-2.amzn2.noarch 2/10
  Verifying : perl-Net-Daemon-0.48-5.amzn2.noarch 3/10
  Verifying : libaio-0.3.109-13.amzn2.0.2.x86_64 4/10
  Verifying : perl-Data-Dumper-2.145-3.amzn2.0.2.x86_64 5/10
  Verifying : perl-DBD-MySQL-4.023-6.amzn2.x86_64 6/10
  Verifying : 1:mariadb-server-5.5.68-1.amzn2.x86_64 7/10
  Verifying : perl-Compress-Raw-Bzip2-2.061-3.amzn2.0.2.x86_64 8/10
  Verifying : perl-DBI-1.627-4.amzn2.0.2.x86_64 9/10
  Verifying : perl-PIRPC-0.2020-14.amzn2.noarch 10/10

Installed:
  mariadb-server.x86_64 1:5.5.68-1.amzn2

Dependency Installed:
  libaio.x86_64 0:0.3.109-13.amzn2.0.2          perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.amzn2.0.2          perl-Compress-Raw-Zlib.x86_64 1:2.061-4.amzn2.0.2          perl-DBD-MySQL.x86_64 0:4.023-6.amzn2
  perl-DBI.x86_64 0:1.627-4.amzn2.0.2          perl-Data-Dumper.x86_64 0:2.145-3.amzn2.0.2          perl-IO-Compress.noarch 0:2.061-2.amzn2          perl-Net-Daemon.noarch 0:0.48-5.amzn2
  perl-PIRPC.noarch 0:0.2020-14.amzn2

Complete!
bash-4.2#
```

5.To install php, type “**yum install php php-mysql**”.



```
ap-south-1

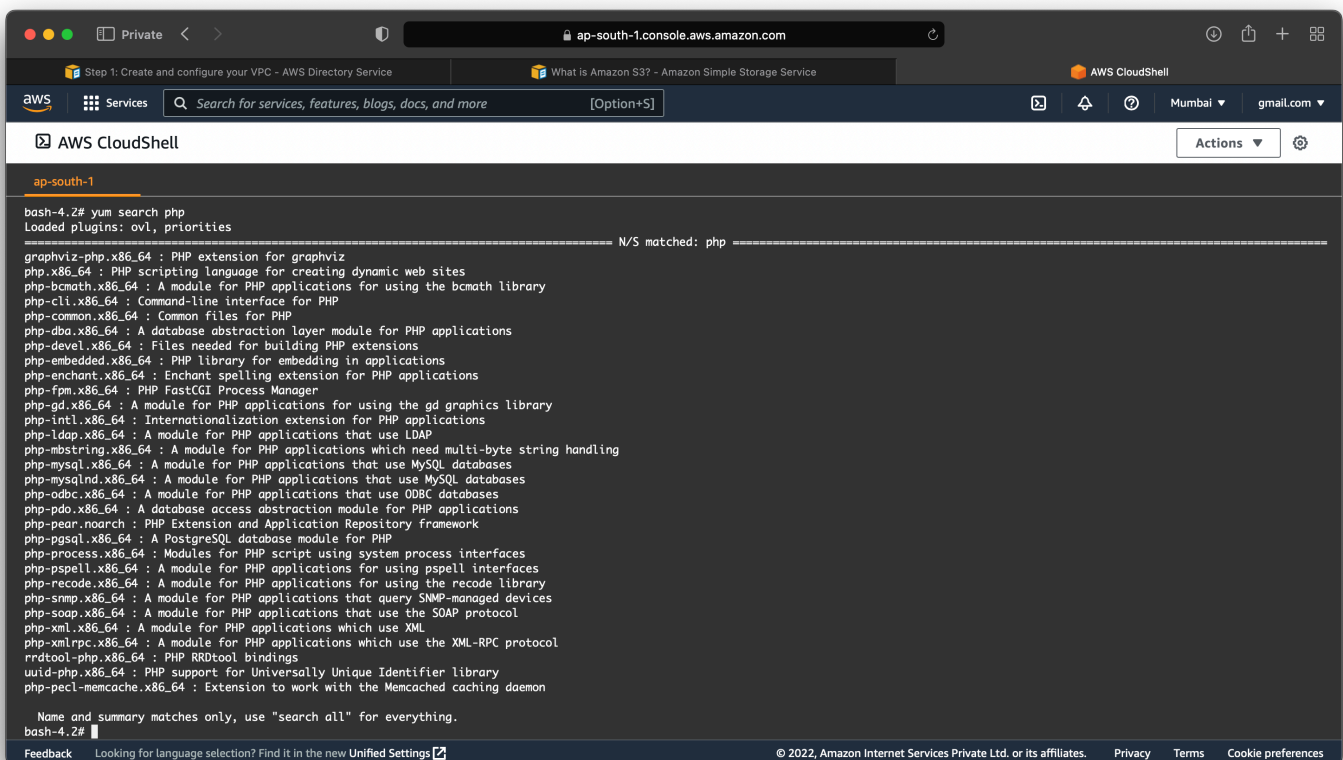
Installed size: 18 M
Is this ok [y/d/N]: y
Downloading packages:
(1/6): libzip010-compat-0.10.1-9.amzn2.0.5.x86_64.rpm | 30 kB 00:00:00
(2/6): php-5.4.16-46.amzn2.0.2.x86_64.rpm | 1.4 MB 00:00:00
(3/6): php-common-5.4.16-46.amzn2.0.2.x86_64.rpm | 563 kB 00:00:00
(4/6): php-mysqld-5.4.16-46.amzn2.0.2.x86_64.rpm | 172 kB 00:00:00
(5/6): php-pdo-5.4.16-46.amzn2.0.2.x86_64.rpm | 98 kB 00:00:00
(6/6): php-cli-5.4.16-46.amzn2.0.2.x86_64.rpm | 2.8 MB 00:00:00
-----
Total | 17 MB/s | 5.0 MB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64 1/6
  Installing : php-common-5.4.16-46.amzn2.0.2.x86_64 2/6
  Installing : php-cli-5.4.16-46.amzn2.0.2.x86_64 3/6
  Installing : php-pdo-5.4.16-46.amzn2.0.2.x86_64 4/6
  Installing : php-mysqld-5.4.16-46.amzn2.0.2.x86_64 5/6
  Installing : php-5.4.16-46.amzn2.0.2.x86_64 6/6
  Verifying : libzip010-compat-0.10.1-9.amzn2.0.5.x86_64 1/6
  Verifying : php-cli-5.4.16-46.amzn2.0.2.x86_64 2/6
  Verifying : php-5.4.16-46.amzn2.0.2.x86_64 3/6
  Verifying : php-pdo-5.4.16-46.amzn2.0.2.x86_64 4/6
  Verifying : php-mysqld-5.4.16-46.amzn2.0.2.x86_64 5/6
  Verifying : php-common-5.4.16-46.amzn2.0.2.x86_64 6/6

Installed:
  php.x86_64 0:5.4.16-46.amzn2.0.2                php-mysqld.x86_64 0:5.4.16-46.amzn2.0.2

Dependency Installed:
  libzip010-compat.x86_64 0:0.10.1-9.amzn2.0.5    php-cli.x86_64 0:5.4.16-46.amzn2.0.2    php-common.x86_64 0:5.4.16-46.amzn2.0.2    php-pdo.x86_64 0:5.4.16-46.amzn2.0.2

Complete!
bash-4.2#
```

6. Type “**yum search php**” to see all the packages installed in the server.



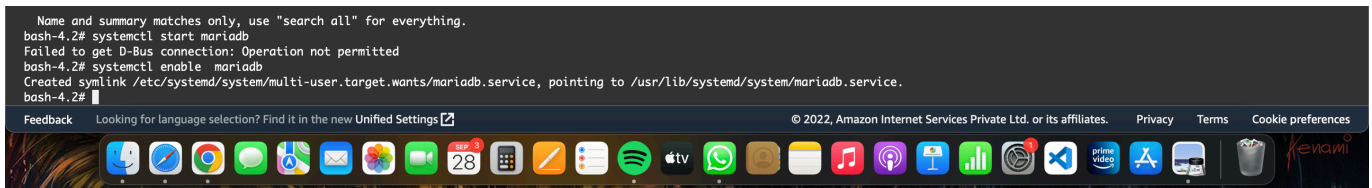
```
bash-4.2# yum search php
Loaded plugins: ovl, priorities
===== N/S matched: php =====

graphviz-php.x86_64 : PHP extension for graphviz
php.x86_64 : PHP scripting language for creating dynamic web sites
php-bcmath.x86_64 : A module for PHP applications for using the bcmath library
php-cli.x86_64 : Command-line interface for PHP
php-common.x86_64 : Common files for PHP
php-dba.x86_64 : A database abstraction layer module for PHP applications
php-devel.x86_64 : Files needed for building PHP extensions
php-embedded.x86_64 : PHP library for embedding in applications
php-enchent.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-mysqld.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-snmp.x86_64 : A module for PHP applications that query SNMP-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
uuid-php.x86_64 : PHP support for Universally Unique Identifier library
php-pecl-memcache.x86_64 : Extension to work with the Memcached caching daemon

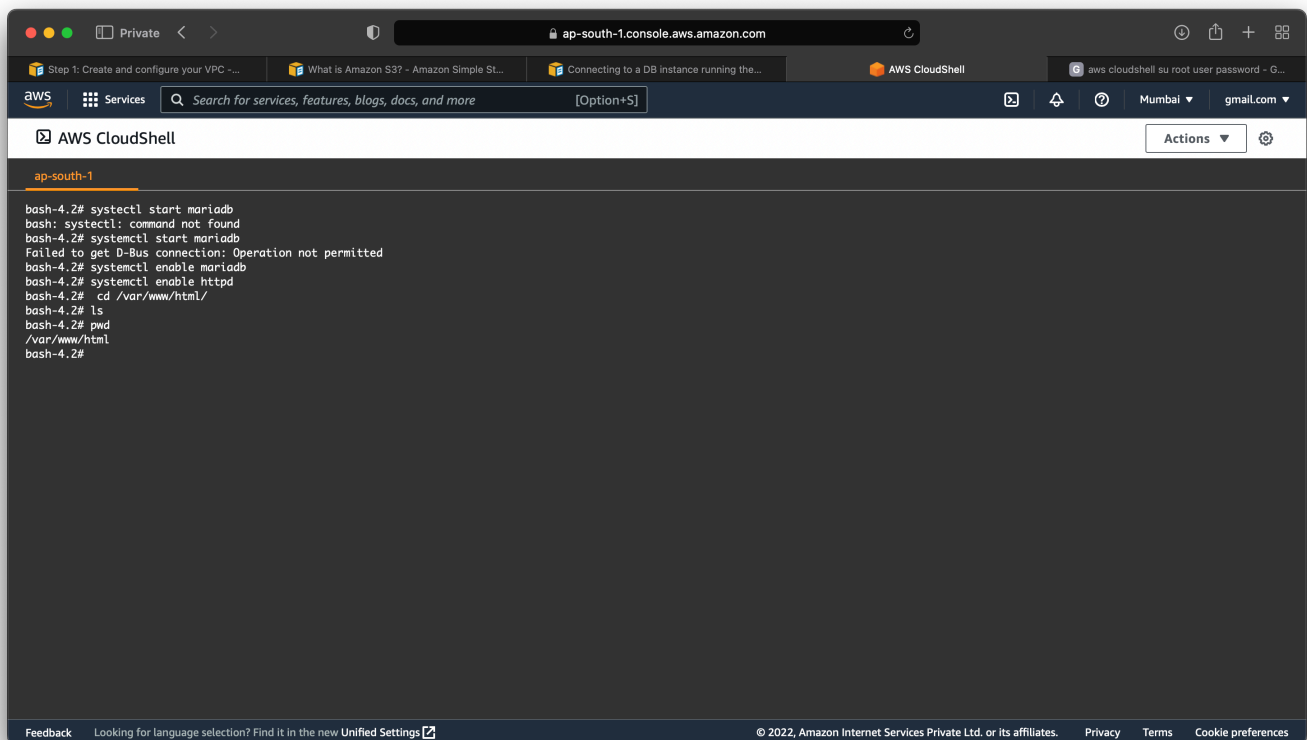
Name and summary matches only, use "search all" for everything.
bash-4.2#
```

7. Enabling the mariadb server.

```
Name and summary matches only, use "search all" for everything.
bash-4.2# systemctl start mariadb
Failed to get D-Bus connection: Operation not permitted
bash-4.2# systemctl enable mariadb
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service, pointing to /usr/lib/systemd/system/mariadb.service.
bash-4.2#
```



8. After enabling httpd (apache server), go to the directory where `cd /var/www/html/`
9. Go to vim and type “`<?php phpinfo(); ?>`”.



```
ap-south-1
bash-4.2# systemctl start mariadb
bash: systemctl: command not found
bash-4.2# systemctl start mariadb
Failed to get D-Bus connection: Operation not permitted
bash-4.2# systemctl enable mariadb
bash-4.2# systemctl enable httpd
bash-4.2# cd /var/www/html/
bash-4.2# ls
bash-4.2# pwd
/var/www/html
bash-4.2#
```

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

RESULT- AWS LAMP was successfully created and executed using was CLI.