

REPORT ON THE USE OF MULTIPLE VIRTUAL MACHINES - USING AWS CLOUD

Name: Maurice Andrews

Student ID: 8833638

Student Username: andma315

Commit ID: 423f531 (Final ReadMe & Report committed after this)

Problem Description

Build an application using multiple virtual machines which interact.

Make use of at least one cloud-provided service in addition to the cloud provider facilitating the use of the virtual machines.

The application should be developed within a git repository, allowing other developers to modify and build upon your application.

Application Running in the Public Cloud

For my assignment I used vagrant files to build two virtual machine web servers and a virtual machine database.

Each virtual machine is an instance in Amazon's Elastic Compute Cloud (EC2). EC2 runs applications on the Amazon Web Services (AWS) infrastructure i.e. Amazon's EC2 service hosts the database, web servers and the accompanying web pages.

To reach the index page of Web Server 1 enter the following address in a web browser: `ec2-18-232-133-240.compute-1.amazonaws.com`

To reach the index page of Web Server 2 enter the following address in a web browser: `ec2-3-215-184-116.compute-1.amazonaws.com`

[The web address of the virtual database is: `ec2-3-221-160-177.compute-1.amazonaws.com`]

The web pages on the two web servers are designed to be used by two different types of users (with different security procedures - firewalls etc).

Web server 1 is to be used by the company's customers and web server 2 is designed to be used by the company's administration function and would only be accessible with login procedures.

Screenshots of three instances in EC2

The screenshot shows the AWS Management Console for the EC2 service in the us-east-1 region. The left sidebar contains navigation links for EC2 Dashboard, Events, Tags, Reports, Limits, INSTANCES, IMAGES, ELASTIC BLOCK STORE, and Lifecycle Manager. The main content area displays a table of three running EC2 instances.

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv4
	i-02927a28bd78305...	t2.micro	us-east-1a	running	2/2 checks ...	None	ec2-3-221-160-177.co...	3.22
	i-0b22c11caadc45b97	t2.micro	us-east-1a	running	2/2 checks ...	None	ec2-18-232-133-240.co...	18.2
	i-0dcbe7bacab9fb9b	t2.micro	us-east-1a	running	2/2 checks ...	None	ec2-3-215-184-116.co...	3.21

Below the table, there is a prompt: "Select an instance above".

Screenshots of images stored on S3

The screenshot shows the AWS Management Console for the S3 service, specifically the 'stainedglasswindow' bucket in the US East (N. Virginia) region. The 'Overview' tab is selected, showing a list of objects in the bucket.

Name	Last modified	Size	Storage class
astrological.jpg	Sep 25, 2019 12:42:06 PM GMT+1200	15.5 KB	Standard
coloured_tree.jpg	Sep 25, 2019 12:42:06 PM GMT+1200	14.1 KB	Standard
error.html	Sep 20, 2019 4:02:25 PM GMT+1200	65.0 B	Standard
gothic.jpg	Sep 25, 2019 12:42:06 PM GMT+1200	18.6 KB	Standard
red_plant.jpg	Sep 25, 2019 12:42:06 PM GMT+1200	13.3 KB	Standard
religious.jpg	Sep 25, 2019 12:42:06 PM GMT+1200	19.9 KB	Standard

Screenshots of web pages on Virtual Machine WebServer1 (note images stored on Amazon S3)

Document1 - Microsoft Word

Stained Glass Windows

Not secure | ec2-18-232-133-240.compute-1.amazonaws.com

Stained Glass Windows

We offer the following stained glass windows:

Window Code	Window Name
1	Astrological
2	Coloured Tree
3	Gothic
4	Red Plant
5	Religious

Select A Stained Glass Window from one of the links below.

[Astrological](#)

[Coloured Tree](#)

[Gothic](#)

[Red Plant](#)

[Religious](#)

This is the link to the Admin Page if you are logged in as Admin

[Admin](#)

Windows taskbar: File Explorer, Edge, Mail, Photos, HP, Outlook, Chrome, VLC, Word


System tray: 3:17 p.m. 2/10/2019

Gothic

Not secure | ec2-18-232-133-240.compute-1.amazonaws.com/gothic.php

Gothic

This is the order page for the Gothic Stained Glass Windows.



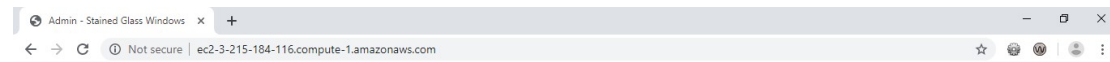
Below is a table of available sizes, the price for each size and the current stock available.

Window code	Window Size code	Size Name	Price	Stock Available
3	7	Small	229.99	5
3	8	Medium	259.99	4
3	9	Large	279.99	10

Windows taskbar: File Explorer, Edge, Mail, Photos, HP, Outlook, Chrome, VLC, Word

System tray: 3:18 p.m. 2/10/2019

Screenshots of web pages on Virtual Machine WebServer2 (note images stored on Amazon S3)



Admin Stained Glass Windows

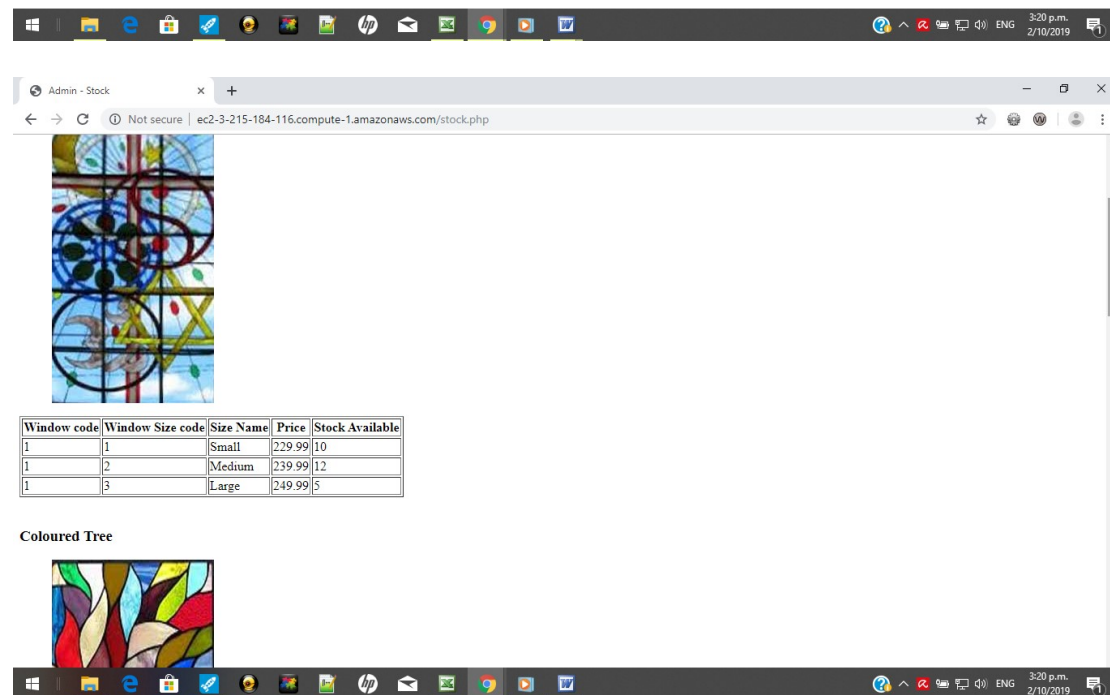
This is the Admin page for Stained Glass Windows

Click the following link to view current stock

[Stock](#)

This is the link to the Customers Page if you are logged in as Admin

[Customers](#)



Cloud Services Used in my Application

Amazon's Elastic Compute Cloud (EC2)

I used Amazon's Elastic Compute Cloud (EC2) service to host the database, web servers and accompanying web pages.

Amazon EC2 enables the tenant to manage their web site traffic loads i.e. to scale up or down. This reduces the need to forecast traffic.

This could be important for the assignment example if the organisation that owns the web site decides to have designated sales periods, which may increase the traffic loads dramatically for an infrequent space of time, depending on the success of the sale.

If the organisation is a start-up, this may be increasingly important, as the organisation that owns the web site will not have past sales history to forecast future sales.

Amazon's Simple Storage Service (S3)

I used Amazon's Simple Storage Service (S3) to store any images used by either web server. The images stored on S3 are linked to the web pages stored on EC2. S3 is a flexible storage device i.e. can boost servers holding the images when demand loads (web site traffic) increase.

This will ensure the images are accessible when there is a high level of load on the web site, which is important as the website can have infrequent increases in traffic (eg when there is a sale).

How I Incrementally Developed and Debugged my Application

I used the vagrant file created in Lab 9 - 'Using Vagrant to Deploy to EC2' to create an instance in Amazon's Elastic Compute Cloud (EC2).

I added code to this vagrant file, allowing the instance to host and display web pages, changing the /Vagrant permissions in the process.

I then stored the website images (accessible by both web servers) on Amazon's Simple Storage Service (S3) and linked the code in the relevant web pages to the ip addresses of these images.

I followed the procedure above to create another Web Server instance.

I then adjusted the code of the vagrant file to create a database virtual machine, adding a security group to allow outside virtual machine traffic to and from the database.

It was then a process of many vagrant up and vagrant destroys to get all the virtual machine addresses interacting correctly.

How the Virtual Machines and Other Cloud APIs are Used

The assignment uses three virtual machines (two web servers and one database). The two web servers host web pages, each suited for different types of users, with accompanying different security measures.

The web pages access images stored on Amazon's Simple Storage Service (S3).