

Webinar Series on Remote Learning

TITLE

Launching an Amazon EC2 Linux Instance in an AWS Educate Starter Account

BRIEF SYNOPSIS

Check out this webinar to learn the steps to launching an Amazon EC2 Linux instance in the AWS Educate Starter Account. Amazon EC2 is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing simpler for developers.

Key takeaways:

- A brief introduction to Amazon EC2
- How to navigate the AWS Management Console
- How to configure an Amazon EC2 Linux instance within the AWS Educate Starter Account
- Launching an Amazon EC2 Linux instance
- Managing Amazon EC2 usage and reviewing and managing AWS Promotional Credit

SPEAKER

Ryan Little

SPEAKER BIO:

Technical Program Manager, AWS WWPS

DURATION

22 min 52 sec

TIMESTAMPS

[0:00](#) – Speaker Introductions

[1:13](#) – AWS Educate Educator Portal Overview

[4:02](#) – How to navigate the AWS Management Console from your AWS Educate Starter Account

[5:27](#) – Introduction to Amazon EC2

[11:08](#) - How to configure an Amazon EC2 Linux instance within the AWS Educate Starter Account

[15:38](#) – Launching an Amazon EC2 Linux instance

[21:50](#) – Managing Amazon EC2 usage and reviewing and managing AWS Promotional Credit

If you enjoyed this content, be sure to check out our live and on-demand webinars [here](#).

Webinar Series on Remote Learning

FAQs

Q: What is Amazon Elastic Compute Cloud (Amazon EC2)?

A: Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. It is designed to make web-scale computing easier for developers.

Q: What can I do with Amazon EC2?

A: Just as Amazon Simple Storage Service (Amazon S3) enables storage in the cloud, Amazon EC2 enables “compute” in the cloud. Amazon EC2’s simple web service interface allows you to obtain and configure capacity with minimal friction. It provides you with complete control of your computing resources and lets you run on Amazon’s proven computing environment. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use.

Q: How can I get started with Amazon EC2?

A: To sign up for Amazon EC2, click the “Sign up for This Web Service” button on the Amazon EC2 detail page. You must have an Amazon Web Services account to access this service; if you do not already have one, you will be prompted to create one when you begin the Amazon EC2 sign-up process. After signing up, please refer to the Amazon EC2 documentation, which includes our Getting Started Guide.

Q: Why am I asked to verify my phone number when signing up for Amazon EC2?

A: Amazon EC2 registration requires you to have a valid phone number and email address on file with AWS in case we ever need to contact you. Verifying your phone number takes only a couple of minutes and involves receiving a phone call during the registration process and entering a PIN number using the phone key pad.

Webinar Series on Remote Learning

FAQs

Q: What can developers now do that they could not before?

A: Until now, small developers did not have the capital to acquire massive compute resources and ensure they had the capacity they needed to handle unexpected spikes in load. Amazon EC2 enables any developer to leverage Amazon's own benefits of massive scale with no up-front investment or performance compromises. Developers are now free to innovate knowing that no matter how successful their businesses become, it will be inexpensive and simple to ensure they have the compute capacity they need to meet their business requirements.

The "Elastic" nature of the service allows developers to instantly scale to meet spikes in traffic or demand. When computing requirements unexpectedly change (up or down), Amazon EC2 can instantly respond, meaning that developers have the ability to control how many resources are in use at any given point in time. In contrast, traditional hosting services generally provide a fixed number of resources for a fixed amount of time, meaning that users have a limited ability to easily respond when their usage is rapidly changing, unpredictable, or is known to experience large peaks at various intervals.

Q: How do I run systems in the Amazon EC2 environment?

A: Once you have set up your account and select or create your AMIs, you are ready to boot your instance. You can start your AMI on any number of On-Demand instances by using the RunInstances API call. You simply need to indicate how many instances you wish to launch. If you wish to run more than 20 On-Demand instances, complete the Amazon EC2 instance request form.

If Amazon EC2 is able to fulfill your request, RunInstances will return success, and we will start launching your instances. You can check on the status of your instances using the DescribeInstances API call. You can also programmatically terminate any number of your instances using the TerminateInstances API call.

If you have a running instance using an Amazon EBS boot partition, you can also use the StopInstances API call to release the compute resources but preserve the data on the boot partition. You can use the StartInstances API when you are ready to restart the associated instance with the Amazon EBS boot partition.

In addition, you have the option to use Spot Instances to reduce your computing costs when you have flexibility in when your applications can run. Read more about Spot Instances for a more detailed explanation on how Spot Instances work.

If you prefer, you can also perform all these actions from the AWS Management Console or through the command line using our command line tools, which have been implemented with this web service API.

Webinar Series on Remote Learning

FAQs

Q: What is the difference between using the local instance store and Amazon Elastic Block Store (Amazon EBS) for the root device?

A: By using Amazon EBS, data on the root device will persist independently from the lifetime of the instance. This enables you to stop and restart the instance at a subsequent time, which is similar to shutting down your laptop and restarting it when you need it again. Alternatively, the local instance store only persists during the life of the instance. This is an inexpensive way to launch instances where data is not stored to the root device. For example, some customers use this option to run large web sites where each instance is a clone to handle web traffic.

Q: How quickly will systems be running?

A: It typically takes less than 10 minutes from the issue of the RunInstances call to the point where all requested instances begin their boot sequences. This time depends on a number of factors including: the size of your AMI, the number of instances you are launching, and how recently you have launched that AMI. Images launched for the first time may take slightly longer to boot.

Q: How do I load and store my systems with Amazon EC2?

A: Amazon EC2 allows you to set up and configure everything about your instances from your operating system up to your applications. An Amazon Machine Image (AMI) is simply a packaged-up environment that includes all the necessary bits to set up and boot your instance. Your AMIs are your unit of deployment. You might have just one AMI or you might compose your system out of several building block AMIs (e.g., web servers, app servers, and databases). Amazon EC2 provides a number of tools to make creating an AMI easy. Once you create a custom AMI, you will need to bundle it. If you are bundling an image with a root device backed by Amazon EBS, you can simply use the bundle command in the AWS Management Console. If you are bundling an image with a boot partition on the instance store, then you will need to use the AMI Tools to upload it to Amazon S3. Amazon EC2 uses Amazon EBS and Amazon S3 to provide reliable, scalable storage of your AMIs so that we can boot them when you ask us to do so.

Or, if you want, you don't have to set up your own AMI from scratch. You can choose from a number of globally available AMIs that provide useful instances. For example, if you just want a simple Linux server, you can choose one of the standard Linux distribution AMIs.

Webinar Series on Remote Learning

FAQs

Q: How do I access my systems?

A: The RunInstances call that initiates execution of your application stack will return a set of DNS names, one for each system that is being booted. This name can be used to access the system exactly as you would if it were in your own data center. You own that machine while your operating system stack is executing on it.

Q: Is Amazon EC2 used in conjunction with Amazon S3?

A: Yes, Amazon EC2 is used jointly with Amazon S3 for instances with root devices backed by local instance storage. By using Amazon S3, developers have access to the same highly scalable, reliable, fast, inexpensive data storage infrastructure that Amazon uses to run its own global network of web sites. In order to execute systems in the Amazon EC2 environment, developers use the tools provided to load their AMIs into Amazon S3 and to move them between Amazon S3 and Amazon EC2. See [How do I load and store my systems with Amazon EC2?](#) for more information about AMIs.

We expect developers to find the combination of Amazon EC2 and Amazon S3 to be very useful. Amazon EC2 provides cheap, scalable compute in the cloud while Amazon S3 allows users to store their data reliably.

Q: How many instances can I run in Amazon EC2?

A: You are limited to running On-Demand Instances per your vCPU-based On-Demand Instance limit, purchasing 20 Reserved Instances, and requesting Spot Instances per your dynamic Spot limit per region. New AWS accounts may start with limits that are lower than the limits described here.

If you need more instances, complete the [Amazon EC2 limit increase request form](#) with your use case, and your limit increase will be considered. Limit increases are tied to the region they were requested for.

Q: Are there any limitations in sending email from Amazon EC2 instances?

A: Yes. In order to maintain the quality of Amazon EC2 addresses for sending email, we enforce default limits on the amount of email that can be sent from EC2 accounts. If you wish to send larger amounts of email from EC2, you can apply to have these limits removed from your account by filling out [this form](#).

FAQs

Q: How quickly can I scale my capacity both up and down?

A: Amazon EC2 enables you to increase or decrease capacity within minutes, not hours or days. You can commission one, hundreds or even thousands of server instances simultaneously. When you need more instances, you simply call RunInstances, and Amazon EC2 will typically set up your new instances in a matter of minutes. Of course, because this is all controlled with web service APIs, your application can automatically scale itself up and down depending on its needs.

Q: What operating system environments are supported?

A: We are looking for ways to expand it to other platforms.

Q: Does Amazon EC2 use ECC memory?

A: In our experience, ECC memory is necessary for server infrastructure, and all the hardware underlying Amazon EC2 uses ECC memory.

FAQs

Q: How is this service different than a plain hosting service?

A: Traditional hosting services generally provide a pre-configured resource for a fixed amount of time and at a predetermined cost. Amazon EC2 differs fundamentally in the flexibility, control and significant cost savings it offers developers, allowing them to treat Amazon EC2 as their own personal data center with the benefit of Amazon.com's robust infrastructure.

When computing requirements unexpectedly change (up or down), Amazon EC2 can instantly respond, meaning that developers have the ability to control how many resources are in use at any given point in time. In contrast, traditional hosting services generally provide a fixed number of resources for a fixed amount of time, meaning that users have a limited ability to easily respond when their usage is rapidly changing, unpredictable, or is known to experience large peaks at various intervals.

Secondly, many hosting services don't provide full control over the compute resources being provided. Using Amazon EC2, developers can choose not only to initiate or shut down instances at any time, they can completely customize the configuration of their instances to suit their needs – and change it at any time. Most hosting services cater more towards groups of users with similar system requirements, and so offer limited ability to change these.

Finally, with Amazon EC2 developers enjoy the benefit of paying only for their actual resource consumption – and at very low rates. Most hosting services require users to pay a fixed, up-front fee irrespective of their actual computing power used, and so users risk overbuying resources to compensate for the inability to quickly scale up resources within a short time frame.

RESOURCES

[Amazon EC2](#) – Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides secure, resizable compute capacity in the cloud. It is designed to make web-scale cloud computing easier for developers.

[Amazon EC2 User Guide for Linux Instances](#) – Getting Started with Amazon EC2

[AWS Educate](#) – AWS Educate gives students and educators access to content and programs that enable them to skill up for cloud careers in growing fields. AWS Educate also connects companies hiring for cloud skills to qualified student job seekers with the AWS Educate Job Board.

[AWS Educate Office Hours for Educators and Students](#) – Webinars, office hours, and training sessions for educators and students

[AWS User Groups](#) - User groups are peer-to-peer communities which meet regularly to share ideas, answer questions, and learn about new services and best practices.



AWS Immersion Day

EC2 Hands-On Lab

Getting Started with Linux on Amazon EC2

EC2 Overview

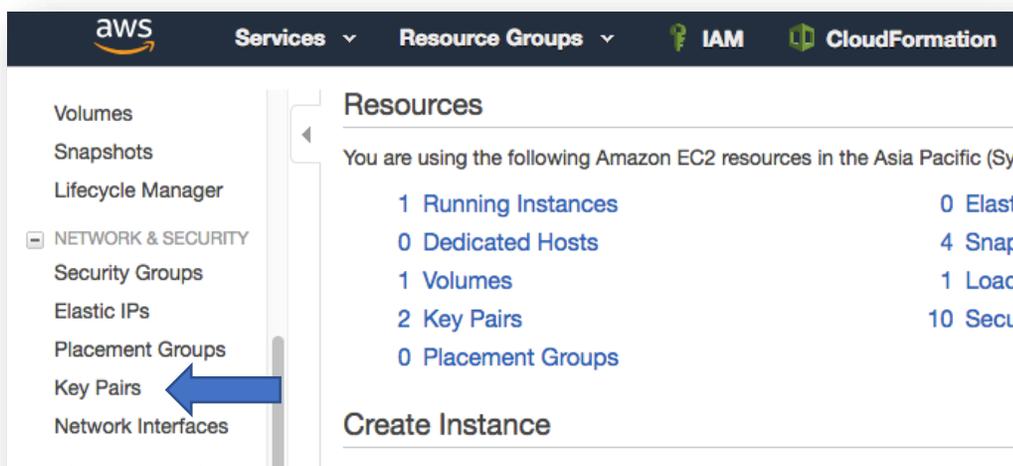
Amazon Elastic Compute Cloud (Amazon EC2) is a web service that provides resizable compute capacity in the cloud. Amazon EC2's simple web service interface allows you to obtain and configure capacity with minimal friction. Amazon EC2 reduces the time required to obtain and boot new server instances to minutes, allowing you to quickly scale capacity, both up and down, as your computing requirements change. Amazon EC2 changes the economics of computing by allowing you to pay only for capacity that you actually use.

This lab will walk you through launching, configuring, and customizing a web server on Amazon EC2 using the AWS Management Console.

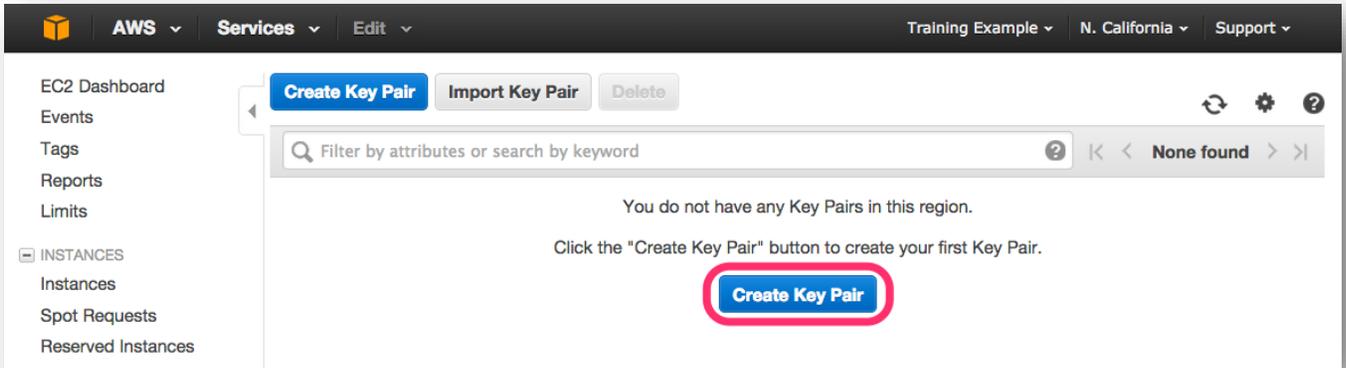
Create a new Key Pair

In this lab, you will create an EC2 instance as your web server. To manage the instance, you need to be able to connect to it via SSH. The following steps outline how to create a unique SSH keypair for this purpose.

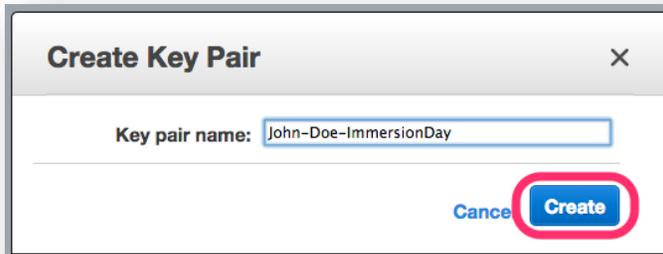
1. Sign into the AWS Management Console and open the Amazon EC2 console at <https://console.aws.amazon.com/ec2>.
2. In the upper-right corner of the AWS Management Console, confirm you are in the desired AWS region (e.g., N. Virginia).
3. Click on **Key Pairs** in the NETWORK & SECURITY section near the bottom of the leftmost menu. This will display a page to manage your SSH key pairs.



- To create a new SSH key pair, click the **Create Key Pair** button at the top of the browser window.



- In the resulting pop up window, type *[First Name]-[Last Name]-ImmersionDay* into the **Key Pair Name:** text box and click **Create**.



- The page prompts you to download the file "*[First Name]-[Last Name]-ImmersionDay.pem*" to the local drive. Follow your browser instructions to save the file to the default download location.
- Remember the full path to this .pem file you just downloaded. This file contains your private key for future SSH connections.



You will use the Key Pair you just created to manage your EC2 instances for the rest of the lab.

Launch a Web Server Instance

In this example, we will launch an Amazon Linux 2 instance, bootstrap Apache/PHP, and install a basic web page that will display information about our instance.

Sign into your AWS Management Console and choose EC2 from the Services menu.



Upon logging into your AWS Console, you should ALWAYS check which region you are operating in. This can be found in the top right of your Console window.

8. Click on Launch Instance

Create Instance

To start using Amazon EC2 you will want to launch a virtual server, known as an Amazon EC2 instance.

[Launch Instance](#)

Note: Your instances will launch in the US East (N. Virginia) region

9. In the **Quick Start** section, select the first Amazon Linux 2 AMI for 64-bit (x86) architecture and click **Select**. Note that the ami-xxxxxxx label and specific versions of the installed package may be different than in the image below.



Amazon Linux
Free tier eligible

Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-0b69ea66ff7391e80 (64-bit x86) / ami-09c61c4850b7465cb (64-bit Arm)

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.

Root device type: ebs Virtualization type: hvm ENA Enabled: Yes

[Select](#)

- 64-bit (x86)
- 64-bit (Arm)

10. In the Step 2. *Choose an Instance Type*, select the **t2.micro** instance size and click **Next: Configure Instance Details**.



If it isn't labeled "Free Tier Eligible", you may incur a charge!

11. On Step 3. **Configure Instance Details** page, expand the **Advanced Details** section located at the bottom of the page, then, copy/paste the script below into the **User Data** field. This shell script will install Apache & PHP, start the web service, and deploy a simple web page. Click **Next: Add Storage**.



'User data' is a method for bootstrapping your instance - Any code placed here will be executed the first time an instance is launched.

```
#include  
https://s3.amazonaws.com/immersionday-labs/bootstrap.sh
```

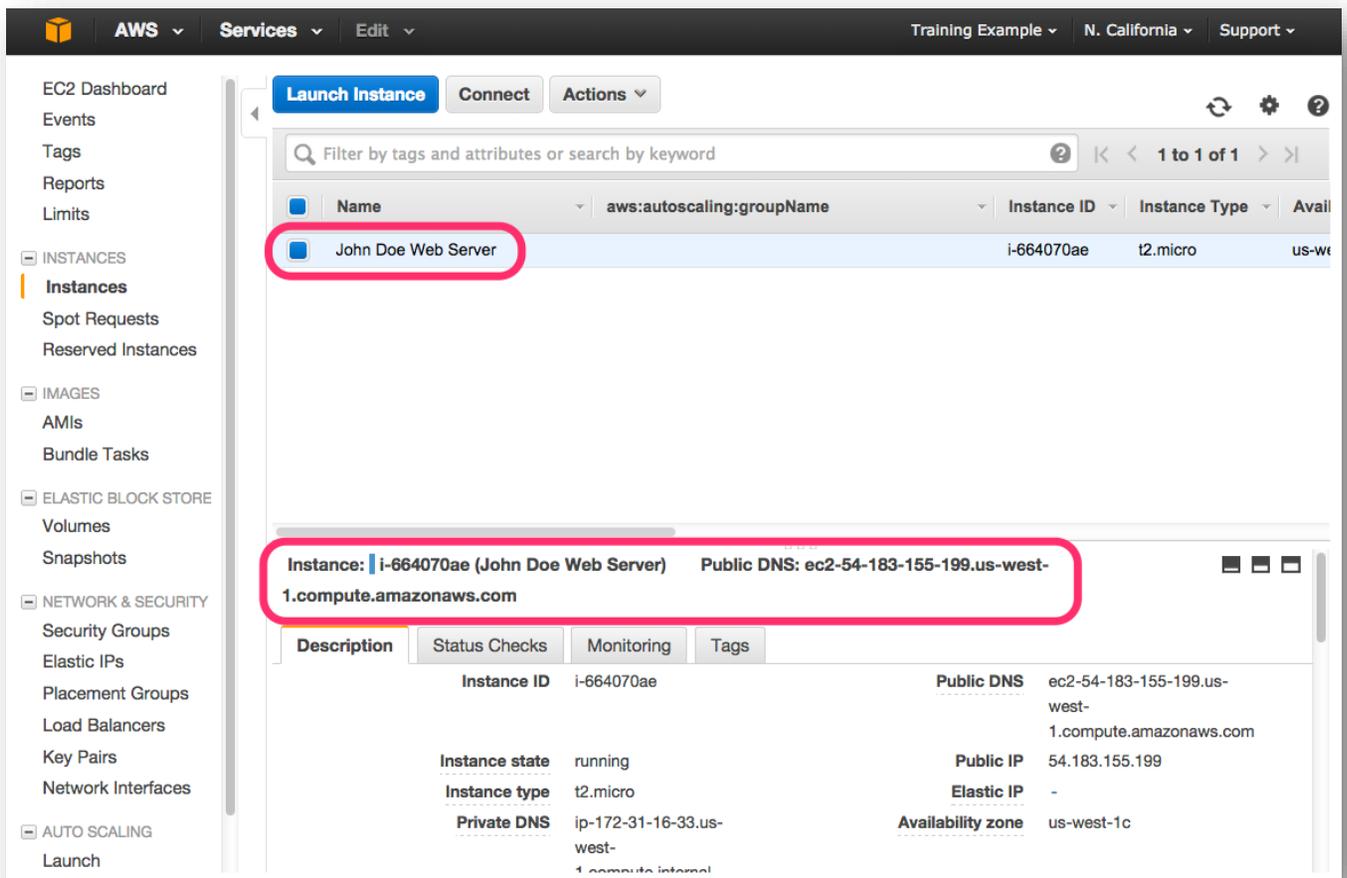
12. On this page you have the ability to modify or add storage and disk drives to the instance. For this lab, we will simply accept the storage defaults and click **Next: Add Tags**.
13. Here, you can choose a “friendly name” for your instance by clicking ‘Add Tag’, and entering “Name” for the Key part and “[Your Name] Web Server” for the Value part. This Name key, more correctly known as a **tag**, will appear in the console once the instance launches. It makes it easy to keep track of running machines in a complex environment. Click **Next: Configure Security Group**.
14. You will be prompted to create a new security group, which will be your firewall rules. On the assumption that we are building out a Web server, name your new security group “[Your Name] Web Tier”, and confirm an existing SSH rule exists which allows TCP port 22 from Anywhere. Click **Add Rule**.
15. Add now another rule with HTTP from the ‘Type’ dropdown menu, and confirm TCP port 80 is allowed from Anywhere (*you’ll notice, that “Anywhere is the same as ‘0.0.0.0/0’*). Click **Add Rule**.

Type <small>i</small>	Protocol <small>i</small>	Port Range <small>i</small>	Source <small>i</small>	Description <small>i</small>
SSH	TCP	22	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop ✕
HTTP	TCP	80	Anywhere 0.0.0.0, ::/0	e.g. SSH for Admin Desktop ✕

Add Rule

16. Click the **Review and Launch** button after configuring the security group.
17. Review your configuration and choices, and then click **Launch**.

18. Select the key pair that you created in the beginning of this lab from the drop-down and check the "I acknowledge" checkbox. Then click the **Launch Instances** button. Your instance will now be starting, which may take a moment.
19. Click the **View Instances** button in the lower right hand portion of the screen to view the list of EC2 instances. Once your instance has launched, you will see your Web Server as well as the Availability Zone the instance is in, and the publicly routable DNS name.
20. Click the checkbox next to your web server to view details about this EC2 instance.



Browse the Web Server

1. Wait for the instance to pass the Status Checks to finish loading.

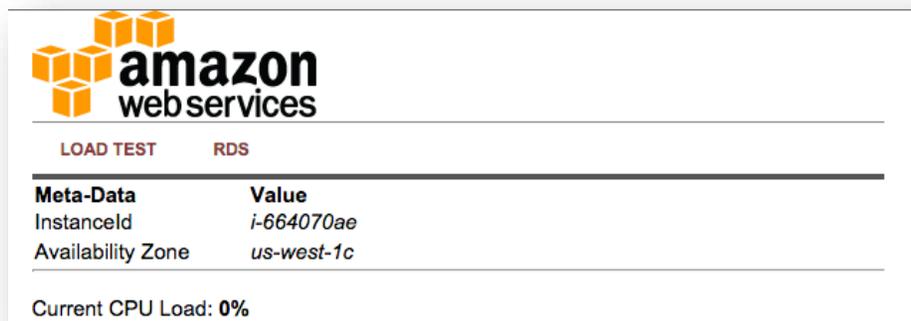
Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
John Doe Web Server	i-664070ae	t2.micro	us-west-1c	running	Initializing

Finished initializing

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks
John Doe Web Server	i-664070ae	t2.micro	us-west-1c	running	2/2 checks passed

Open a new browser tab and browse the Web Server by entering the EC2 instance's Public DNS name into the browser. The EC2 instance's Public DNS name can be found in the console by reviewing the "Public DNS" name line highlighted above.

You should see a website that looks like the following:



The screenshot shows the Amazon Web Services logo at the top. Below the logo, there are two tabs: "LOAD TEST" and "RDS". Under the "LOAD TEST" tab, there is a table with the following data:

Meta-Data	Value
Instanceld	i-664070ae
Availability Zone	us-west-1c

Below the table, it says "Current CPU Load: 0%".



If you don't see the web page (and you've waited a sufficient time for the instance to boot), try rebooting the instance via the console. Can you find it??

Great Job! You have deployed a server and launched a web site in a matter of minutes!!