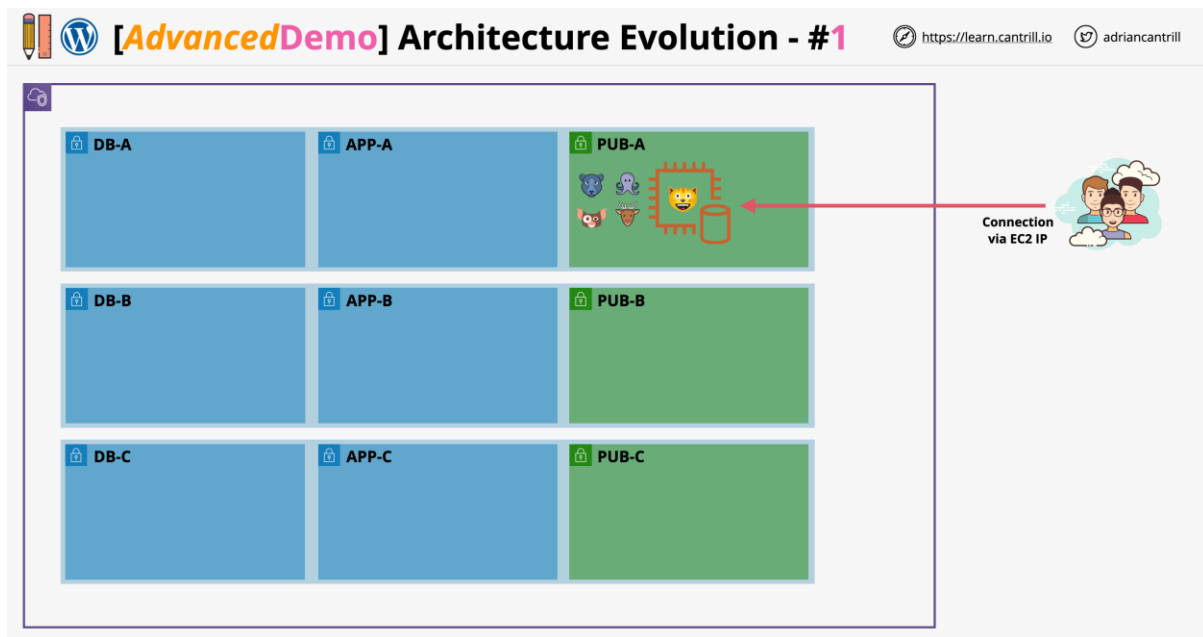


## Advanced Demo - Web App - Single Server to Elastic Evolution



In stage 1 of this advanced demo you will:

- Setup the environment which WordPress will run from.
- Configure some SSM Parameters which the manual and automatic stages of this advanced demo series will use
- and perform a manual install of wordpress and a database on the same EC2 instance.

This is the starting point .. the common wordpress configuration which you will evolve over the coming demo stages.

### STAGE 1A - Login to an AWS Account

Login to an AWS account using a user with admin privileges and ensure your region is set to us-east-1 N. Virginia

Click [HERE](#) to auto configure the VPC which WordPress will run from

Wait for the STACK to move into the CREATE\_COMPLETE state before continuing.

### STAGE 1B - Create an EC2 Instance to run wordpress

Move to the EC2 console <https://console.aws.amazon.com/ec2/v2/home?region=us-east-1>  
Click Launch Instance

For name use Wordpress-Manual

Select Amazon Linux

From the dropdown make sure Amazon Linux 2023 AMI is selected  
ensure 64-bit (x86) is selected in the architecture dropdown.

Under instance type

Select whatever instance shows as Free tier eligible (probably t2 or t3.micro)

Under Key Pair(login) select Proceed without a KeyPair (not recommended)

For Network Settings, click Edit and in the VPC dropdown select A4LVPC

for Subnet select sn-Pub-A

Make sure for both Auto-assign public IP and Auto-assign IPv6 IP you set to Enable

Under security Group Check Select an existing security group

Select A4LVPC-SGWordpress it will have randomness after it, thats ok :)

We will leave storage as default so make no changes here

Expand Advanced Details

For IAM instance profile role select A4LVPC-WordpressInstanceProfile **THIS BIT IS IMPORTANT**

Find the Credit Specification Dropdown and choose Standard (some accounts aren't enabled for Unlimited) Click Launch Instance

Click View All instances

Wait for the instance to be in a RUNNING state

*you can continue to stage 1B below while the instance is provisioning*

### **STAGE 1B - Create SSM Parameter Store values for wordpress**

Storing configuration information within the SSM Parameter store scales much better than attempting to script them in some way. In this sub-section you are going to create parameters to store the important configuration items for the platform you are building.

Open a new tab to <https://console.aws.amazon.com/systems-manager/home?region=us-east-1>

Click on Parameter Store on the menu on the left

**MAKE SURE WITH THE BELOW ... NO WHITESPACE BEFORE OR AFTER .. MAKE SURE THE UPPER/LOWER CASE IS CORRECT**

#### **Create Parameter - DBUser (the login for the specific wordpress DB)**

Click Create Parameter Set Name to /A4L/Wordpress/DBUser Set Description to Wordpress Database User

Set Tier to Standard

Set Type to String

Set Data type to text

Set Value to a4lwordpressuser

Click Create parameter

#### **Create Parameter - DBName (the name of the wordpress database)**

Click Create Parameter Set Name to /A4L/Wordpress/DBName Set Description to Wordpress Database Name

Set Tier to Standard

Set Type to String

Set Data type to text

Set Value to a4lwordpressdb

Click Create parameter

#### **Create Parameter - DBEndpoint (the endpoint for the wordpress DB .. )**

Click Create Parameter Set Name to /A4L/Wordpress/DBEndpoint Set Description to Wordpress Endpoint Name

Set Tier to Standard

Set Type to String

Set Data type to text

Set Value to localhost  
Click Create parameter

### Create Parameter - DBPassword (the password for the DBUser)

Click Create Parameter Set Name to /A4L/Wordpress/DBPassword Set Description to Wordpress DB Password  
Set Tier to Standard  
Set Type to SecureString  
Set KMS Key Source to My Current Account  
Leave KMS Key ID as default (should be alias/aws/ssm).  
Set Value to 4n1m4I54L1f3  
Click Create parameter

### Create Parameter - DBRootPassword (the password for the database root user, used for self-managed admin)

Click Create Parameter Set Name to /A4L/Wordpress/DBRootPassword Set Description to Wordpress DBRoot Password  
Set Tier to Standard  
Set Type to SecureString  
Set KMS Key Source to My Current Account  
Leave KMS Key ID as default (should be alias/aws/ssm).  
Set Value to 4n1m4I54L1f3  
Click Create parameter

### STAGE 1C - Connect to the instance and install a database and wordpress

Right click on Wordpress-Manual choose Connect Choose Session Manager  
Click Connect (if connect isn't highlighted then just wait a few minutes and try again). *if after 5-10 minutes this still doesn't let you connect, it's possible you didn't add the A4LVPC-WordpressInstanceProfile instance role. You need to right click on the instance, security, modify IAM role and then add A4LVPC-WordpressInstanceProfile. Once done, reboot the instance*  
type sudo bash and press enter  
type cd and press enter  
type clear and press enter

### Bring in the parameter values from SSM

Run the commands below to bring the parameter store values into ENV variables to make the manual build easier. **IF AT ANY POINT YOU ARE DISCONNECTED, RERUN THE BLOCK BELOW, THEN CONTINUE FROM WHERE YOU DISCONNECTED FROM**

```
DBPassword=$(aws ssm get-parameters --region us-east-1 --names /A4L/Wordpress/DBPassword --with-decryption --query Parameters[0].Value)
```

```
DBPassword=`echo $DBPassword | sed -e 's/^"'/' -e 's/"$//'`
```

```
DBRootPassword=$(aws ssm get-parameters --region us-east-1 --names /A4L/Wordpress/DBRootPassword --with-decryption --query Parameters[0].Value)
```

```
DBRootPassword=`echo $DBRootPassword | sed -e 's/^"'/' -e 's/"$//'`
```

```
DBUser=$(aws ssm get-parameters --region us-east-1 --names /A4L/Wordpress/DBUser --query Parameters[0].Value)
```

```
DBUser=`echo $DBUser | sed -e 's/^"' -e 's/"$/'`
```

```
DBName=$(aws ssm get-parameters --region us-east-1 --names /A4L/Wordpress/DBName --query Parameters[0].Value)
```

```
DBName=`echo $DBName | sed -e 's/^"' -e 's/"$/'`
```

```
DBEndpoint=$(aws ssm get-parameters --region us-east-1 --names /A4L/Wordpress/DBEndpoint --query Parameters[0].Value)
```

```
DBEndpoint=`echo $DBEndpoint | sed -e 's/^"' -e 's/"$/'`
```

### **Install updates**

```
sudo dnf -y update
```

### **Install Pre-Req's and Web Server**

```
sudo dnf install wget php-mysqlnd httpd php-fpm php-mysqli mariadb105-server php-json php php-devel stress -y
```

### **Set DB and HTTP Server to running and start by default**

```
sudo systemctl enable httpd
```

```
sudo systemctl enable mariadb
```

```
sudo systemctl start httpd
```

```
sudo systemctl start mariadb
```

### **Set the MariaDB Root Password**

```
sudo mysqladmin -u root password $DBRootPassword
```

### **Download and extract Wordpress**

```
sudo wget http://wordpress.org/latest.tar.gz -P /var/www/html
```

```
cd /var/www/html
```

```
sudo tar -zxvf latest.tar.gz
```

```
sudo cp -rvf wordpress/* .
```

```
sudo rm -R wordpress
```

```
sudo rm latest.tar.gz
```

### **Configure the wordpress wp-config.php file**

```
sudo cp ./wp-config-sample.php ./wp-config.php
```

```
sudo sed -i "s/'database_name_here'/'$DBName'/g" wp-config.php
```

```
sudo sed -i "s/'username_here'/'$DBUser'/g" wp-config.php
```

```
sudo sed -i "s/'password_here'/'$DBPassword'/g" wp-config.php
```

### **Fix Permissions on the filesystem**

```
sudo usermod -a -G apache ec2-user
```

```
sudo chown -R ec2-user:apache /var/www
```

```
sudo chmod 2775 /var/www
```

```
sudo find /var/www -type d -exec chmod 2775 {} \;
```

```
sudo find /var/www -type f -exec chmod 0664 {} \;
```

### **Create Wordpress User, set its password, create the database and configure permissions**

```
sudo echo "CREATE DATABASE $DBName;" >> /tmp/db.setup
```

```
sudo echo "CREATE USER '$DBUser'@'localhost' IDENTIFIED BY '$DBPassword';" >> /tmp/db.setup
```

```
sudo echo "GRANT ALL ON $DBName.* TO '$DBUser'@'localhost';" >> /tmp/db.setup
```

```
sudo echo "FLUSH PRIVILEGES;" >> /tmp/db.setup
```

```
sudo mysql -u root --password=$DBRootPassword < /tmp/db.setup
```

```
sudo rm /tmp/db.setup
```

### **Test Wordpress is installed**

Open the EC2 console <https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:sort=desc:tag:Name>

Select the Wordpress-Manual instance

copy the IPv4 Public IP into your clipboard (**DON'T CLICK THE OPEN LINK ... just copy the IP**) Open that IP in a new tab

You should see the wordpress welcome page

### **Perform Initial Configuration and make a post**

in Site Title enter Catagram

in Username enter admin in Password enter 4n1m4l54L1f3

in Your Email enter your email address

Click Install WordPress Click Log In

In Username or Email Address enter admin

in Password enter the previously noted down strong password

Click Log In

Click Posts in the menu on the left

Select Hello World! Click Bulk Actions and select Move to Trash Click Apply

Click Add New

If you see any popups close them down

For title The Best Animal(s)!

Click the + under the title, select Gallery Click Upload

Select some animal pictures.... if you dont have any use google images to download some

Upload them

Click Publish

Click Publish Click view Post

This is your working, manually installed and configured wordpress

## **STAGE 1 - FINISH**

This configuration has several limitations which you will resolve one by one within this lesson :-

- The application and database are built manually, taking time and not allowing automation
- ^^ it was slow and annoying ... that was the intention.
- The database and application are on the same instance, neither can scale without the other
- The database of the application is on an instance, scaling IN/OUT risks this media
- The application media and UI store is local to an instance, scaling IN/OUT risks this media
- Customer Connections are to an instance directly ... no health checks/auto healing
- The IP of the instance is hardcoded into the database ....
- Go to <https://console.aws.amazon.com/ec2/v2/home?region=us-east-1#Instances:sort=desc:tag:Name>
- Right click Wordpress-Manual , Stop Instance, Stop
- Right click Wordpress-Manual , Start Instance, Start
- the IP address has changed ... which is bad
- Try browsing to it ...
- What about the images....?
- The images are pointing at the old IP address...
- Right click Wordpress-Manual , Terminate Instance, Terminate