**PROJECT 6: WEB SOLUTION WITH WORDPRESS**

WordPress is an open-source software that has evolved in progressive ways over time—supported by skilled, enthusiastic developers, designers, scientists, bloggers, and more. It provides the opportunity for anyone to create and share, from handcrafted personal anecdotes to world-changing movements. WordPress is built on PHP and MySQL, and licensed under the General Public License Version 2.0 (GPLv2). To run the open-source WordPress web solution, it is important to have the following:

- PHP 7.4 or greater
- MySQL 5.7 or greater OR MariaDB 10.3 or greater
- Nginx or Apache with mod_rewrite module
- HTTPS support

In this project, I will be preparing a storage infrastructure on two Linux servers and implement a basic web solution using WordPress. This will require me to configure storage subsystems for Web and Database servers based on Linux OS thereby working with disks, partitions and volumes in Linux. The next part will be to install WordPress and connect it to a remote MySQL database server thereby deploying Web and Database tiers of Web solution. Web solutions are generally implemented in a Three-tier Architecture comprising of the client-server, web server and database server or NFS server. The following components will be required to implement this solution:1 Webserver Linux: Red Hat Enterprise Linux 8 and 1 Database Server: Ubuntu 20.04 + MySQL

Create an EC2 instance and EBS volume

Created an instance running on RHEL which will be used as the Web server and 1 instance running on Ubuntu for the Database Server. The volumes should have the same availability zone as the Web server, with 10 GB and 3 Logical Volume Manager (LVM). You will need to use the Elastic Block Store menu on AWS to create a volume, screenshots have been added below.
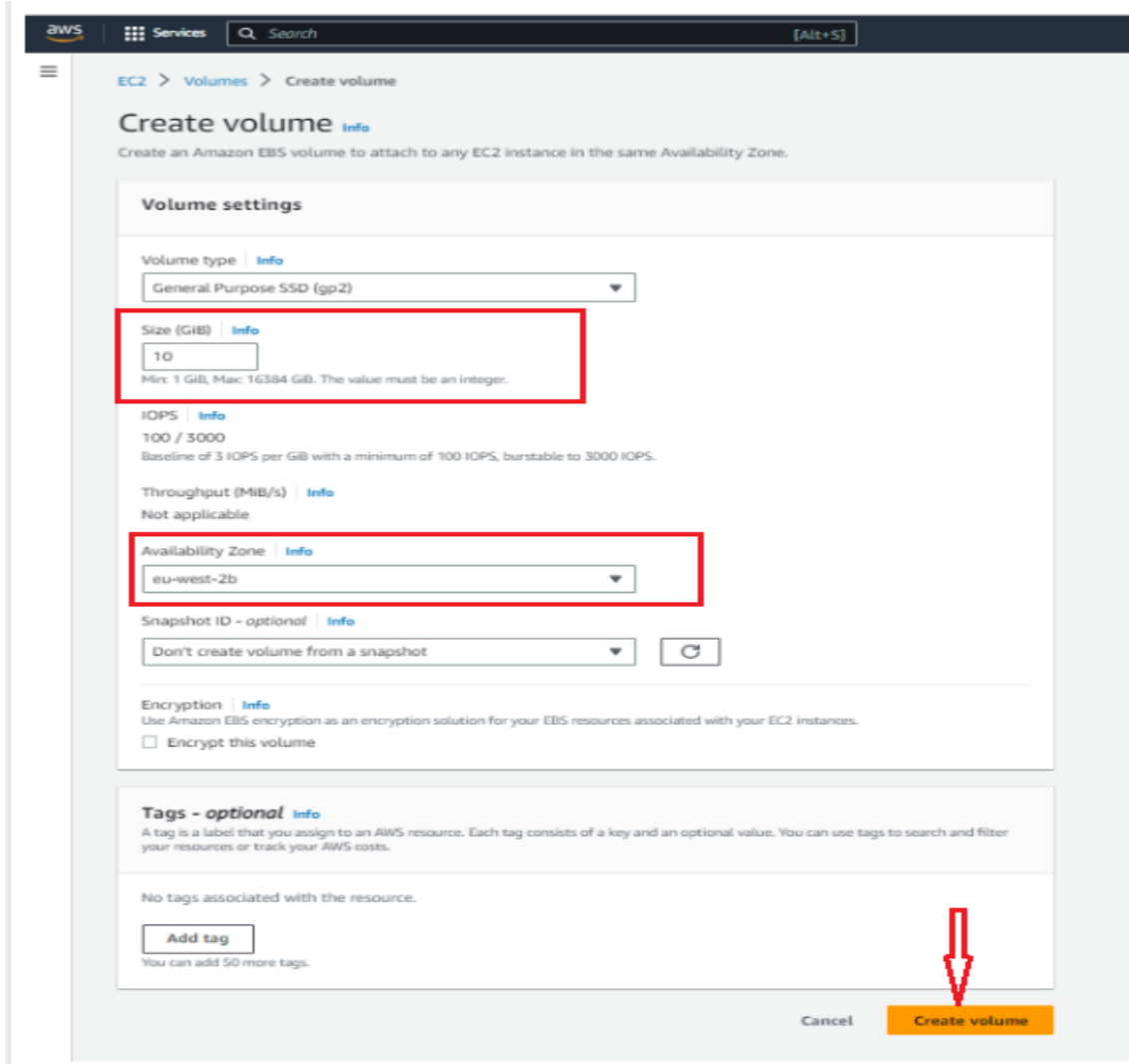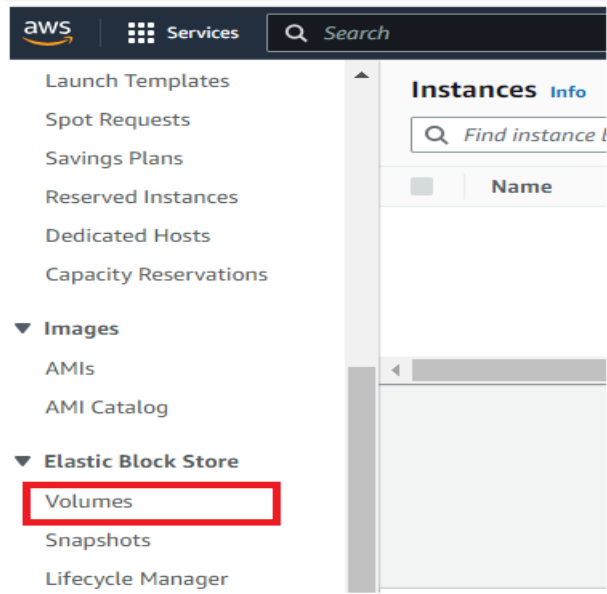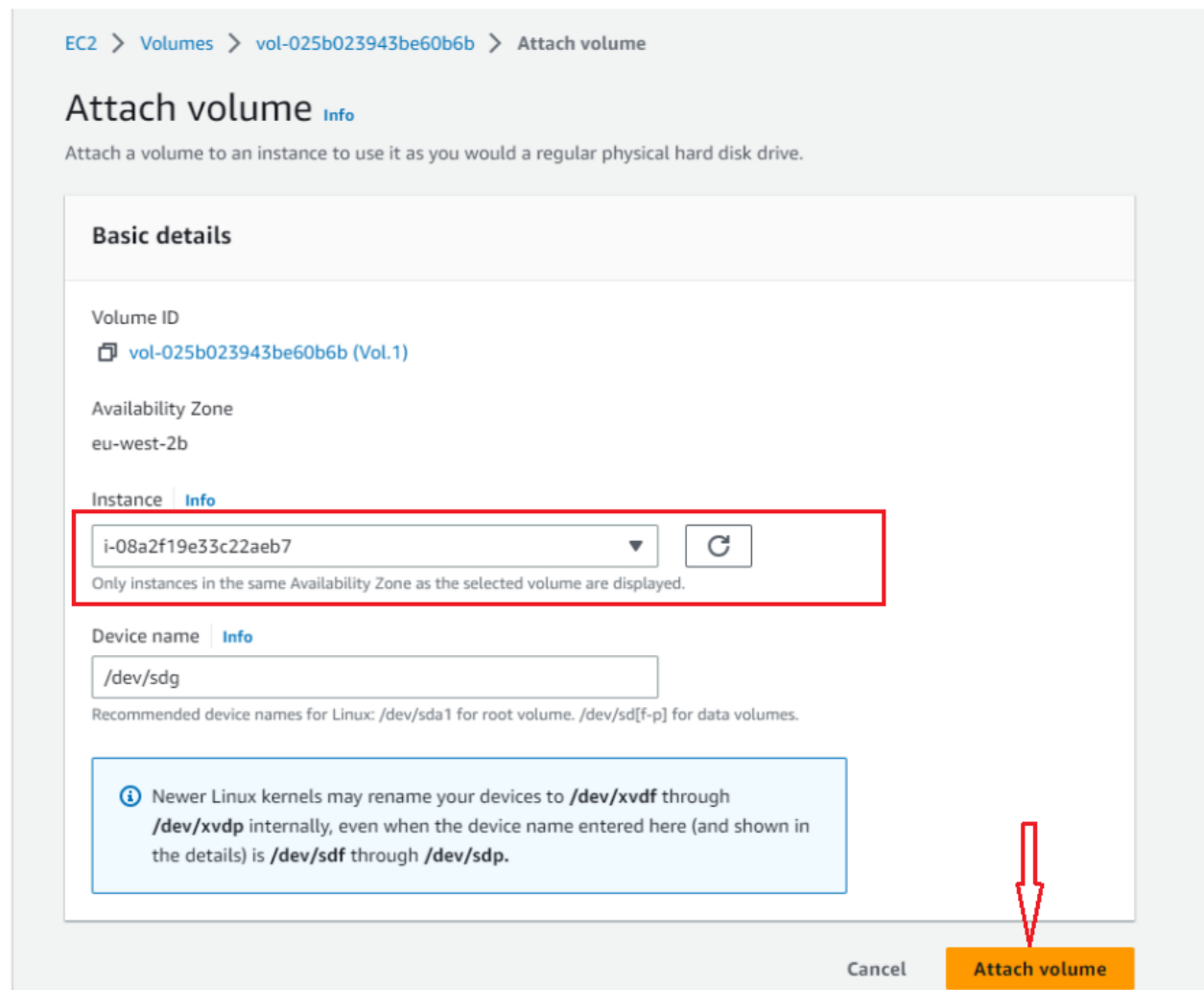
Once the volumes have been created, select actions and choose Attach volume so that the volume can now be attached to the web server instance. Ensure to use the correct instance from the drop down especially if you have more than one instance running.

Prepare the Web Server

To get started ssh into your terminal with the web server using the ssh client and then list all the blocks on the terminal using *lsblk* to confirm if it's attached. The attached disks should appear as the below showing xvdf, xvdg, and xvdh.

```
[ec2-user@ip-172-31-46-207 ~]$ lsblk
NAME      MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
xvda      202:0    0    10G  0 disk
├─xvda1   202:1    0     1M  0 part
├─xvda2   202:2    0   200M  0 part /boot/efi
├─xvda3   202:3    0   500M  0 part /boot
└─xvda4   202:4    0   9.3G  0 part /
xvdf      202:80   0    10G  0 disk
xvdg      202:96   0    10G  0 disk
xvdh      202:112  0    10G  0 disk
```

Use df -h to see all mounts and free space on the web server

```
[ec2-user@ip-172-31-46-207 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        4.0M     0  4.0M   0% /dev
tmpfs           385M     0  385M   0% /dev/shm
tmpfs           154M  4.4M  150M   3% /run
/dev/xvda4      9.4G  1.2G  8.2G  13% /
/dev/xvda3      495M  153M  343M  31% /boot
/dev/xvda2      200M  8.0K  200M   1% /boot/efi
tmpfs            77M     0   77M   0% /run/user/1000
```

Create a single partition on each of the 3 disks: xvdf, xvdg, xvdh

```
[ec2-user@ip-172-31-42-27 ~]$ sudo gdisk /dev/xvdf
```

The output should appear as below where - p - prints the partition table, - n - adds a new partition, - w - writes the table to the disk and exit/save and -y- saves the changes. This process will need to be repeated across xvdg and xvdh.

```
[ec2-user@ip-172-31-46-207 ~]$ sudo gdisk /dev/xvdf
GPT fdisk (gdisk) version 1.0.7

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.

Command (? for help): p
Disk /dev/xvdf: 20971520 sectors, 10.0 GiB
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): AAE26169-5636-4113-983B-6DE0FA1030A4
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 20971486
Partitions will be aligned on 2048-sector boundaries
Total free space is 20971453 sectors (10.0 GiB)

Number  Start (sector)    End (sector)  Size       Code  Name

Command (? for help): n
Partition number (1-128, default 1): 1
First sector (34-20971486, default = 2048) or {+-}size{KMGTP}: 2048
Last sector (2048-20971486, default = 20971486) or {+-}size{KMGTP}: 20971486
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): 8E00
Changed type of partition to 'Linux LVM'

Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): y
OK; writing new GUID partition table (GPT) to /dev/xvdf.
The operation has completed successfully.
```

Again, use *lsblk* to view the newly configured partition on each of the 3 disks.

```
[ec2-user@ip-172-31-46-207 ~]$ lsblk
NAME      MAJ:MIN RM   SIZE RO TYPE MOUNTPOINTS
xvda      202:0    0    10G  0 disk
├─xvda1 202:1    0     1M  0 part
├─xvda2 202:2    0   200M  0 part /boot/efi
├─xvda3 202:3    0   500M  0 part /boot
└─xvda4 202:4    0   9.3G  0 part /
xvdf      202:80   0    10G  0 disk
└─xvdf1 202:81   0    10G  0 part
xvdg      202:96   0    10G  0 disk
└─xvdg1 202:97   0    10G  0 part
xvdh      202:112  0    10G  0 disk
└─xvdh1 202:113  0    10G  0 part
```

Check for available partitions with *sudo lvmdiskscan*

```
[ec2-user@ip-172-31-46-207 ~]$ sudo lvmdiskscan
  /dev/xvda2 [       200.00 MiB]
  /dev/xvda3 [       500.00 MiB]
  /dev/xvda4 [         9.31 GiB]
  /dev/xvdf1 [      <10.00 GiB]
  /dev/xvdg1 [      <10.00 GiB]
  /dev/xvdh1 [      <10.00 GiB]
  0 disks
  6 partitions
  0 LVM physical volume whole disks
  0 LVM physical volumes
```

Create physical volume by running the pvcreate command to mark each of the 3 disks as physical volumes (PVs) to be used by LVM with output as physical volume successfully created. Use the *pvs* command to the physical volumes have been created successfully.

```
[ec2-user@ip-172-31-46-207 ~]$ sudo pvcreate /dev/xvdf1
  Physical volume "/dev/xvdf1" successfully created.
  Creating devices file /etc/lvm/devices/system.devices
[ec2-user@ip-172-31-46-207 ~]$ sudo pvcreate /dev/xvdg1
  Physical volume "/dev/xvdg1" successfully created.
[ec2-user@ip-172-31-46-207 ~]$ sudo pvcreate /dev/xvdh1
  Physical volume "/dev/xvdh1" successfully created.
```

```
[ec2-user@ip-172-31-46-207 ~]$ sudo pvs
  PV            VG Fmt   Attr PSize    PFree
  /dev/xvdf1       lvm2 ---  <10.00g <10.00g
  /dev/xvdg1       lvm2 ---  <10.00g <10.00g
  /dev/xvdh1       lvm2 ---  <10.00g <10.00g
```

Create a volume group and check for the created volume group

```
[ec2-user@ip-172-31-46-207 ~]$ sudo vgcreate webdata-vg /dev/xvdh1 /dev/xvdg1 /dev/xvdf1
  Volume group "webdata-vg" successfully created
```

```
[ec2-user@ip-172-31-46-207 ~]$  sudo vgs
  VG           #PV #LV #SN Attr   VSize   VFree
  webdata-vg    3   0   0 wz--n- <29.99g <29.99g
```

Use *lvcreate* command to create 2 logical volumes: lv-apps, lv-logs. lv-apps will be used to store data for the Website while lv-logs will be used to store data for logs. To verify that the Logical Volume has been created successfully run *sudo lvs.*

```
[ec2-user@ip-172-31-46-207 ~]$ sudo lvcreate -n apps-lv -L 14G webdata-vg
  Logical volume "apps-lv" created.
[ec2-user@ip-172-31-46-207 ~]$ sudo lvcreate -n logs-lv -L 14G webdata-vg
  Logical volume "logs-lv" created.
```

```
[ec2-user@ip-172-31-46-207 ~]$ sudo lvs
  LV      VG         Attr       LSize  Pool Origin Data%  Meta%  Move Log Cpy%Sync Convert
  apps-lv webdata-vg -wi-a----- 14.00g
  logs-lv webdata-vg -wi-a----- 14.00g
```

To view the complete setup use *sudo vgdisplay* and *sudo lsblk*

```
[ec2-user@ip-172-31-46-207 ~]$ sudo vgdisplay -v #view complete setup - VG, PV, and LV
    --- Volume group ---
    VG Name               webdata-vg
    System ID
    Format                lvm2
    Metadata Areas        3
    Metadata Sequence No  3
    VG Access             read/write
    VG Status             resizable
    MAX LV                0
    Cur LV                2
    Open LV               0
    Max PV                0
    Cur PV                3
    Act PV                3
    VG Size               <29.99 GiB
    PE Size               4.00 MiB
    Total PE              7677
    Alloc PE / Size       7168 / 28.00 GiB
    Free  PE / Size       509 / <1.99 GiB
    VG UUID               fYqF59-L60g-D9Vn-zq9w-iHUM-krhO-zKWHse

    --- Logical volume ---
    LV Path               /dev/webdata-vg/apps-lv
    LV Name               apps-lv
    VG Name               webdata-vg
    LV UUID               TcuxCv-NOGF-YPoL-gzIN-VVfZ-8kAQ-FsKru0
    LV Write Access       read/write
    LV Creation host, time ip-172-31-46-207.eu-west-2.compute.internal, 2023-06-10 20:27:06 +0000
    LV Status             available
    # open                0
    LV Size               14.00 GiB
    Current LE            3584
    Segments              2
    Allocation            inherit
    Read ahead sectors    auto
    - currently set to    8192
    Block device          253:0
```

```
--- Logical volume ---
LV Path                /dev/webdata-vg/logs-lv
LV Name                logs-lv
VG Name                webdata-vg
LV UUID                BCinl6-693E-xaUI-k5no-sYYp-WA1p-UUg6jC
LV Write Access        read/write
LV Creation host, time ip-172-31-46-207.eu-west-2.compute.internal, 2023-06-10 20:27:25 +0000
LV Status              available
# open                 0
LV Size                14.00 GiB
Current LE             3584
Segments               2
Allocation             inherit
Read ahead sectors     auto
- currently set to     8192
Block device           253:1

--- Physical volumes ---
PV Name                /dev/xvdh1
PV UUID                LIy4qN-3gxg-bkaj-Qnkh-ou6I-Fzuh-p6dL5R
PV Status              allocatable
Total PE / Free PE     2559 / 0

PV Name                /dev/xvdg1
PV UUID                cZEeQT-p2op-UKaU-VI4z-YWeg-KbLE-nEdO1g
PV Status              allocatable
Total PE / Free PE     2559 / 509

PV Name                /dev/xvdf1
PV UUID                kvkfdb-n5Ek-yR1n-5Fqw-sILs-3F1c-Od6AcY
PV Status              allocatable
Total PE / Free PE     2559 / 0
```

```
[ec2-user@ip-172-31-46-207 ~]$ sudo lsblk
NAME                      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINTS
xvda                      202:0    0   10G  0 disk
├─xvda1                   202:1    0    1M  0 part
├─xvda2                   202:2    0  200M  0 part /boot/efi
├─xvda3                   202:3    0  500M  0 part /boot
└─xvda4                   202:4    0  9.3G  0 part /
xvdf                      202:80   0   10G  0 disk
└─xvdf1                   202:81   0   10G  0 part
  └─webdata--vg-logs--lv 253:1    0   14G  0 lvm
xvdg                      202:96   0   10G  0 disk
└─xvdg1                   202:97   0   10G  0 part
  ├─webdata--vg-apps--lv 253:0    0   14G  0 lvm
  └─webdata--vg-logs--lv 253:1    0   14G  0 lvm
xvdh                      202:112  0   10G  0 disk
└─xvdh1                   202:113  0   10G  0 part
  └─webdata--vg-apps--lv 253:0    0   14G  0 lvm
```

To format the disk as ext4 filesystem, run each of the below commands

sudo mkfs -t ext4 /dev/webdata-vg/apps-lv

sudo mkfs -t ext4 /dev/webdata-vg/logs-lv

```
[ec2-user@ip-172-31-46-207 ~]$ sudo mkfs -t ext4 /dev/webdata-vg/apps-lv
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 3670016 4k blocks and 917504 inodes
Filesystem UUID: 0589f129-8b8d-4027-b17d-7e62e68a98c1
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

```
[ec2-user@ip-172-31-46-207 ~]$ sudo mkfs -t ext4 /dev/webdata-vg/logs-lv
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 3670016 4k blocks and 917504 inodes
Filesystem UUID: 9add1b12-da5c-452e-970a-404c5f1b5d18
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

Create /var/www/html directory to store website files and /home/recovery/logs to store the backup of log data with.

sudo mkdir -p /var/www/html

sudo mkdir -p /home/recovery/logs

Mount /var/www/html on apps-lv logical volume and backup all the files in the log directory /var/log into /home/recovery/logs.

sudo mount /dev/webdata-vg/apps-lv /var/www/html/

sudo rsync -av /var/log/. /home/recovery/logs/

```
[ec2-user@ip-172-31-46-207 ~]$ sudo rsync -av /var/log/. /home/recovery/logs/
sending incremental file list
./
README -> ../../usr/share/doc/systemd/README.logs
btmp
choose_repo.log
cloud-init-output.log
cloud-init.log
cron
dnf.librepo.log
dnf.log
dnf.rpm.log
hawkey.log
kdump.log
lastlog
maillog
messages
secure
spooler
tallylog
wtmp
audit/
audit/audit.log
chrony/
insights-client/
private/
rhsm/
rhsm/rhsm.log
rhsm/rhsmcertd.log
sssd/
tuned/
tuned/tuned.log

sent 713,676 bytes  received 457 bytes  1,428,266.00 bytes/sec
total size is 711,759  speedup is 1.00
```

The UUID of the device will be used to update the /etc/fstab file; Use *sudo blkid* to view the
UUID and update the same using *sudo vi /etc/fstab*

```
[ec2-user@ip-172-31-46-207 ~]$ sudo blkid
/dev/xvda4: LABEL="root" UUID="287d9c0b-0e0f-4e92-8534-45733aa3dc68" TYPE="xfs" PARTUUID="6264d520-3fb9-423f-8ab8-7a0a8e3d3562"
/dev/mapper/webdata--vg-logs--lv: UUID="9add1b12-da5c-452e-970a-404c5f1b5d18" TYPE="ext4"
/dev/xvda2: SEC_TYPE="msdos" UUID="7B77-95E7" TYPE="vfat" PARTUUID="68b2905b-df3e-4fb3-80fa-49d1e773aa33"
/dev/xvda3: LABEL="boot" UUID="7bc24af7-289d-4bce-b17e-300c3aafe968" TYPE="xfs" PARTUUID="cb07c243-bc44-4717-853e-28852021225b"
/dev/xvda1: PARTUUID="fac7f1fb-3e8d-4137-a512-961de09a5549"
/dev/xvdh1: UUID="LIy4qN-3gxg-bkaj-Qnkh-ou6I-Fzuh-p6dL5R" TYPE="LVM2_member" PARTLABEL="Linux LVM" PARTUUID="93fc9bfa-9730-4932-bec8-478fef1c835d"
/dev/xvdf1: UUID="kvkfdb-n5Ek-yR1n-5Fqw-sILs-3F1c-Od6AcY" TYPE="LVM2_member" PARTLABEL="Linux LVM" PARTUUID="782ee01c-86db-449e-8018-5aa61f9ab734"
/dev/mapper/webdata--vg-apps--lv: UUID="0589f129-8b8d-4027-b17d-7e62e68a98c1" TYPE="ext4"
/dev/xvdg1: UUID="cZEeQT-p2op-UKaU-VI4z-YWeg-KbLE-nEdO1g" TYPE="LVM2_member" PARTLABEL="Linux LVM" PARTUUID="cb9af7b3-fd12-4341-8999-1d633a685487"
```

```
UUID=9add1b12-da5c-452e-970a-404c5f1b5d18          /var/www/html ext4      defaults      0      0
UUID=0589f129-8b8d-4027-b17d-7e62e68a98c1          /var/log      ext4      defaults      0      0
# UUID=7B77-95E7        /boot/efi      vfat    defaults,uid=0,gid=0,umask=077,shortname=winnt 0      2
```

To test the configuration and reload the daemon use.

sudo mount -a
sudo systemctl daemon-reload

There is a need to verify that the setup is running by using the *df -h* command.

```
[ec2-user@ip-172-31-46-207 ~]$ df -h
Filesystem                       Size  Used Avail Use% Mounted on
devtmpfs                         4.0M     0  4.0M   0% /dev
tmpfs                            385M     0  385M   0% /dev/shm
tmpfs                            154M  4.4M  150M   3% /run
/dev/xvda4                       9.4G  1.3G  8.1G  14% /
/dev/xvda3                       495M  153M  343M  31% /boot
/dev/xvda2                       200M  8.0K  200M   1% /boot/efi
tmpfs                             77M     0   77M   0% /run/user/1000
/dev/mapper/webdata--vg-logs--lv  14G  788K   13G   1% /var/www/html
/dev/mapper/webdata--vg-apps--lv  14G   24K   13G   1% /var/log
```

Install WordPress on the Web Server

Update the repository and install wget, Apache and its dependencies.

sudo yum -y update

sudo yum -y install wget httpd php php-mysqlnd php-fpm php-json

```
[ec2-user@ip-172-31-42-75 ~]$ sudo wget http://wordpress.org/latest.tar.gz
--2023-06-10 22:22:15--  http://wordpress.org/latest.tar.gz
Resolving wordpress.org (wordpress.org)... 198.143.164.252
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://wordpress.org/latest.tar.gz [following]
--2023-06-10 22:22:15--  https://wordpress.org/latest.tar.gz
Connecting to wordpress.org (wordpress.org)|198.143.164.252|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 23020109 (22M) [application/octet-stream]
Saving to: 'latest.tar.gz'

latest.tar.gz           100%[===================================================================>]  21.95M  10.3MB/s    in 2.1s

2023-06-10 22:22:18 (10.3 MB/s) - 'latest.tar.gz' saved [23020109/23020109]
```

Enable and start Apache

sudo systemctl enable httpd
sudo systemctl start httpd

Install PHP from Remi's repository on the web server

sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
sudo dnf install dnf-utils http://rpms.remirepo.net/enterprise/remi-release-8.rpm
sudo dnf module reset php
sudo dnf module enable php -y

sudo dnf install php php-opcache php-gd php-curl php-mysqlnd -y

sudo systemctl start php-fpm

sudo systemctl enable php-fpm

sudo systemctl status php-fpm

sudo setsebool -P httpd_execmem 1

```
[ec2-user@ip-172-31-46-207 ~]$ sudo systemctl status php-fpm
● php-fpm.service - The PHP FastCGI Process Manager
     Loaded: loaded (/usr/lib/systemd/system/php-fpm.service; enabled; preset: disabled)
     Active: active (running) since Sat 2023-06-10 22:44:28 UTC; 5min ago
   Main PID: 16161 (php-fpm)
     Status: "Processes active: 0, idle: 5, Requests: 0, slow: 0, Traffic: 0req/sec"
      Tasks: 6 (limit: 4421)
     Memory: 13.3M
        CPU: 74ms
     CGroup: /system.slice/php-fpm.service
             ├─16161 "php-fpm: master process (/etc/php-fpm.conf)"
             ├─16163 "php-fpm: pool www"
             ├─16164 "php-fpm: pool www"
             ├─16165 "php-fpm: pool www"
             ├─16166 "php-fpm: pool www"
             └─16167 "php-fpm: pool www"

Jun 10 22:44:28 ip-172-31-46-207.eu-west-2.compute.internal systemd[1]: Starting The PHP FastCGI Process Manager...
Jun 10 22:44:28 ip-172-31-46-207.eu-west-2.compute.internal systemd[1]: Started The PHP FastCGI Process Manager.
```

Restart Apache with *sudo systemctl restart httpd* and download and copy Wordpress to var/www/html

mkdir wordpress

cd wordpress

sudo wget http://wordpress.org/latest.tar.gz

sudo tar xzvf latest.tar.gz

sudo rm -rf latest.tar.gz

sudo cp -R wp-config-sample.php wp-config.php

sudo cp -R wordpress/. /var/www/html/

Install Mysql

sudo yum install mysql-server

sudo systemctl start mysqld

*sudo systemctl enable mysqld*

sudo systemctl status mysqld

Configure SELinux Policies

sudo chown -R apache:apache /var/www/html/

sudo chcon -t httpd_sys_rw_content_t /var/www/html/ -R

sudo setsebool -P httpd_can_network_connect=1

sudo setsebool -P httpd_can_network_connect_db 1

To disable the welcome page on the web server use the below and ensure that the service is running afterwards. Run mysql -h <databse-private-ip> -u <db-username> -p <db-pasword> < tooling-db.sql to see if you are now able to connect to the database. Ensure port 80 is opened in the security group on the web server and the source will be anywhere.

sudo mv /etc/httpd/conf.d/welcome.conf /etc/httpd/conf.d/welcome.conf_backup

sudo mysql -h 172.31.7.146 -u wordpress -p
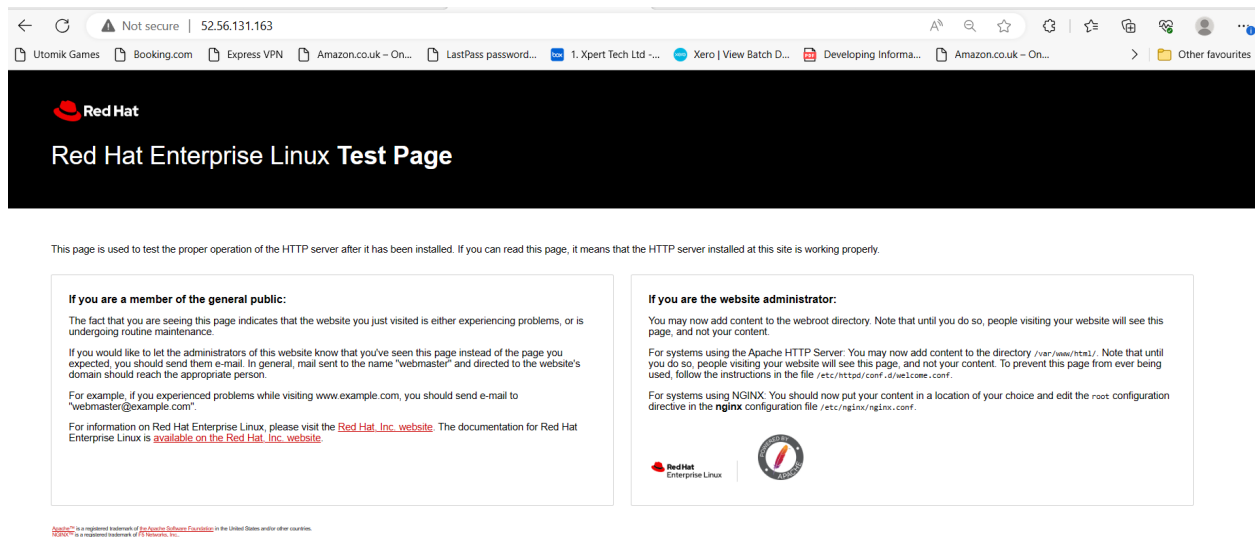
Prepare the Database Server

It is important to follow the same steps used to prepare the Web Server, but instead of apps-lv create db-lv and mount it to /db directory instead of /var/www/html/. Below are the commands to use for ease of access.

```
sudo gdisk /dev/xvdf
sudo gdisk /dev/xvdg
sudo gdisk /dev/xvdh
lsblk
sudo yum install lvm2
lsblk
sudo yum install lvm2
sudo lvmdiskscan
sudo pvcreate /dev/xvdf1
sudo pvcreate /dev/xvdg1
sudo pvcreate /dev/xvdh1
sudo pvs
sudo vgcreate database-vg /dev/xvdh1 /dev/xvdg1 /dev/xvdf1
sudo lvcreate -n db-lv -L 5G database-vg
sudo lvcreate -n logs-lv -L 5G database-vg
sudo vgdisplay -v
lsblk
```

```
sudo mkfs -t ext4 /dev/database-vg/db-lv
sudo mkfs -t ext4 /dev/database-vg/logs-lv
sudo mkdir -p /db
sudo mount /dev/database-vg/db-lv /db
sudo mount /dev/database-vg/logs-lv /var/log
sudo rsync -av /var/log/. /home/recovery/logs/
sudo mkdir -p /home/recovery/logs
sudo mount /dev/database-vg/logs-lv /var/log
sudo rsync -av /var/log/. /home/recovery/logs/
sudo mount /dev/database-vg/logs-lv /var/log
sudo rsync -av /home/recovery/logs/. /var/log
sudo vi /etc/fstab
sudo blkid
sudo vi /etc/fstab
sudo mount -a
sudo systemctl daemon-reload
```

Install Mysql on the Database Server

Run the application update before installing the package and then connect to the MySQL database.

```
ubuntu@ip-172-31-35-205:~$ sudo apt update
```
```
ubuntu@ip-172-31-35-205:~$ sudo apt install mysql-server -y
```
```
ubuntu@ip-172-31-35-205:~$ sudo mysql
```

I will need to create the database as WordPress, create the database user, and grant privileges using the Web Server Private IP Address.

CREATE DATABASE wordpress;
CREATE USER `myuser`@`172.31.46.207` IDENTIFIED BY 'definedpassword';
GRANT ALL ON wordpress.* TO 'myuser'@'172.31.46.207';
FLUSH PRIVILEGES;
SHOW DATABASES;

You should now be able to see the created database with show databases;

```
mysql> SHOW DATABASES;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
| wordpress          |
+--------------------+
5 rows in set (0.01 sec)
```

Change the MySQL bind address to 0.0.0.0 with the below steps.

```
ubuntu@ip-172-31-35-205:~$ cd /etc/mysql/mysql.conf.d/
```

```
ubuntu@ip-172-31-35-205:/etc/mysql/mysql.conf.d$ sudo vi mysqld.cnf
```

```
#
# This group is read both both by the client and the server
# use it for options that affect everything
#
[client-server]

#
# include all files from the config directory
#
!includedir /etc/my.cnf.d
[mysqld]
bind-address=0.0.0.0
~
```

Restart the MySQL service afterwards.

```
ubuntu@ip-172-31-35-205:/etc/mysql/mysql.conf.d$ sudo systemctl restart mysql.service
```

Configure WordPress to connect to the remote database.

To ssh into it the Database from the web server, I will open port 3306 for My SQL and add the Web Server's IP address in the Security Group on the Database server on EC2. Ensure the MySQL client has already been installed and test connection from the Web Server to the Database server using the mysql-client. Once the below page is displayed. It shows that the connection is successful.

```
[ec2-user@ip-172-31-46-207 /]$ sudo mysql -u myuser -h 172.31.42.75 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Type show databases; to confirm everything works well.

```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| performance_schema |
| wordpress          |
+--------------------+
3 rows in set (0.00 sec)
```

Now edit the permissions and configuration so Apache can make use WordPress.

sudo vi wp-config.php

```
// ** Database settings - You can get this
info from your web host ** //
/** The name of the database for
WordPress*/
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'dbuser' );

/** Database password */
define( 'DB_PASSWORD', '*********' );

/** Database hostname */
define( 'DB_HOST', '172.31.38.50' );

/** Database charset to use in creating
database tables. */
define( 'DB_CHARSET', 'utf8' );
```

Browse the webservers with their respective IPs or DNS Try to access the link to WordPress http://<Web-Server-Public-IP-Address>/WordPress/ and it should display as the WordPress.

This means that I have now implemented a web solution with WordPress and connected it successfully to the MySQL database.
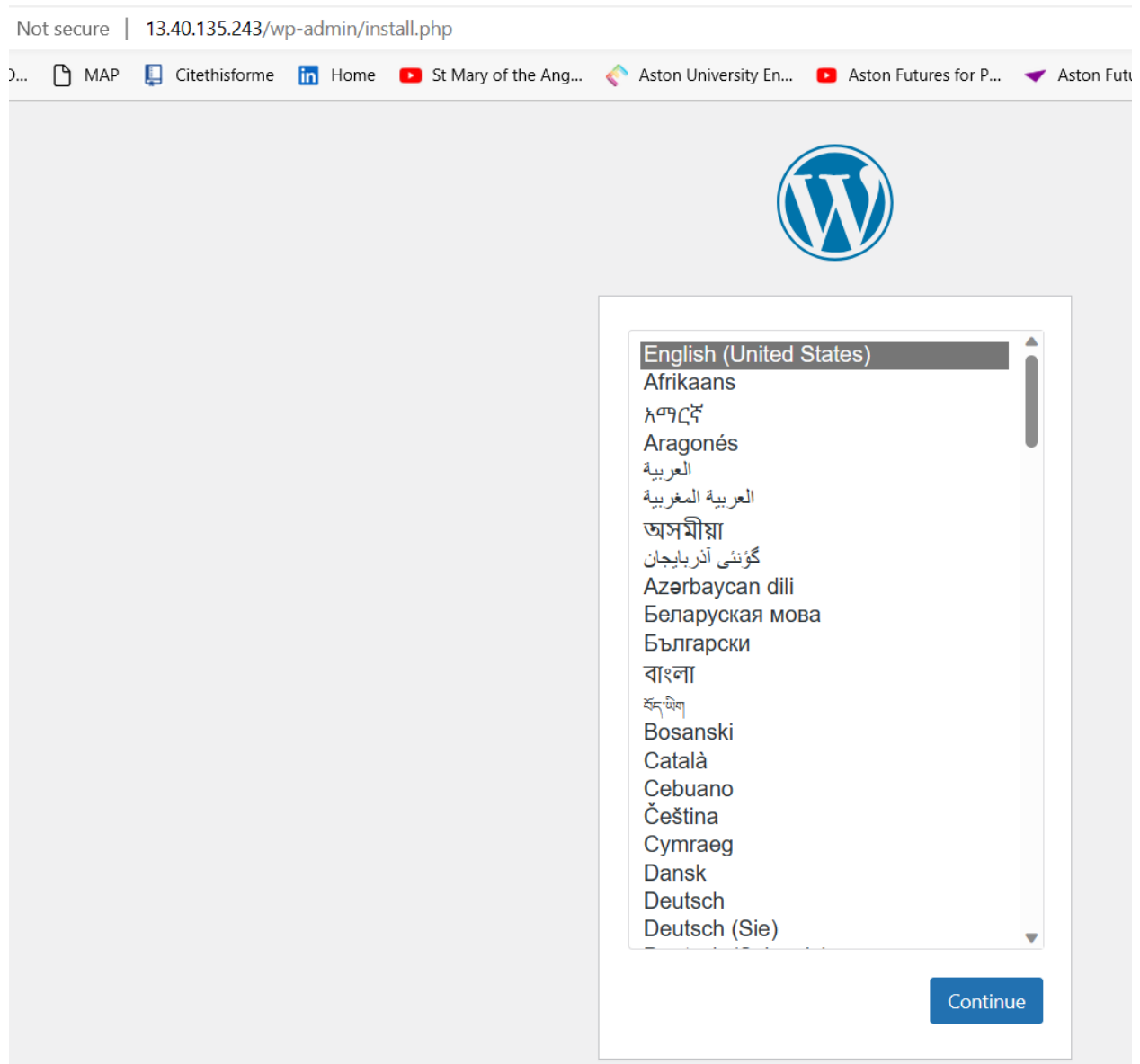
Welcome

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Do not worry, you can always change these settings later.

Site Title

DevOps

Username

Nenye

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password

👁 Hide

Strong

Important: You will need this password to log in. Please store it in a secure location.

Your Email

Double-check your email address before continuing.

Search engine visibility

☐ Discourage search engines from indexing this site

It is up to search engines to honor this request.

Install WordPress

← C ⚠ Not secure | 13.40.135.243/wp-admin/

Xero | View Batch D...   MAP   Citethisforme   Home   St Mary of the Ang...   Aston University En...   Aston Futures for P...   Aston Futures   LibGuides at Aston...   o   Other favourite

W 🏠 DevOps 💬 0 + New   Howdy, Nenye

🎛 Dashboard   **Dashboard**   Screen Options ▼   Help ▼

Home
Updates
                                Welcome to WordPress!   ✕ Dismiss
📌 Posts
📷 Media   Learn more about the 6.2.2 version.
📄 Pages
💬 Comments

🎨 Appearance
🔌 Plugins   ✏ Author rich content with blocks   ▦ Customize your entire site with   ◑ Switch up your site's look & feel
👤 Users   and patterns   block themes   with Styles
🔧 Tools
⚙ Settings   Block patterns are pre-configured block   Design everything on your site — from the   Tweak your site, or give it a whole new look!
◀ Collapse menu   layouts. Use them to get inspired or create new   header down to the footer, all using blocks and   Get creative — how about a new color palette
                                 pages in a flash.   patterns.   or font?

                                 Add a new page   Open site editor

                                 Site Health Status   ∧ ∨ ▲   **Quick Draft**   ∧ ∨ ▲

                                                                              Title
                                 No information yet...   Site health checks will automatically run periodically to gather
                                                          information about your site. You can also visit the Site Health
                                                          screen to gather information about your site now.

REFERENCES

WordPress Available at: https://wordpress.org/about/ (Accessed: 17 June 2023).