Extended Capstone Project

Setting up As-Found State and then Upgrading to Decoupled Data Solution

2023,S2 (*29 Sep*)

Background Reading - About Learner Lab Environment

In the current version of the BCCS355 practical assessment (called “Capstone Project”) we use the AWS Academy Learner Lab as the environment to work in. The Learner Lab brings the advantage of US$100 credit for each student, and is a non-destructive lab environment with an extendable timer - at the end of the lab timer your resources will not be terminated, but they may be suspended. This is in contrast to the standard lab environments in your course, which are destructive after timer expiry.

Setting Up Scenario As-Found State

While AWS Academy offers a good Capstone Project, at Ara we need to modify and extend that project so we meet all required course outcomes and meet BICT degree compliance. While the academy’s associate architect course itself has a lab environment for the capstone project, it does not grant enough permissions for you to meet all of our course outcomes.
In Ara’s extended capstone project we need you to set up the whole solution from scratch, including VPC and subnets, and to set up an IPsec VPN connection thus achieving a hybrid network. Therefore we choose to host our extended capstone project in the “AWS Academy Learner Lab”, which has all the permissions you need. You should have received an invite and it will appear as a separate class in your AWS Academy.

Thus during the project we will play the role of different people:

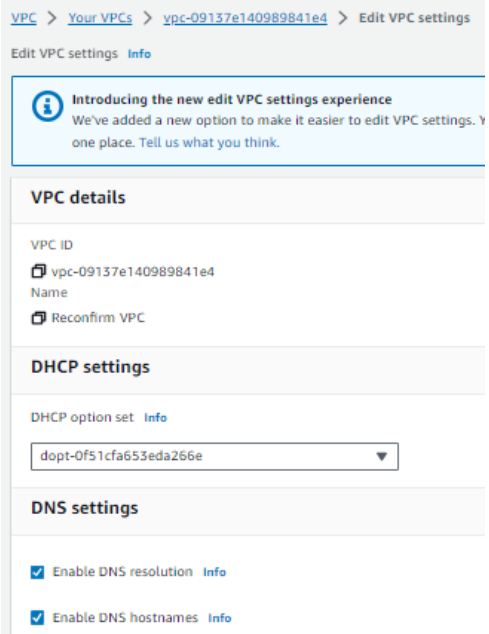
- **Phase 1 Overview: Playing the role of Shirley Rodriguez** we create a new VPC with public and private subnets, then deploy an EC2 using a particular AMI image, and using a defined instance profile role that gives access to reading parameters in AWS Systems Manager.
  The instance needs to deploy with a supplied user data script. This script will deploy a php web server and Mariadb database, and set this instance to fetch parameters from the AWS System Parameter Store - such as database username and password. *Initially Shirley’s web front end will be querying her in-built MariaDb database, due to the as-found parameter settings.*
- **Phase 2 Overview:** In this phase we **play the role of a consultant in a cloud architecting company** that Shirley approaches to check the integrity of her system, and to improve it. Therefore we decide it is important to decouple her data away from the existing web front-end instance to a separate data-tier service that is redundant and can scale. We also need to create better security and redundancy of the system, and allow the front-end to scale to meet increasing customer demand. For the decoupled data we decided to use RDS - a managed service offered by AWS.

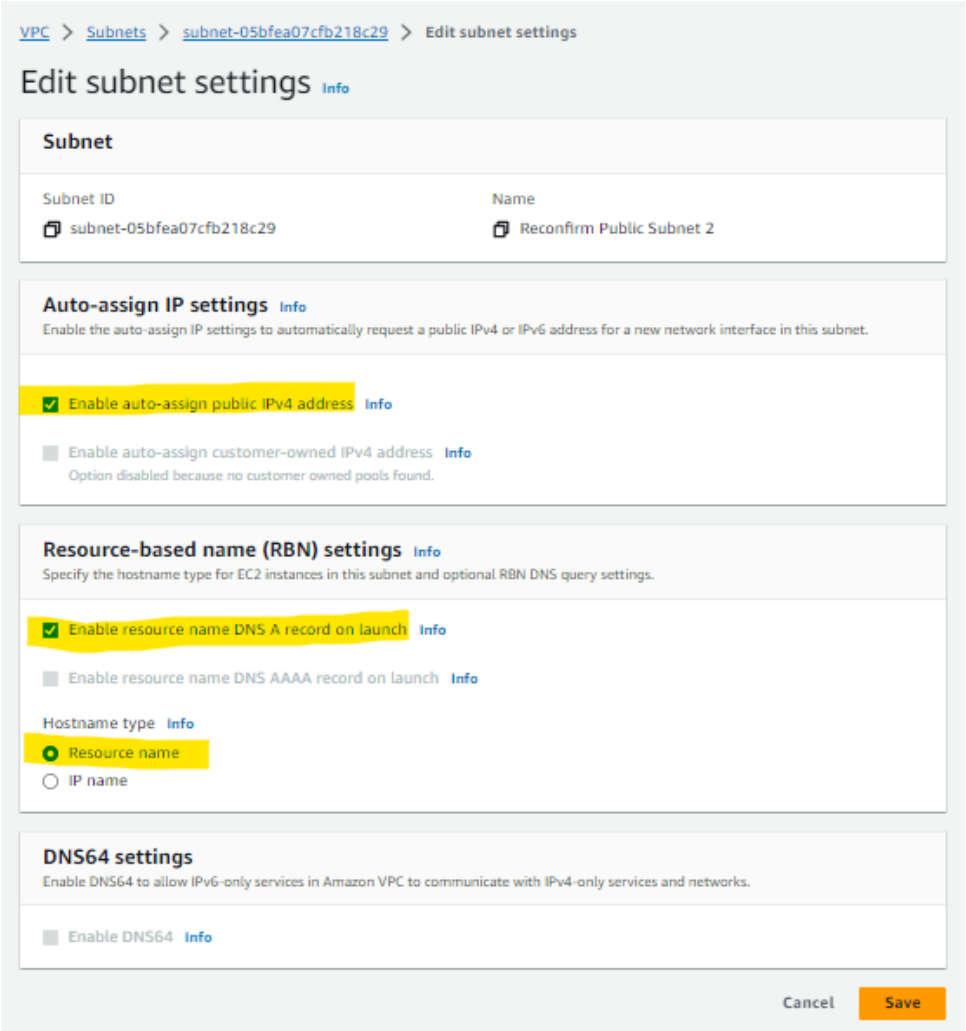
Phase 1

Deployment of VPC and Subnets

We need two public and two private subnets with appropriate routing tables and an Internet Gateway, such that only the public subnets are accessible from the Internet. Don’t forget to define appropriate subnet associations.

It will be important to ensure that your deployed VPC has settings to support DNS names being available for any web front-end resources, and subnet settings so instances deployed in your public subnets will receive a public IP address by default.





Setting Up a Private Key for Project

Although AWS Academy will set up a vockey for you, it is recommended to create your own public /private key-pair to use for encrypted access to your instances, because this is what you will need to do in industry (in private AWS accounts). (*e.g.: Capstone Key 2023,S2*)

Deployment of Shirley's Initial Front-End Instance (As Found)

Do we need a Bastion Host?

Shirley initially labelled her instance as a bastion host, because she can access it as administrator. But as-found that instance is also her web and data server, which is not good practice.
She has asked you if it is important to have a bastion host. You advise it is, but it should be solely for administrative access into the cloud, not for web or other services. The web front-end functionality should be separated away to other instances.

However in this project, initially you need to set up Shirley's "As-Found" conditions.

Deploying EC2 for Web Front-End

For this we will use a certain version of AMI image - ami-0e1c5d8c23330dee3 and a defined User Data script. ami-0e1c5d8c23330dee3 can be found under Community AMIs (AWS is the verified publisher). An instance-profile called "LabInstanceProfile" is supplied in the Learner Lab environment, and it must be attached to the EC2 instance. This instance profile calls an IAM Role called LabRole, which allows this EC2 to successfully query for SSM parameters.

Ensure you deploy into your VPC - it needs to be a public subnet where you receive a public IP address, unless you are deploying behind a load balancer - in which case deploying into private subnets is an option for your web-tier. In the latter case, public access to your website will be via your load balancer's URL (Public DNS) reference. I suggest initial experimental deployment will avoid use of the load balancer, just to prove functionality - but ultimately you do need the load balancer to complete the customer's solution requirements.

Security Groups

You need to ensure security groups for your web-tier have appropriate settings, and that security groups for your data tier have appropriate settings.

User Data

I have proven this User Data is successful, if you deploy on the AMI image specified above. Otherwise commands such as `amazon-linux-extras` do not work.

```
#!/bin/bash -ex
yum -y update
amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
yum install -y httpd mariadb-server
chkconfig httpd on
service httpd start
cd /home/ec2-user
wget
https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACACAD-2/21-course-proje
ct/s3/Countrydatadump.sql
chown ec2-user:ec2-user Countrydatadump.sql
cd /var/www/html
```

```
wget
https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACACAD-2/21-course-proje
ct/s3/Example.zip
unzip Example.zip -d /var/www/html/
chown -R ec2-user:ec2-user /var/www/html
```

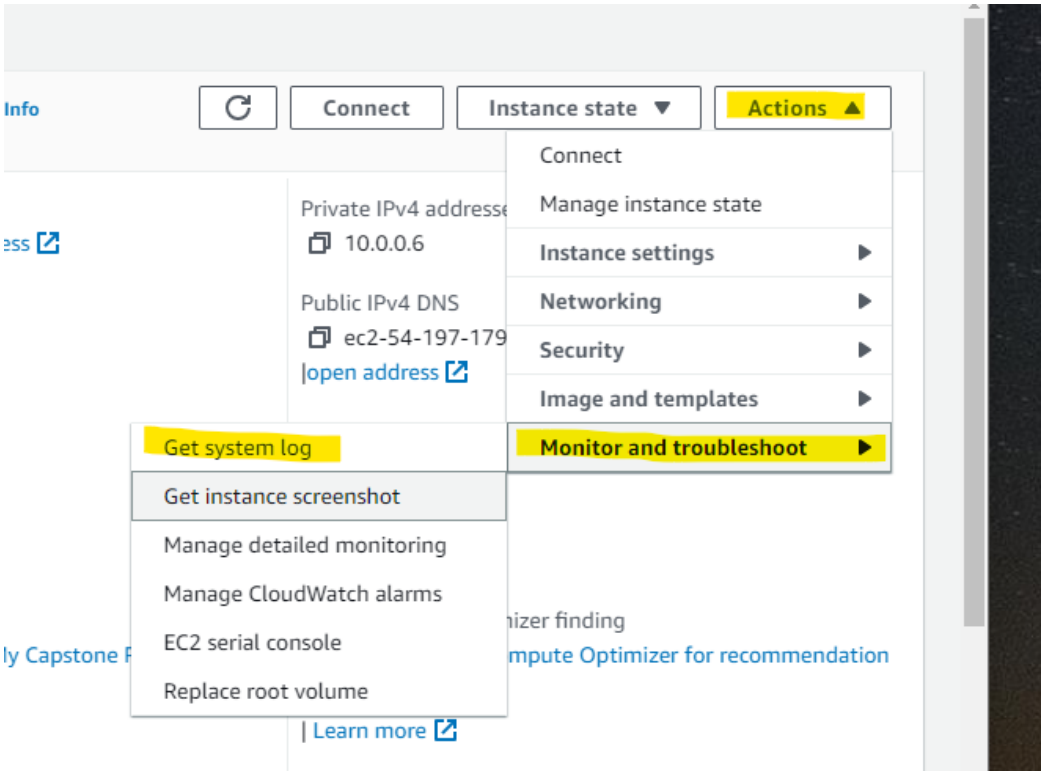
Some tips about User Data Pasting

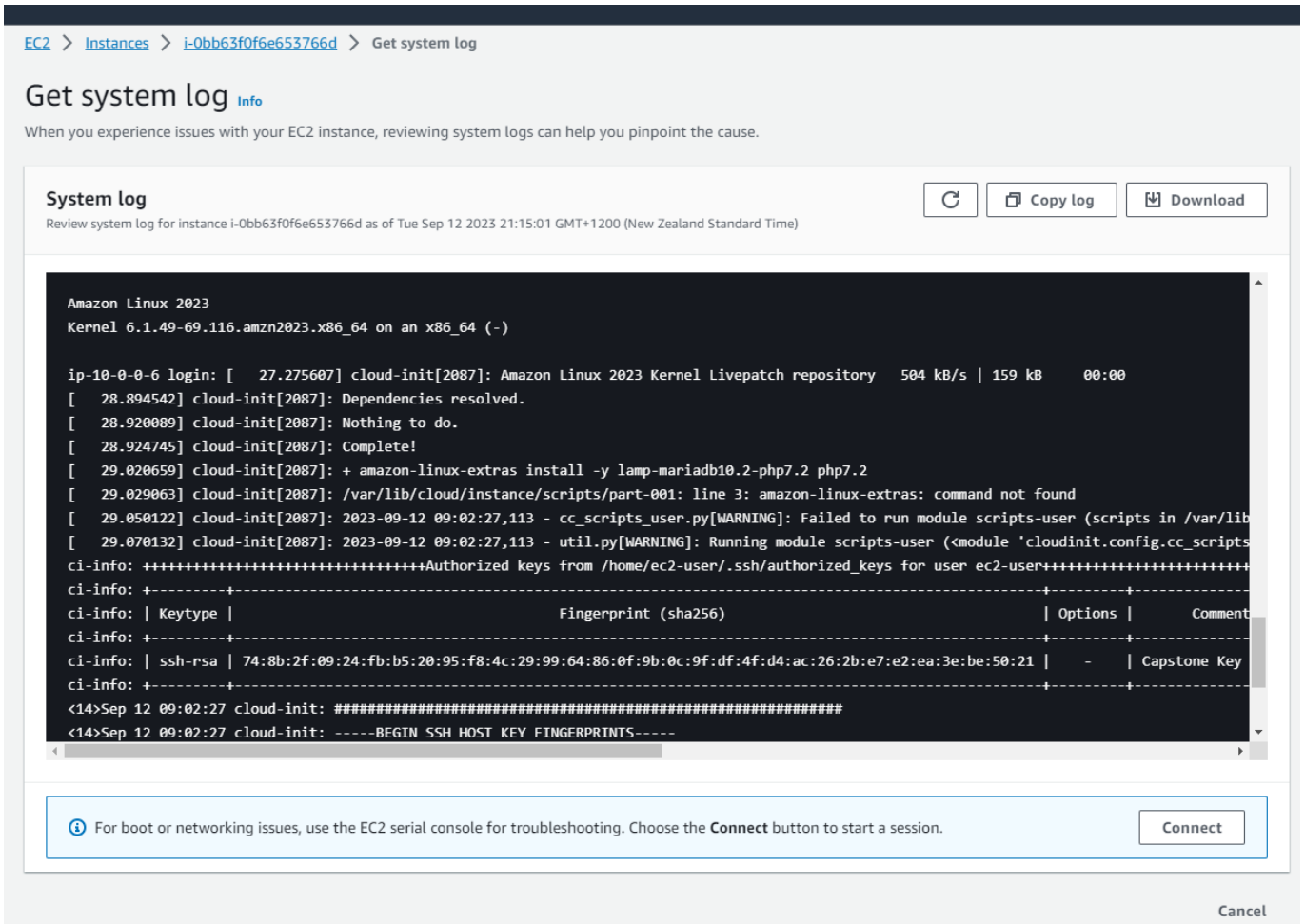
Sometimes there can be encoding issues when you copy and paste user data from one environment (such as Word or a PDF document) to another environment (such as the EC2 setup dialogue of the AWS Console). Therefore I recommend:

- Initially paste into Notepad ++, check all characters look correct, then copy. *This step helps to remove unusual control characters and encoding.*
- Right click and paste as plain text, into the User data field of the EC2 setup dialogue in AWS Console.

Has Your User Data Worked?

You need to be sure there were no errors in your User Data script during EC2 startup.
While you could temporarily run up an EC2 with no user data (clean start) and then test the proposed script line by line at the bash prompt, a quicker way would be to just check for errors of running your user data deployment. We can inspect the EC2 system log as follows.
(This system log example was gathered after a failed attempt to run user data - so we can see errors).






```
[    5.968846] systemd-journald[1030]: Received client request to flush runtime journal.
[    6.021934] ACPI: bus type drm_connector registered
[    6.569446] input: Power Button as /devices/LNXSYSTM:00/LNXPWRBN:00/input/input0
[    6.603438] vif vif-0 enX0: renamed from eth0
[    6.622703] ACPI: button: Power Button [PWRF]
[    6.628219] input: Sleep Button as /devices/LNXSYSTM:00/LNXSLPBN:00/input/input1
[    6.757550] SCSI subsystem initialized
[    6.808011] ACPI: button: Sleep Button [SLPF]

....   Output omitted   ....


[   15.347154] cloud-init[2087]: Cloud-init v. 22.2.2 running 'modules:final' at Tue, 12 Sep 2023
09:02:13 +0000. Up 15.20 seconds.
[   15.499653] cloud-init[2087]: + yum -y update
2023/09/12 09:02:14Z: Amazon SSM Agent v3.2.1377.0 is running
2023/09/12 09:02:14Z: OsProductName: Amazon Linux
2023/09/12 09:02:14Z: OsVersion: 2023
[   18.747963] cloud-init[2087]: Amazon Linux 2023 repository                    21 MB/s |  18 MB
00:00


Amazon Linux 2023
Kernel 6.1.49-69.116.amzn2023.x86_64 on an x86_64 (-)

ip-10-0-0-6 login: [   27.275607] cloud-init[2087]: Amazon Linux 2023 Kernel Livepatch repository
504 kB/s | 159 kB     00:00
[   28.894542] cloud-init[2087]: Dependencies resolved.
[   28.920089] cloud-init[2087]: Nothing to do.
[   28.924745] cloud-init[2087]: Complete!
[   29.020659] cloud-init[2087]: + amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
```

```
[   29.029063] cloud-init[2087]: /var/lib/cloud/instance/scripts/part-001: line 3:
amazon-linux-extras: command not found
[   29.050122] cloud-init[2087]: 2023-09-12 09:02:27,113 - cc_scripts_user.py[WARNING]: Failed to run
module scripts-user (scripts in /var/lib/cloud/instance/scripts)
[   29.070132] cloud-init[2087]: 2023-09-12 09:02:27,113 - util.py[WARNING]: Running

....   Extract ends   ....
```

Verify Fetching of AWS System Parameters

Using `sudo aws ssm get-parameter --name "/example/password" --region "us-east-1"` should successfully fetch resultant parameter values, provided all prerequisites are met.

OPTIONAL: Activating the Local MariaDb Server to see if it can accept queries

This will depend on system parameters being correct, and that the mariadb and mysql services are on. It also depends on having done the initial MariaDb setup. Some clues about doing this are seen in the service --status command

Some clues are seen after enabling the mariadb service and seeing feedback on status:

```
sudo systemctl enable mariadb.service
sudo systemctl start mariadb.service
# The status command below produces some interesting information and hints about setting up mariadb
as a server.
# Setting up Mariadb as a server for the as-found state is not mandatory, but it can be more
satisfying to see queries work locally, provided the parameter store is referring to the internal DNS
name of the EC2, and you have been through Mariadb setup.
# Achieving this now may also be informative for the final stanza of the Capstone Project where you
want to implement a server on-premise (a classroom VM on VMWare)
# sudo systemctl status mariadb.service
```

```
857]: you need to be the system 'mysql' user to connect.
857]: After connecting you can set the password, if you would need to be
857]: able to connect as any of these users with a password and without sudo
857]: See the MariaDB Knowledgebase at https://mariadb.com/kb
857]: Please report any problems at https://mariadb.org/jira
857]: The latest information about MariaDB is available at https://mariadb.org/.
857]: Consider joining MariaDB's strong and vibrant community:
857]: https://mariadb.org/get-involved/
10  3:48:26 0 [Note] /usr/libexec/mariadbd (mysqld 10.5.18-MariaDB) starting as process 26901 ...
db.service - MariaDB 10.5 database server.
~
~
```

From the info above we can see the initial user is “mysql”. So, provided the service on this same front-end instance is running, **you can verify you can connect to the service** this way:

```
[ec2-user@i-01dbda5d063b3d1c6 ~]$ mysql -u mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 9
Server version: 10.2.38-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]>
```

- If you’d like to make your “local” MariadB database fully functional - ie: the database on EC2 - you should research how to set up a new user and password, and how to create the initial database - which needs to be called `country_schema`.
- If not, skip this section and [move onto setting up the AWS RDS service in Phase 2](#).

This initial database is needed before you can import Shirley's database file to the local mariadb service. Use these commands to import:

```
cd /home/ec2-user
mysql -u <your dB username> -p --database country_schema < Countrydatadump.sql
```

You can verify creation of the local database, tables and content using mysql commands, as shown in the RDS verification phase below.

Updating AWS System Parameters for Local Database

Does Shirley's Site successfully Query Local Database? The way our User Data set up the web front-end means it fetches connection values from the AWS System Parameter Store. We need to set this up to point to the EC2-hosted database:

- /example/endpoint You can consider using EC-2 DNS reference or simply loopback address 127.0.0.1
- /example/username Use the username and password you created for the MariadB service
- /example/password
- /example/database This should be "country_schema"

These parameter values are case sensitive.

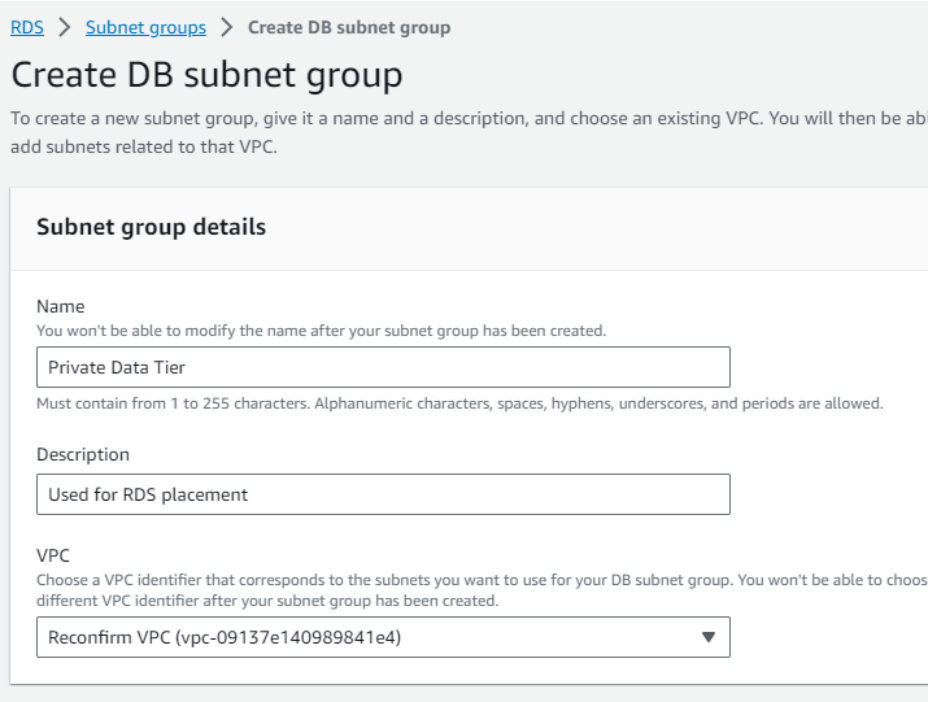
You may also need to open up port 3306 on the EC2 security group. Only do this temporarily, because normally you would not send mysql queries to the web front-end.

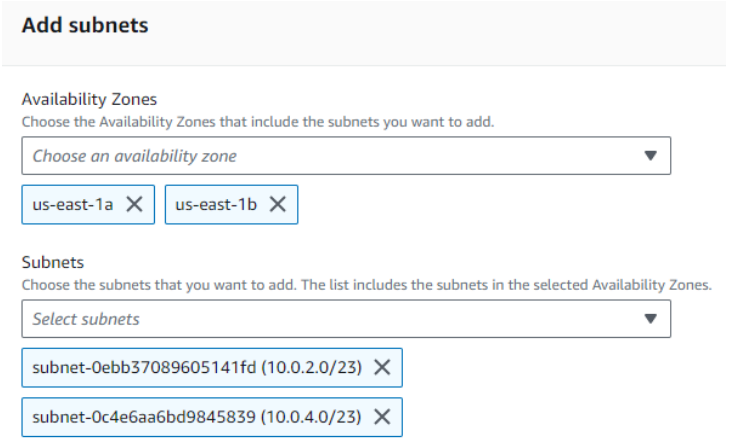
If you can't get the local database working on EC2, just move onto Phase 2.

Phase 2 - Decoupling the Data to AWS RDS Service

Create DB Subnet Group

We need to make sure your RDS database can only be deployed in your private subnets. To do this, predefine a database subnet group that only has private subnets as members.

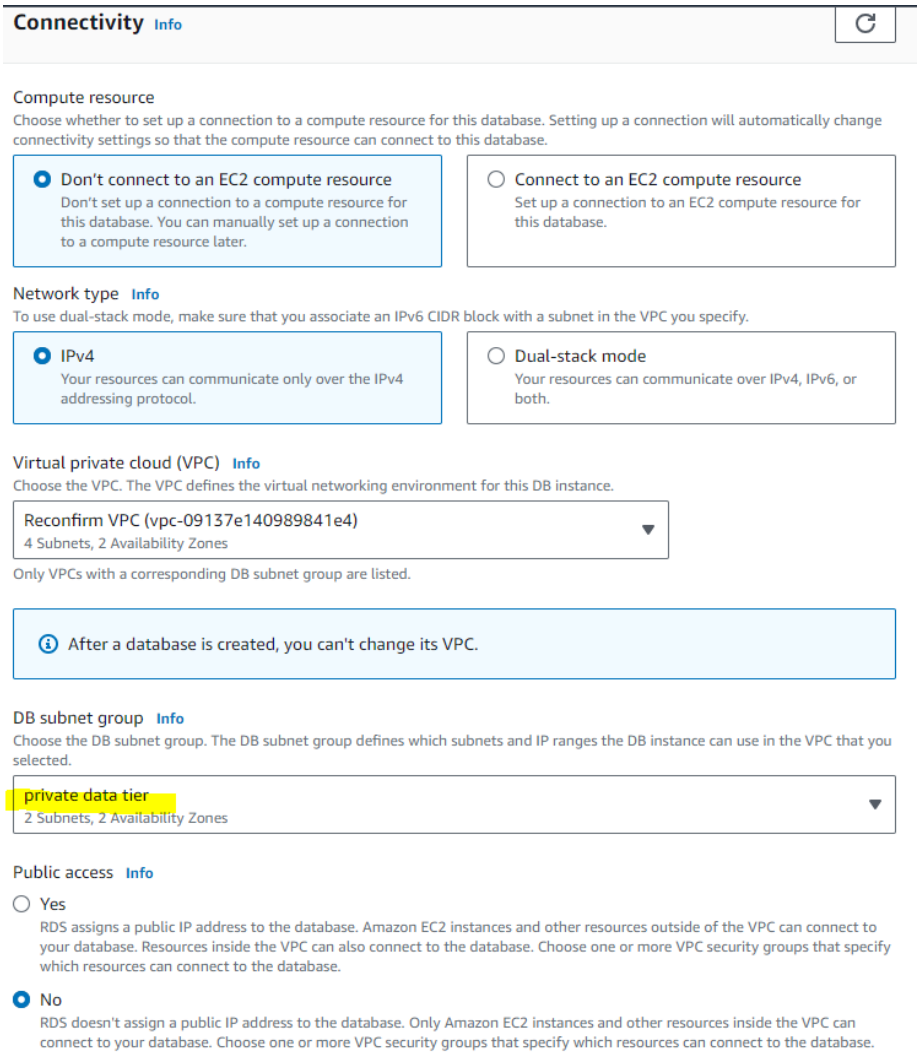




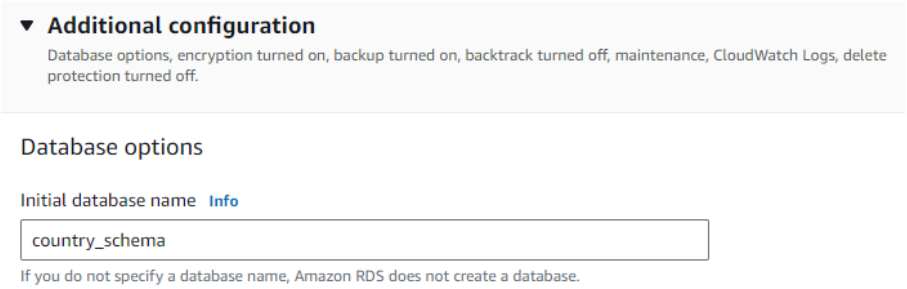
Setting Up AWS RDS Service

Setting up a MariaDb database using the Dev/Test option. Define the Initial Database Name as "country_schema". This setting is hidden under Additional configuration (second graphic below).

Please carefully note your created username and password. Note that username is case sensitive !!



Initial Database Name



Migrating Data to RDS

```
cd /home/ec2-user
mysql -u <your dB username> -p --host<RDS endpoint> --database country_schema <
Countrydatadump.sql
```

Using MySQL Commands to Verify Decoupled Database, Tables and Entries

To verify the database in RDS:

```
mysql -u admin -p --host<RDS endpoint> --database country_schema
Enter password:
```

To show what databases are present on your RDS service use:

```
MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| Social2            |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
```

Then to switch to a database use:

```
MySQL [(none)]> use Social2;
... Database changed
MySQL [Social2]>
```

Example of a Query:

```
MySQL [Social_Research]> select name, mobilephones from `countrydata_final` \g
+-------------------------------+--------------+
| name                          | mobilephones |
+-------------------------------+--------------+
| Afghanistan                   |            0 |
| Albania                       |        29791 |
| Algeria                       |        86000 |
```

Updating System Parameters

The parameters used are:

- /example/endpoint
- /example/username
- /example/password
- /example/database

These parameter values are case sensitive.

Does Shirley's Site successfully Query the decoupled RDS Database?

If your system parameters are correctly updated, and if you have successfully uploaded the database, then the web-site should be able to send queries to the web front-end now.

Setting Up Auto Scaler and Load Balancer

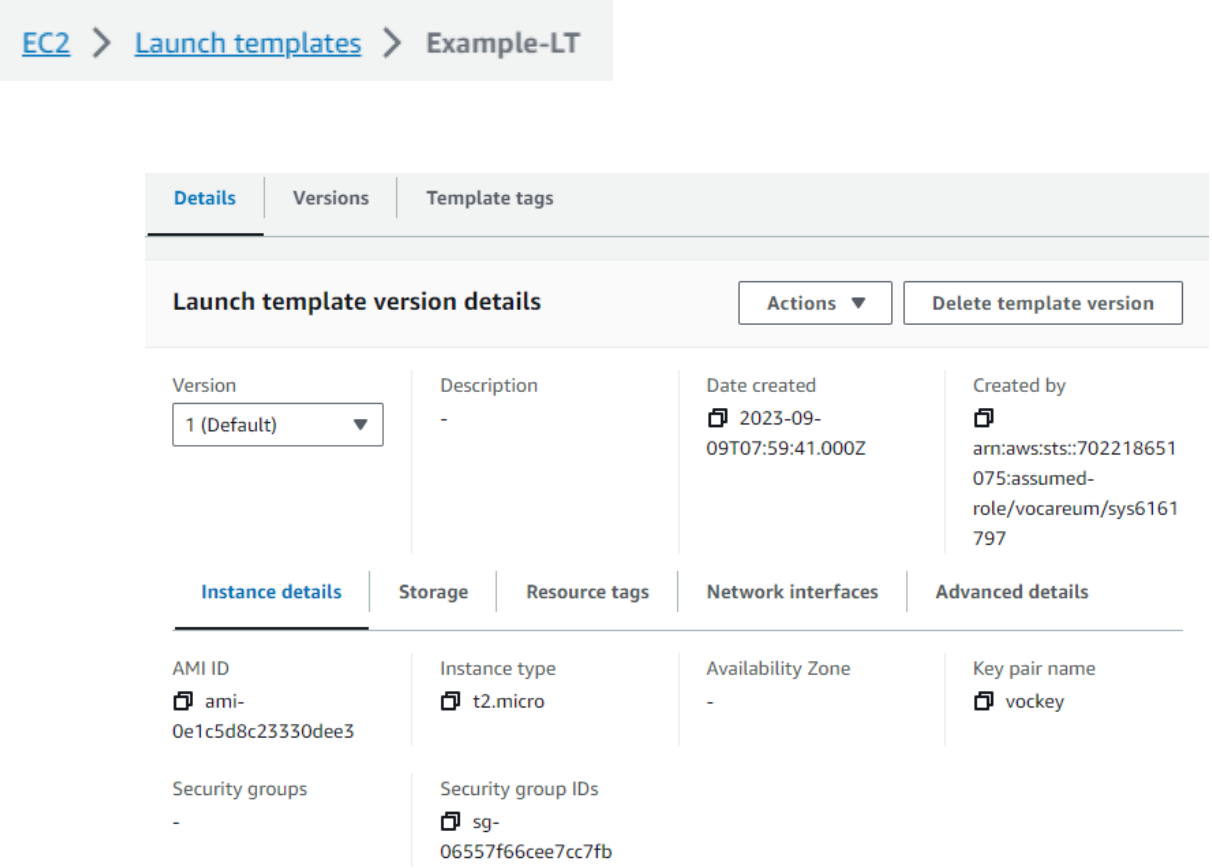
Set these up to harness the launch template described below.

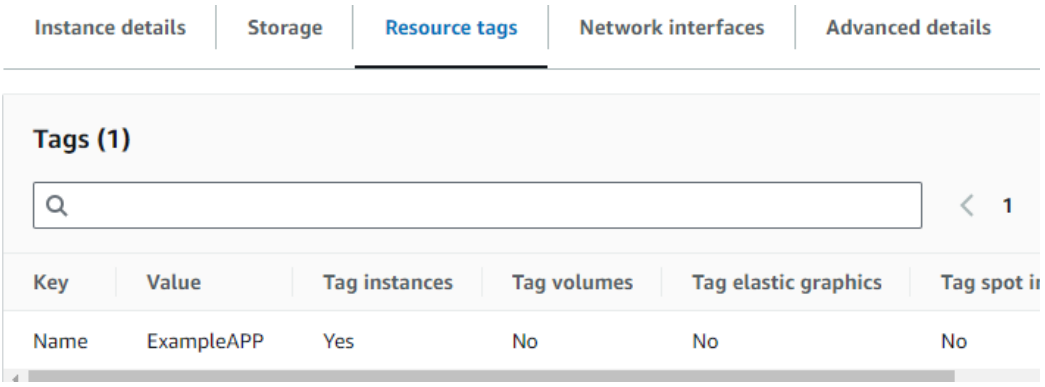
Setting up Launch Template

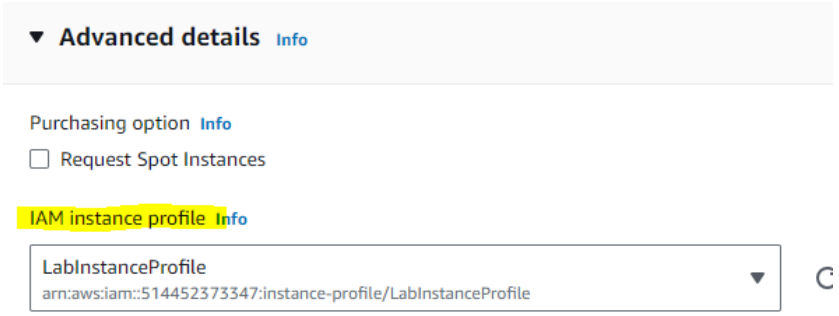
In the original Capstone project it came with a predefined launch template. We need to replicate this including using a certain version of AMI image - ami-0e1c5d8c23330dee3. *Several screenshots are provided below to illustrate the required Launch Template.*

Tip: When creating your Launch Template in AWS Console, an option is given that will give tips on how to set up a template suitable for an auto-scaler. I used this and it pointed out that it is better not to specify subnets for deployment in the template. That makes sense as subnets get specified under Load Balancer Network mapping. See the later section about this.

Required Launch Template Illustrated







We will also use the same user data script as part of this launch template:

```
#!/bin/bash -ex
yum -y update
amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
yum install -y httpd mariadb-server
chkconfig httpd on
service httpd start
cd /home/ec2-user
wget
https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACACAD-2/21-course-proje
ct/s3/Countrydatadump.sql
chown ec2-user:ec2-user Countrydatadump.sql
cd /var/www/html
wget
https://aws-tc-largeobjects.s3.us-west-2.amazonaws.com/CUR-TF-200-ACACAD-2/21-course-proje
ct/s3/Example.zip
unzip Example.zip -d /var/www/html/
chown -R ec2-user:ec2-user /var/www/html
```

Note that **Launch Templates use versioning**. This means if you alter your Launch Template for a second attempt - you will need to **ensure you have set the latest version of your template as default**.

Setting Up Load Balancer

As you are aware from labs in this course, the EC-2 Auto-Scaler and Application Load Balancer work in tandem.
Note: “Load Balancer Network Mapping” will specify which subnets your Auto-Scaler instances will deploy into. (e.g.: for your web-tier instances).

Scheme Info

Scheme can't be changed after the load balancer is created.

- (●) Internet-facing
  An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. Learn more
- ( ) Internal
  An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type Info

Select the type of IP addresses that your subnets use.

- (●) IPv4
  Recommended for internal load balancers.
- ( ) Dualstack
  Includes IPv4 and IPv6 addresses.

Network mapping Info

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info

Select the virtual private cloud (VPC) for your targets or you can create a new VPC. Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups.

Reconfirm VPC
vpc-09137e140989841e4
IPv4: 10.0.0.0/16

Mappings Info

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

[x] **us-east-1a (use1-az2)**

Subnet

subnet-01a57eb905e4e9de9 Reconfirm Public Subnet 1

IPv4 address

Assigned by AWS

[x] **us-east-1b (use1-az4)**

Subnet

subnet-05bfea07cfb218c29 Reconfirm Public Subnet 2

IPv4 address

Assigned by AWS

It will also be necessary to create a target group:

▼ Listener **HTTP:80** Remove

Protocol: HTTP : Port: 80 (1-65535)

Default action Info: Forward to Social-Research-Web (Target type: Instance, IPv4) HTTP

Create target group

Listener tags - *optional*

Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

Add listener tag

You can add up to 50 more tags.

Add listener

▼ Add-on services - *optional*

Additional AWS services can be integrated with this load balancer at launch. You can also add these and other services after your load balancer is created by reviewing the "Integrated Services" tab for the selected load balancer.

AWS Global Accelerator Info

☐ Create an accelerator to get static IP addresses and improve the performance and availability of your applications. Additional charges apply

▶ Load balancer tags - *optional*

Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.

Summary

Review and confirm your configurations. Estimate cost

Basic configuration Edit

Social

- Internet-facing
- IPv4

Security groups Edit

- Reconfirm Web-Tier Security Group sg-0d60614e54d5158c4

Network mapping Edit

VPC vpc-09137e140989841e4
Reconfirm VPC

- us-east-1a
  subnet-01a57eb905e4e9de9
  Reconfirm Public Subnet 1
- us-east-1b
  subnet-05bfea07cfb218c29
  Reconfirm Public Subnet 2

Listeners and routing Edit

- HTTP:80 defaults to Social-Research-Web

EC2 > Load balancers

Load balancers (1)

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

Actions | Create load balancer

Filter by property or value

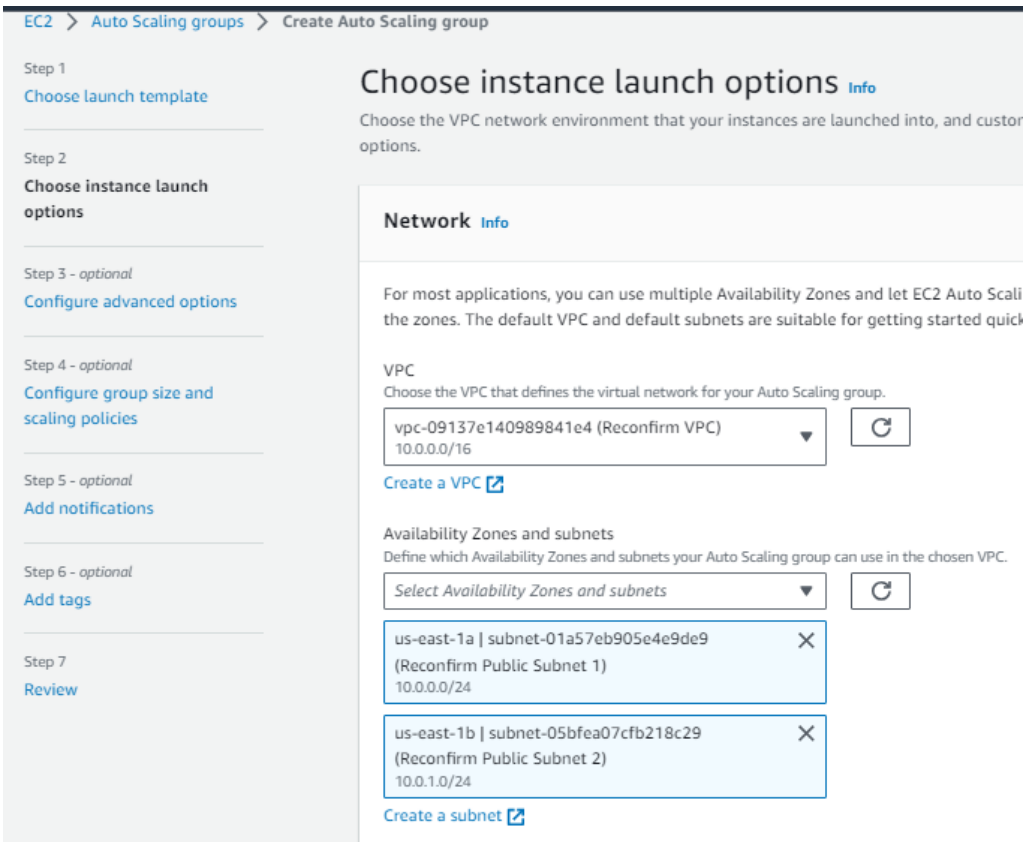
Social ✕ | Clear filters

1

Name	DNS name	State	VPC ID	Availability Zones	Type
Social	Social-1277953120.us-eas...	Provisioning	vpc-09137e140989841e4	2 Availability Zones	application

Setting Up Your Auto Scaling Group

Follow the guidelines of previous labs to set this up and associate with the Load Balancer and Target Groups.
Tip:



Testing Solution Under Load Stress /Auto-Scaling Proof

You need to prove that auto-scaling of your solution works in response to CPU load. Linux has a built in tool called `stress` that may be useful to you.

- Use SSH daisy-chaining via your Bastion Host to reach the initial two EC2s behind your load balancer - simultaneously. You may need duplicate Putty sessions to achieve this.
- Install `stress` then use the appropriate command to busy up the CPU on each. (You need to use `man` or other help options to find `stress` commands).
- See if your Auto-Scaler responds correctly to scale up the number of instances attached to your load balancer.
- Demonstrate this stress-test and auto scaling as part of your evidence video.

Set up VPN Connection to On-Premise (VMWare)

Refer back to our Custom VPN Lab to ensure you have the skills to get an IPsec VPN tunnel up between your AWS VPC and your on-premise VMWare machine using pfSense.

Change VPC Route Tables to Allow Data Migration to On-Premise

Migrate Data to On-Premise

Activate the On-Premise Database Service

Prepare VPC Routes and System Parameters for Fail-Over

Prove a Fail-Over of AWS Web Front-End to use On-Premise Data Tier