



**Cloud Computing CSE4001**  
**Title: HOSTING A GAME WEBSITE IN AWS**

*Submitted by*

Reg no	Name
18BCE7183	I Ajay Kumar Reddy
18BCD7043	Malladi Adithya prahasith
18BCD7115	Ch Sai Sreekar

**Under the Guidance of**

**Prof. Ganesh Reddy K**

**SCOPE**

**VIT-AP**

**Abstract:**

Highly available and scalable web hosting can be a complex and expensive proposition. Traditional scalable web architectures have not only needed to implement complex solutions to ensure high levels of reliability, but they have also required an accurate forecast of traffic to provide a high level of customer service. Dense peak traffic periods and wild swings in traffic patterns result in low utilization rates of expensive hardware. This yields high operating costs to maintain idle hardware, and an inefficient use of capital for underused hardware. Amazon Web Services (AWS) provides a reliable, scalable, secure, and highly performing infrastructure for the most demanding web applications. This infrastructure matches IT costs with customer traffic patterns in real time. This whitepaper is for IT managers and system architects who look to the cloud to help them achieve the scalability to meet their on-demand computing needs.

**Introduction:**

In the traditional hosting model, you have to provision servers to handle peak capacity. Unused cycles are wasted outside of peak periods. Web applications hosted by AWS can leverage on-demand provisioning of additional servers, so we can constantly adjust capacity and costs to actual traffic patterns. Business have led to build and maintain infrastructure to run on-premises applications. With the Software as a service(SaaS) model, business can consume applications that are hosted online, enabling them to lower their costs by paying only for they use, enjoy seamless and painless upgrades in functionality, and integrate easily with their existing data and systems. Application providers who are building SaaS-based applications quickly learn that owning and operating the infrastructure on which these solutions are hosted can be expensive and complex, especially when customer demand is uncertain. Whether you are an enterprise looking for a cloud environment in which to deploy your existing on-premises solutions, or an application vendor evaluating a cloud platform on which to deploy a new application or SaaS offering, you should consider a lot of things in your mind before implementing.

## Process Steps:-

### LAUNCHING A VIRTUAL MACHINE INSTALLATION IN AWS STEPS:

#### **Step 1:** Choose an Amazon Machine Image (AMI)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance.

AMI :

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-08f63db601b82ff5f (64-bit x86) / ami-0e502bbbe5de26d28 (64-bit Arm)

#### **Step 2:** choose instance type

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications.

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, -, 1 GiB memory, EBS only)

#### **Step 3:** Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances

#### **Step 4:** Add Storage

Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes.

#### **Step 5:** Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.

A copy of a tag can be applied to volumes, instances or both.

Tags will be applied to all instances and volumes

#### **Step 6:** Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow Internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports.

**Step 7: Review and launch**

We need to have a keypair value, which we can setup in the step6, then download the key pair value which is saved as .pem file.

we use puttygen to change the file format from .pem to .ppk and then using the public ip of the instance we can access the ec2 instance and then upload the html file.

**Commands used for hosting the html file into instance cloudgame :**

```
yum update -y
#Install apache
yum install httpd -y
cd/var/www/html
```

**HTML CODE:-**

```
<!DOCTYPE html>
<html>
<head>
  <title>Basic Snake HTML Game</title>
  <meta charset="UTF-8">
  <style>
    html, body {
      height: 100%;
      margin: 0;
    }

    body {
      background: black;
      display: flex;
      align-items: center;
      justify-content: center;
    }
    canvas {
      border: 1px solid white;
    }
  </style>
</head>
<body>
  <canvas width="400" height="400" id="game"></canvas>
  <script>
    var canvas = document.getElementById('game');
    var context = canvas.getContext('2d');

    var grid = 16;
```

```

var count = 0;

var snake = {
  x: 160,
  y: 160,

  // snake velocity. moves one grid length every frame in either the x or y direction
  dx: grid,
  dy: 0,

  // keep track of all grids the snake body occupies
  cells: [],

  // length of the snake. grows when eating an apple
  maxCells: 4
};

var apple = {
  x: 320,
  y: 320
};

// get random whole numbers in a specific range
// @see https://stackoverflow.com/a/1527820/2124254
function getRandomInt(min, max) {
  return Math.floor(Math.random() * (max - min)) + min;
}

// game loop
function loop() {
  requestAnimationFrame(loop);

  // slow game loop to 15 fps instead of 60 (60/15 = 4)
  if (++count < 4) {
    return;
  }

  count = 0;
  context.clearRect(0,0,canvas.width,canvas.height);

  // move snake by it's velocity
  snake.x += snake.dx;
  snake.y += snake.dy;

  // wrap snake position horizontally on edge of screen

```

```

if (snake.x < 0) {
    snake.x = canvas.width - grid;
}
else if (snake.x >= canvas.width) {
    snake.x = 0;
}

// wrap snake position vertically on edge of screen
if (snake.y < 0) {
    snake.y = canvas.height - grid;
}
else if (snake.y >= canvas.height) {
    snake.y = 0;
}

// keep track of where snake has been. front of the array is always the head
snake.cells.unshift({x: snake.x, y: snake.y});

// remove cells as we move away from them
if (snake.cells.length > snake.maxCells) {
    snake.cells.pop();
}

// draw apple
context.fillStyle = 'red';
context.fillRect(apple.x, apple.y, grid-1, grid-1);

// draw snake one cell at a time
context.fillStyle = 'green';
snake.cells.forEach(function(cell, index) {

    // drawing 1 px smaller than the grid creates a grid effect in the snake body so you can see
    // how long it is
    context.fillRect(cell.x, cell.y, grid-1, grid-1);

    // snake ate apple
    if (cell.x === apple.x && cell.y === apple.y) {
        snake.maxCells++;

        // canvas is 400x400 which is 25x25 grids
        apple.x = getRandomInt(0, 25) * grid;
        apple.y = getRandomInt(0, 25) * grid;
    }
}

```

```

// check collision with all cells after this one (modified bubble sort)
for (var i = index + 1; i < snake.cells.length; i++) {

    // snake occupies same space as a body part. reset game
    if (cell.x === snake.cells[i].x && cell.y === snake.cells[i].y) {
        snake.x = 160;
        snake.y = 160;
        snake.cells = [];
        snake.maxCells = 4;
        snake.dx = grid;
        snake.dy = 0;

        apple.x = getRandomInt(0, 25) * grid;
        apple.y = getRandomInt(0, 25) * grid;
    }
}
});
}

```

```

// listen to keyboard events to move the snake
document.addEventListener('keydown', function(e) {
    // prevent snake from backtracking on itself by checking that it's
    // not already moving on the same axis (pressing left while moving
    // left won't do anything, and pressing right while moving left
    // shouldn't let you collide with your own body)

    // left arrow key
    if (e.which === 37 && snake.dx === 0) {
        snake.dx = -grid;
        snake.dy = 0;
    }
    // up arrow key
    else if (e.which === 38 && snake.dy === 0) {
        snake.dy = -grid;
        snake.dx = 0;
    }
    // right arrow key
    else if (e.which === 39 && snake.dx === 0) {
        snake.dx = grid;
        snake.dy = 0;
    }
    // down arrow key
    else if (e.which === 40 && snake.dy === 0) {
        snake.dy = grid;
    }
}

```

```

snake.dx = 0;
}
});

// start the game
requestAnimationFrame(loop);
</script>
</body>
</html>
save file : ctrl+x and Y

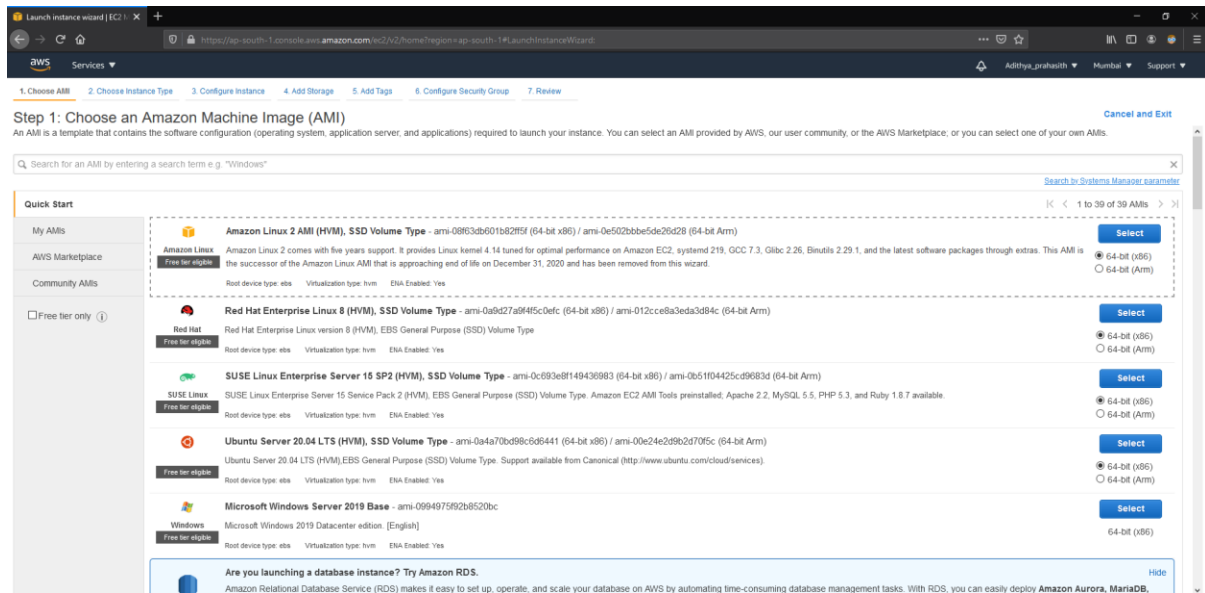
```

```

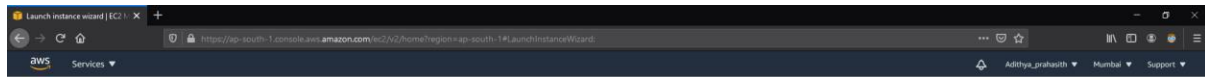
#start apache
sudo su
service httpd start
#check status
service httpd status
# for restarting apache at reboot
chkconfig httpd on

```

## Screenshots:-







## Step 2: Choose an Instance Type

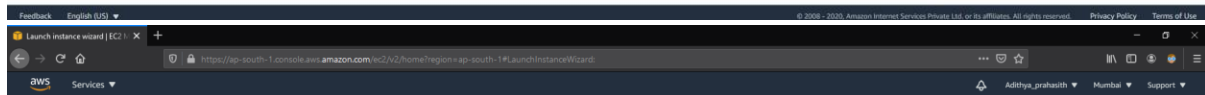
Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. Instances are virtual servers that can run applications. They have varying combinations of CPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your applications. [Learn more](#) about instance types and how they can meet your computing needs.

Filter by: **All instance families** **Current generation** **Show/Hide Columns**

Currently selected: t2.micro (- ECUs, 1 vCPUs, 2.5 GHz, ~, 1 GB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="checkbox"/>	t2	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="checkbox"/>	t2	t2.micro <small>Launch Instance</small>	1	1	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.small	1	2	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.medium	2	4	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.large	2	8	EBS only	-	Low to Moderate	Yes
<input type="checkbox"/>	t2	t2.xlarge	4	16	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t2	t2.2xlarge	8	32	EBS only	-	Moderate	Yes
<input type="checkbox"/>	t3	t3.nano	2	0.5	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.micro	2	1	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.small	2	2	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.medium	2	4	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.large	2	8	EBS only	Yes	Up to 5 Gigabit	Yes
<input type="checkbox"/>	t3	t3.xlarge	4	16	EBS only	Yes	Up to 5 Gigabit	Yes

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Configure Instance Details](#)



## Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

**Number of instances**  [Launch into Auto Scaling Group](#)

**Purchasing option** ☐ Request Spot instances

**Network**  [Create new VPC](#)

**Subnet**  [Create new subnet](#)

**Auto-assign Public IP**

**Placement group** ☐ Add instance to placement group

**Capacity Reservation**

**Domain join directory**  [Create new directory](#)

**IAM role**  [Create new IAM role](#)

**CPU options** ☐ Specify CPU options

**Shutdown behavior**

**Stop - Hibernate behavior** ☐ Enable hibernation as an additional stop behavior

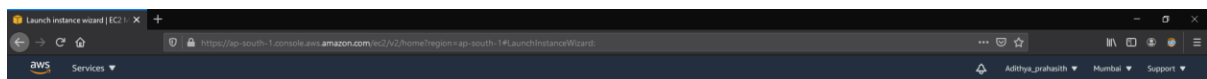
**Enable termination protection** ☐ Protect against accidental termination

**Monitoring** ☐ Enable CloudWatch detailed monitoring  
Additional charges apply.

**Tenancy**   
Additional charges will apply for dedicated tenancy

**Credit specification** ☐ Unlimited  
Additional charges may apply

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Storage](#)



#### Step 4: Add Storage

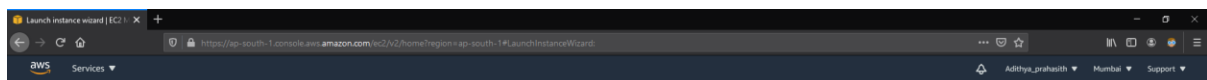
Your instance will be launched with the following storage device settings. You can attach additional EBS volumes and instance store volumes to your instance, or edit the settings of the root volume. You can also attach additional EBS volumes after launching an instance, but not instance store volumes. [Learn more](#) about storage options in Amazon EC2.

Volume Type <sup>(i)</sup>	Device <sup>(i)</sup>	Snapshot <sup>(i)</sup>	Size (GiB) <sup>(i)</sup>	Volume Type <sup>(i)</sup>	IOPS <sup>(i)</sup>	Throughput (MB/s) <sup>(i)</sup>	Delete on Termination <sup>(i)</sup>	Encryption <sup>(i)</sup>
Root	/dev/xvda	snap-0b7ac3c15dc29d594	8	General Purpose SSD (gp2)	100 / 3000	N/A	<input checked="" type="checkbox"/>	Not Encrypted

[Add New Volume](#)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

[Cancel](#) [Previous](#) [Review and Launch](#) [Next: Add Tags](#)

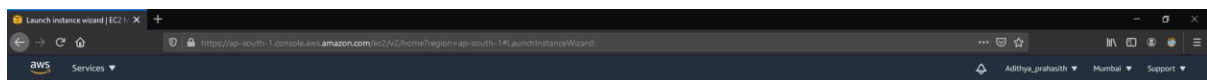


#### Step 5: Add Tags

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. A copy of a tag can be applied to volumes, instances or both. Tags will be applied to all instances and volumes. [Learn more](#) about tagging your Amazon EC2 resources.

Key <sup>(i)</sup> (128 characters maximum)	Value <sup>(i)</sup> (256 characters maximum)	Instances <sup>(i)</sup>	Volumes <sup>(i)</sup>
KEY	webserv	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

[Add another tag](#) (Up to 50 tags maximum)



#### Step 6: Configure Security Group

A security group is a set of firewall rules that control the traffic for your instance. On this page, you can add rules to allow specific traffic to reach your instance. For example, if you want to set up a web server and allow internet traffic to reach your instance, add rules that allow unrestricted access to the HTTP and HTTPS ports. You can create a new security group or select from an existing one below. [Learn more](#) about Amazon EC2 security groups.

Assign a security group: ☒ Create a new security group  
☐ Select an existing security group

Security group name:   
Description:

Type <sup>(i)</sup>	Protocol <sup>(i)</sup>	Port Range <sup>(i)</sup>	Source <sup>(i)</sup>	Description <sup>(i)</sup>
SSH	TCP	22	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop
HTTP	TCP	80	Custom 0.0.0.0/0	e.g. SSH for Admin Desktop

[Add Rule](#)

**Warning**  
Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

[Cancel](#) [Previous](#) [Review and Launch](#)

Launch instance wizard | EC2 | X

https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#LaunchInstanceWizard

Services

Addithya\_prahalathMumbaiSupport

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

Step 7: Review Instance Launch

You can also open additional ports in your security group to facilitate access to the application or service you're running, e.g., HTTP (80) for web servers. [Edit security groups](#)

AMI Details

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-08f53db601b82ff5f

Free tier eligible

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is a...

Root device type: ebsVirtualization type: hvm

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GiB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Security Groups

Security group name: launch-wizard-4

Description: launch-wizard-4 created 2020-12-10T11:28:21.568+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	*/0	

Instance Details

Storage

Tags

CancelPreviousLaunch

Launch instance wizard | EC2 | X

https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#LaunchInstanceWizard

Services

Addithya\_prahalathMumbaiSupport

1. Choose AMI2. Choose Instance Type3. Configure Instance4. Add Storage5. Add Tags6. Configure Security Group7. Review

Step 7: Review Instance Launch

Network: vpc-459/b12e  
Subnet: No preference (default subnet in any Availability Zone)  
EBS-optimized: No  
Monitoring: No  
Termination protection: No  
Shutdown behavior: Stop  
Stop / Hibernate behavior: Disabled  
Capacity Reservation: open  
IAM role: None  
Domain join directory: None  
Tenancy: default  
Credit specification: Standard  
Host ID  
Host resource group name  
Affinity: Off  
Kernel ID: Use default  
RAM disk ID: Use default  
Enclave: false  
Metadata accessible: Enabled  
Metadata version: V1 and V2 (token optional)  
Metadata token response hop limit: 1  
User data  
Assign Public IP: Use subnet setting (Enable)  
Assign IPv6 IP: Use subnet setting (Enable)  
Assign Carrier IP

Storage

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/xvda	snap-0b7ac3c15de29d...	8	gp2	100 / 3000	N/A	Yes	Not Encrypted

Tags

Key	Value	Instances	Volumes
KEY	webserver	1	1

CancelPreviousLaunch

aws

Services

Adithya\_prahasith

Mumbai

Support

1. Choose AMI

2. Choose Instance Type

3. Configure Instance

4. Add Storage

5. Add Tags

6. Configure Security Group

7. Review

Step 7: Review Instance Launch

AMI Details

Free tier eligible

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-08f63db01b82ff5f

Amazon Linux 2 comes with five years support. It provides Linux kernel 4.14 tuned for optimal performance on Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is a...

Root device type: ebs

Virtualization type: hvm

Edit AMI

Instance Type

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	-	1	1	EBS only	-	Low to Moderate

Edit instance type

Security Groups

Security group name

launch-wizard-4

Description

launch-wizard-4 created 2020-12-10T11:28:21.568+05:30

Type	Protocol	Port Range	Source	Description
SSH	TCP	22	0.0.0.0/0	
HTTP	TCP	80	0.0.0.0/0	
HTTP	TCP	80	...	

Edit security groups

Instance Details

Number of instances

1

Purchasing option

On demand

Network

ipc-4567b12e

Subnet

No preference (default subnet in any Availability Zone)

EBS-optimized

No

Monitoring

No

Termination protection

No

Shutdown behavior

Stop

Stop - Hibernate behavior

Disabled

Edit instance details

Cancel

Previous

Launch

Instances | EC2 Management

https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#instances:sort=instanceState

aws

Services

Adithya\_prahasith

Mumbai

Support

New EC2 Experience

EC2 Dashboard

Events

Tags

Limits

Instances

Instances

Instance Types

Launch Templates

Instances (3)

Info

Connect

Instance state

Actions

Launch instances

Filter instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
	cloudgame	i-086a26825b7775ce8	Running	t2.micro	Initializing	No alarms	ap-south-1a	ec2-13-233-237-96.ap...	13.233.237.96	-
	game	i-068e8dae069f839d	Stopped	t2.micro	-	No alarms	ap-south-1a	-	-	-
	ubuntu	i-0fe0388a3e2af95a8	Termina...	t2.micro	-	No alarms	ap-south-1b	-	-	-

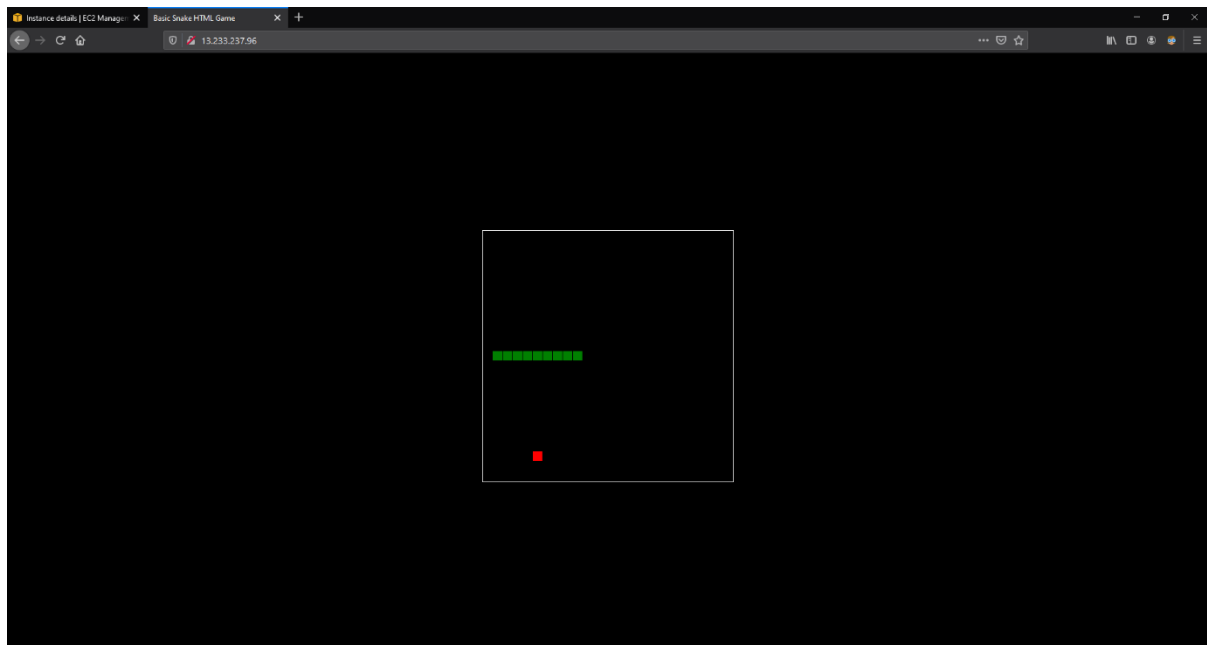
```
root@ip-172-31-42-143:/var/www/html
login as: ec2-user
Authenticating with public key "imported-openssh-key"
Last login: Wed Dec 9 18:58:16 2020 from 157.48.198.61

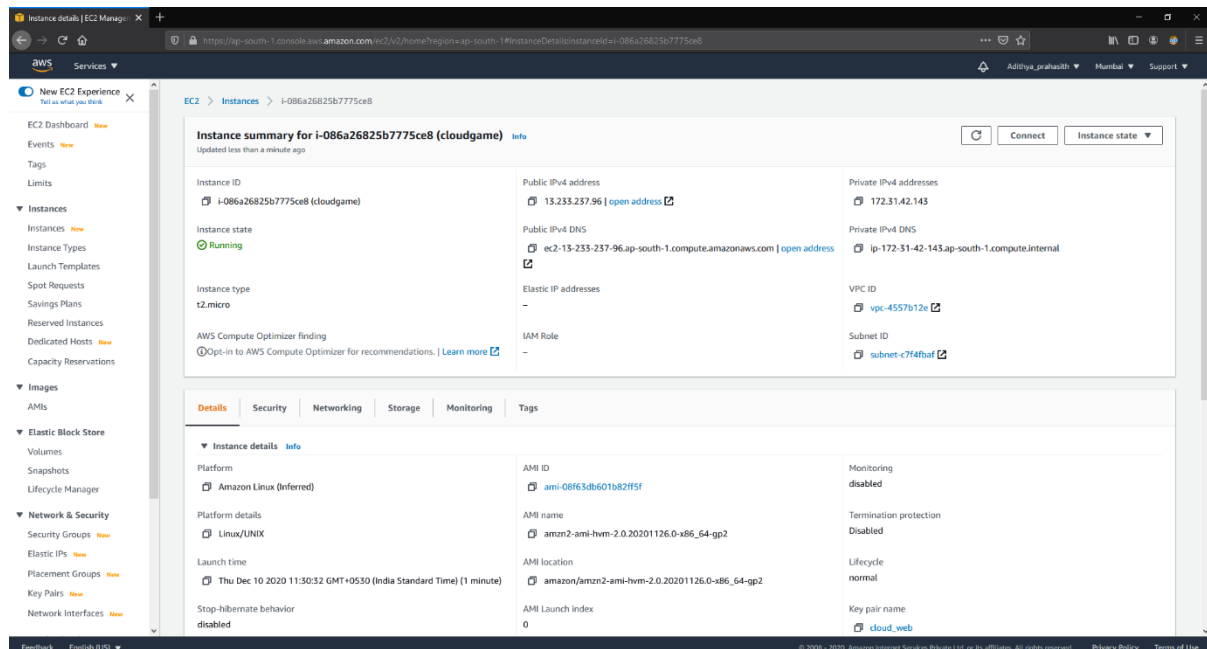
 _ _ | _ _ |
 _ | ( _ _ /
 _ | \ _ _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-42-143 ~]$ yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
You need to be root to perform this command.
[ec2-user@ip-172-31-42-143 ~]$ sudo su
[root@ip-172-31-42-143 ec2-user]# yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No packages marked for update
[root@ip-172-31-42-143 ec2-user]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package httpd-2.4.46-1.amzn2.x86_64 already installed and latest version
Nothing to do
[root@ip-172-31-42-143 ec2-user]# cd /var/www/html
[root@ip-172-31-42-143 html]# nano html
[root@ip-172-31-42-143 html]# sudo su
[root@ip-172-31-42-143 html]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-42-143 html]# service httpd status
Redirecting to /bin/systemctl status httpd.service
• httpd.service - The Apache HTTP Server
  Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
  Active: active (running) since Thu 2020-12-10 06:07:42 UTC; 10s ago
    Docs: man:httpd.service(8)
  Main PID: 3258 (httpd)
  Status: "Total requests: 0; Idle/Busy workers 100/0;Requests/sec: 0; Bytes served/sec: 0 B/sec"
  CGroup: /system.slice/httpd.service
          └─3258 /usr/sbin/httpd -DFOREGROUND
            └─3259 /usr/sbin/httpd -DFOREGROUND
              └─3260 /usr/sbin/httpd -DFOREGROUND
                └─3261 /usr/sbin/httpd -DFOREGROUND
                  └─3262 /usr/sbin/httpd -DFOREGROUND
                    └─3263 /usr/sbin/httpd -DFOREGROUND

Dec 10 06:07:42 ip-172-31-42-143.ap-south-1.compute.internal systemd[1]: Starting The Apache HTTP Server...
Dec 10 06:07:42 ip-172-31-42-143.ap-south-1.compute.internal systemd[1]: Started The Apache HTTP Server.
[root@ip-172-31-42-143 html]# chkconfig httpd on
Note: Forwarding request to 'systemctl enable httpd.service'.
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.servi
ce.
[root@ip-172-31-42-143 html]#
```





## Conclusion:

There are numerous architectural and conceptual considerations when you are contemplating migrating your web application to the AWS Cloud. The benefits of having a cost-effective, highly scalable, and fault-tolerant infrastructure that grows with your business far outstrips the efforts of migrating to the AWS Cloud. One can build their own web application based on the users requirements, their conditions and restrictions.