

# Creating a Linux Virtual Workstation on AWS

*Estimated Time to Complete: 40 minutes*

## Overview

Welcome to this tutorial about setting up a streaming Linux virtual workstation. This step-by-step tutorial is written for creative studios interested in learning how to stream a powerful graphics workstation from the cloud. It has been created by artists including animators and composers, so its approach will resonate with an audience less familiar with cloud technology and using the AWS console. By the end of the tutorial, you'll have a fully cloud-based Linux workstation with Blender installed that demonstrates the power of creating content in the cloud.

## Prerequisites

Before you begin, please check the following prerequisites to make sure that you'll be able to successfully complete the tutorial.

## Region support

Before you jump into creating your Workstation in the Cloud, you should make sure that the geographic region you are in (or the one closest to you) supports all the services and instance types required.

A Region represents a specific geographic area around the world where AWS maintains a network and servers. Each Region has one or more Availability Zone, which are isolated locations inside the Region. For the best performance, you want AWS services to be located as geographically close to you as possible. With over 20 Regions, there is a good chance that there is a Region located near you. Having multiple Availability Zones in a Region provides additional reliability.

At the time of publication (April 2020), the following AWS Regions fully support all the services needed for this Virtual Workstation tutorial:

- **N. Virginia (us-east-1)**
- **Ohio (us-east-2)**
- **N. California (us-west-1)**

- **Oregon (us-west-2)**
- **Canada (ca-central-1)**
- **Frankfurt (eu-central-1)**
- **Ireland (eu-west-1)**
- **London (eu-west-2)**
- **Paris (eu-west-3)**
- **Stockholm (eu-north-1)**
- **Hong Kong (ap-east-1)**
- **Singapore (ap-southeast-1)**
- **Sydney (ap-southeast-2)**
- **Seoul (ap-northeast-2)**
- **Tokyo (ap-northeast-1)**
- **Mumbai (ap-south-1)**
- **Sao Paulo (sa-east-1)**

If you're not sure which Region is closest to you, please see the [Regions and Availability Zones](#) webpage. You may also want to use a website such as [ping.psa.fun](#) to measure the latency from your location to the different Regions. The Region with the **lowest latency** value is usually the one you want. For optimal performance, we recommend less than 20 ms latency.

**Please note:** If your nearest Region does not have all the required services, there may be another nearby Region that does. In addition, Regions are being updated all the time to add support for additional services and instance types. So in time, your closest Region may become supported. Please see the [Appendix](#) for information on the specific services that are required and links to webpages with the most up-to-date information on whether they are available in your nearest Region.

### **AWS account**

Now that you know whether your nearest AWS Region is supported, you should create an account, if you don't already have one: [Create an AWS account](#)

If you already have an AWS account, we have an extra question for you...is it your personal account or a linked account provided by your employer? If it's your personal account, then please be aware that there will be costs associated with using the AWS services required for this tutorial. We provide a rough estimate of those costs in the next section.


If your account is linked to your employer or another party, then you should check that they are prepared to accept responsibility for the costs involved. For more information on linked accounts and consolidated billing, please see this AWS documentation: [Consolidated Billing Process](#).

You will also need to verify that your account has sufficient permissions to access all the AWS services required to complete the tutorials.

### Account permissions

#### *If you are the owner of your account:*

If you just created a new AWS account or are the owner of your account, you may be logging in as the root user of your account. How can you tell? If you're logging in using an email address, rather than a user name, you're using the root account.

 **Note:** Rather than continue to login as the root user, we strongly suggest that you create a new user and login with those credentials instead. Why? If your root user email and password are ever stolen, then the person who stole them will have full access to your account and there's nothing you can do about it, unless you close the account. Creating a new user for yourself or others on your account allows you to have more control. You can specify exactly what permissions each user has and if that user information is ever lost, you can simply remove that user from your account.

If you're unfamiliar with how to create a new user and add permissions to it, we've provided instructions in the Appendix: [Creating an IAM User](#).

Even if you know how to create a user, please click the link above and follow the directions to add the specific permissions needed for this tutorial.

#### *If you are using an account provided by a third party:*

If your account is provided by your employer or a third party, you're likely already logging in as a user and not as the root. In this case, you do not need to create a new user, but you may need to request to have additional permission policies added in order to complete the tutorials. Please be aware that the owner of your account may have rules about what policies you can access.

Please click on the following link to a JSON file which lists all of the permissions needed for this tutorial: [Streaming Workstation Permissions Policy](#). It's best to check with the administrator for your account and request that these permissions be added to your user.

## On-Demand G Instance Quota

Many resources in AWS are subject to quotas, also called limits. Each resource has a quota that represents the maximum value of that resource you can use. These quotas are in place to ensure that AWS can provide highly available and reliable service to all our customers, but also to protect you against accidentally creating too many resources and incurring unexpected charges.

In this tutorial you will connect to a GPU-enabled virtual workstation (G4 instance). If you are using a new AWS account or have never used GPU instances before, your quota for that instance type may be at the default value of zero. Before continuing, you should check the value of your G instance quota and request an increase, if necessary. Please login to your AWS account and follow the instructions in the [Checking Your G Instance Quota](#) section of the Appendix.

## Keeping Track of Important Information

A few times in this tutorial you will be asked to refer back to information about components you created in a previous step, including names, IDs, IP addresses, and more. To make it easier to keep track of this information without having to constantly backtrack, we've created a cheat sheet for you to fill out as you go. Please follow this link to download it: [Important Information Cheat Sheet](#). It's a form that you can either fill out in a PDF viewer or print out and fill out by hand.

## Teradici Cloud Access Software

In order to work with your virtual workstation, you will need to have a way to interact with it from your personal desktop. Essentially you will be *streaming* the pixels and audio of the machine that is running in the cloud to your desktop, then sending keyboard and mouse data back. There are a couple of ways to do this, but one of the most common in the VFX and Animation industry is to use [Teradici Cloud Access Software](#).

For this tutorial, we will be using the [Teradici Linux AMI from the AWS Marketplace](#). This AMI already has Teradici installed. Instead of requiring a license, the cost is built into the hourly pricing for the AMI. You pay an extra \$0.50/hour on top of the normal hourly price for your instance.

Rather than pay by the hour for Teradici software, it is also possible to purchase an annual subscription. For more information, please see the [Teradici Linux AMI Subscription page](#). **Note:** If you are having trouble viewing the Teradici Marketplace or Subscription pages, you may need to login to your AWS account first. For help on logging in, please see the [Login to Your Account](#) section below.

## Managing Costs

AWS services are charged using a pay-as-you-go model. That means that you only pay for the individual services you use, for the time that you use them. For example, virtual workstations are charged per hour for each hour that they are running (whether you are logged in or not).

The ultimate cost of your Virtual Workstation will depend on how many hours it is running and how much of each service you consume. Costs for AWS services differ slightly from region to region. However, we understand that before even starting this tutorial you may want some idea of how much it will cost to complete. Depending on how much time you need to complete this tutorial, the cost will be about \$1-\$5 USD.

For a more accurate estimate of your cost, we recommend using the pricing tool located on the [Teradici Linux AMI page on the AWS Marketplace](#). To see a personalized estimate of your cost per hour, first select your closest region, then select the g4dn.4xlarge instance. We estimate that your usage will be 2 hours or less.

Our estimate above does not account for any existing credits that you may have in your account. In addition, if you decide to keep your infrastructure running after completing the tutorial, you will continue to accrue costs based on your hourly cost from above. Because these additional costs will vary depending on your individual needs, we do not attempt to estimate those costs here. This tutorial does not make use of any licensed software, so any licensing costs for additional software that you choose to use are also not covered.

If you have a contact in AWS Business Development or Sales or if you have a relationship with an AWS Business Partner, you can also contact them for help in estimating your costs. The [AWS Activate Program](#) may also be able to provide qualifying startups with credits that can be used to offset costs.

## Monitoring Your AWS Costs and Usage

As you work through the tutorial, you can keep an eye on your current costs using the [AWS Billing Console](#). In addition, AWS has many other tools for budgeting and tracking usage. Please see the following blog post for more information and helpful tips: [Keeping an Eye on Your AWS Costs and Usage](#).

## Getting Started: Using the AWS Console

Ready to begin? We're going to begin by introducing you to the **AWS Console**, which is your main interface to all AWS services.

In this tutorial, we're going to perform one of the most basic AWS tasks: launching an EC2 instance. EC2 stands for **Elastic Compute Cloud**. It's the AWS service that you use to launch the virtual computer that you will use for your Virtual Workstation.


An **instance** is a single virtual computer in EC2.

### Log in to Your Account

- If you haven't already, follow this link to [log in to your AWS account](#).
- As a reminder, make sure you log in as an IAM user and not as the root user for your account. If you need to create a new user, please see the directions in the Appendix: [Creating an IAM User](#).

### Fill Out Your Cheat Sheet

- We'll put reminders in *italics* to prompt you to fill out the [Important Information Cheat Sheet](#) as you go.
- *Now is a good time to fill out the cheat sheet with your AWS Account email address and account ID or alias, as well as your IAM username.*

 *Note:* You'll also need to remember your password, but for security reasons we don't recommend writing that down on the cheat sheet.

### Set Your Region

Next, you should check that your Region is set correctly. After logging in, your current Region is listed in the top right corner of the AWS Console, to the left of the Support drop down menu.



- To change your Region, click the down arrow next to the current Region and then select a different Region from the list.
- As a reminder, for this tutorial you must select a supported Region from the [list](#) above.
- *Please take a moment to note the region you are using (e.g. N. Virginia (us-east-1)) in the [Important Information Cheat Sheet](#).*

### Review EC2 Quotas

You probably already checked your service quota limit while going through the [Prerequisites](#) section at the beginning of the tutorial. If you have already verified that you have an applied quota value of at least 16, then you can skip to the next section.

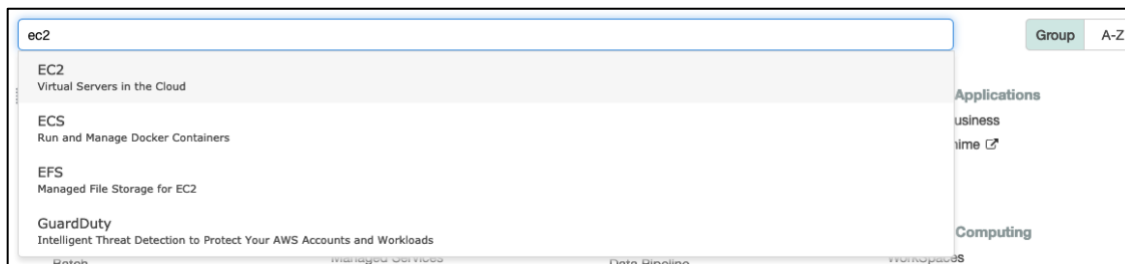
If you haven't checked your quota limit or if you needed to submit a quota increase request, please refer to the [Checking Your G Instance Quota](#) section in the Appendix to check it before continuing.

## Using AWS Marketplace AMIs


Instead of installing the Teradici Cloud Access Software from scratch on a basic Linux instance, we're going to start with an Amazon Machine Image (AMI) from the AWS Marketplace that already has the software installed and configured. The AWS Marketplace hosts thousands of AMIs that have already been created with specific software installed. Using one of these AMIs will save you the time that would have spent installing and troubleshooting software on your own. All you need to do is launch an instance and it will already be ready to go!

## Launch an EC2 Instance

- Click the **Services** drop down menu, then in the search field type **ec2**



- Select **EC2** from the list of services
- On the **EC2 Dashboard** page that appears, click the **Instances** link in the left side panel.
- On the **Instances** page, click the **Launch Instance** button.
- For **Step 1: Choose an Amazon Machine Image (AMI)**, enter **Teradici** in the search box and hit <enter>.
- On the left, below the search box, click on **AWS Marketplace** to list only AMIs that exist in the AWS Marketplace.
- In the list, find the entry for **Teradici Cloud Access Software for CentOS 7** and click the **Select** button next to it. Note: If you get an error after clicking on the Select button, make sure that you added all the necessary permissions to your user. Please the [Creating an IAM User](#) section of the Appendix for more details on adding permissions.



Select

**Teradici Cloud Access Software for CentOS 7**

★★★★★ (1) | 20.01.1 | By Teradici

Starting from \$0.50/hr or from \$240.00/yr (95% savings) for software + AWS usage fees

Linux/Unix, CentOS 7.6 | 64-bit (x86) Amazon Machine Image (AMI) | Updated: 3/6/20

Teradici Cloud Access Software enables secure remote visualization of workstations and graphics-intensive Windows and Linux applications from Amazon G3 instances using the PCoIP protocol

[More info](#)

- A pricing list for the various instance types will come up, showing you what the costs will be.
- This page also has links to more information about the Teradici software as well as usage instructions, but don't worry, we'll walk you through all the steps below!
- Click **Continue**
- For **Step 2: Choose an Instance Type**, open the **All instance types** drop down menu and select **GPU instances**

### 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 vary in vCPU, memory, storage, and networking capacity, and give you the flexibility to choose the appropriate mix of resources for your application.

Filter by: All instance types Current generation [Show/Hide Columns](#)

Currently **4 vCPUs, 2.7 GHz, Intel Xeon E5-2686 v4, 30.5 GiB memory, EBS only**

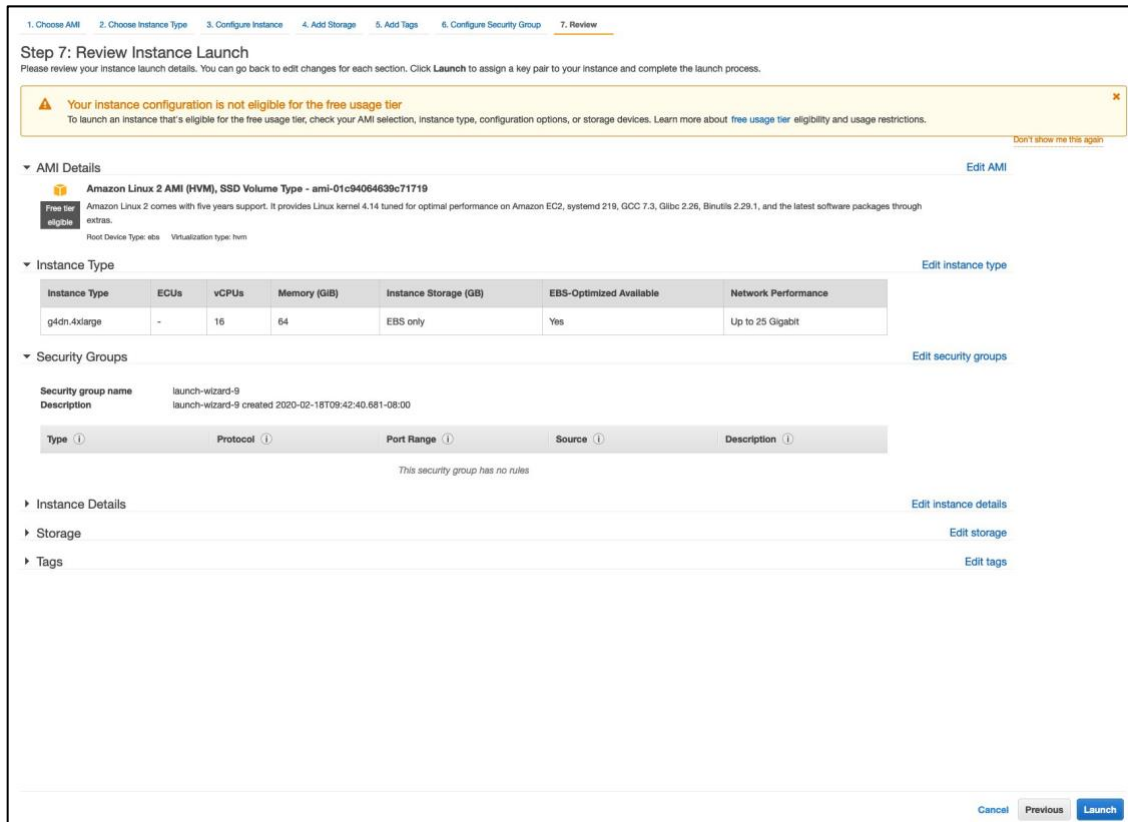
**Note:** The **large** instance (or larger) for the best experience with this product.

	Instance type	vCPUs	Memory (GiB)	Instance Storage (GB)
	nano	1	0.5	EBS only
	micro	1	1	EBS only

- All instance types
- Micro instances
- General purpose
- Compute optimized
- FPGA instances
- GPU instances
- Machine learning ASIC instances
- Memory optimized
- Storage optimized

- Next, choose a **g4dn.4xlarge** instance type and click the **Review and Launch** button.
- This will take you to **Step 7. Review Instance Launch**. A bunch of default settings have been filled in for you already. Feel free to look around at the details on this page. You may see a lot of unfamiliar terms, but don't worry. We'll explain anything that you do need to know about as you go through the tutorial.







- Next click the **Launch** button at the bottom of the screen.
- In the **Select an existing key pair or create a new key pair** popup window, select **Create a new key pair** from the first drop down menu and then enter a key pair name (e.g., **workstation-keypair**)
  - A **key pair** consists of public and private key files that are used to encrypt data between two computers. AWS stores the public key file, but you need to store the private key file.
- After entering your key pair name, be sure to click the **Download Key Pair** button
  - **Important:** You must download the private key file when you create a new key pair and store it securely on your local computer. You will not have another chance to download the private key file, so save it in a safe place on your computer where you can find it again!
  - Since we're connecting to our instance with Teradici, we actually won't need to use the private key file because Teradici's PCoIP protocol already encrypts the data for us. However, it's still a good idea to download the private key file just in case you ever need to use something else, such as SSH, to connect to your instance.

- You should also note the name of the key pair file in the [Important Information Cheat Sheet](#) for future reference.
- After downloading the private key file click the **Launch Instances** button.

### View the Status of your instance

- On the **Launch Status** page click the **View Instances** button at the bottom right corner of the page.
- This will take you back to the **Instances** page which will show a list of your running instances and their status. Initially, you will see the **Instance State** listed as **pending**, before changing to **running**, and the **Status Checks** field will start out as **Initializing**.

Instance Type ▾	Instance State ▾	Status Checks
g4dn.4xlarge	 running	 Initializing

- After about 5-10 minutes the **Status Checks** field will change to **2/2 checks passed** and your instance will be ready. While you're waiting for your instance to finish initializing, please continue with the steps below.

### Find the Instance ID and IPv4 Public IP

- Click on your instance to see more details in the panel at the bottom of the screen. You'll see many terms and acronyms like **VPC**, **Subnet** and **Security groups**, but don't worry, for right now you just need to find the **Instance ID** and the **IPv4 Public IP**
  - Note both the **Instance ID** (ex. *i-xxxxxxxxxxxxxxxxxx*) and the **IPv4 Public IP** (ex: *xx.xx.xxx.xx*) in the [Important Information Cheat Sheet](#) for use later.

### Download the Teradici Client to your local machine




In order to connect to your instance, you'll need to download and install the Teradici client software to your local computer.

- On your local computer, go back to the [Teradici Cloud Access Software webpage](#).
- In the **PCoIP Clients section**, download the version of the **Software Client** for the OS of your local machine.

### PCoIP Clients

PCoIP clients enable you or your users to connect securely to a remote desktop.

### Software Clients

 Software Client for Windows 20.01	 Software Client for macOS 20.01	 Software Client for Linux 20.01
--	--	--

- In the popup window, click **Download**, accept the end use license agreement and click **Agree and download**
- **Save** the installer to your local machine and then run it when the download is complete.
  - Proceed through the installation prompts. If you would like a desktop shortcut, make sure to check the box next to that option at the end of the install process.

### Check the Status of Your Instance

After installing the client software, you should check if your instance has finished initializing.

- Back in the AWS Console, find your instance again in the list of instances.
- Once the **Status Checks** field shows **2/2 checks passed**, you are good to go.

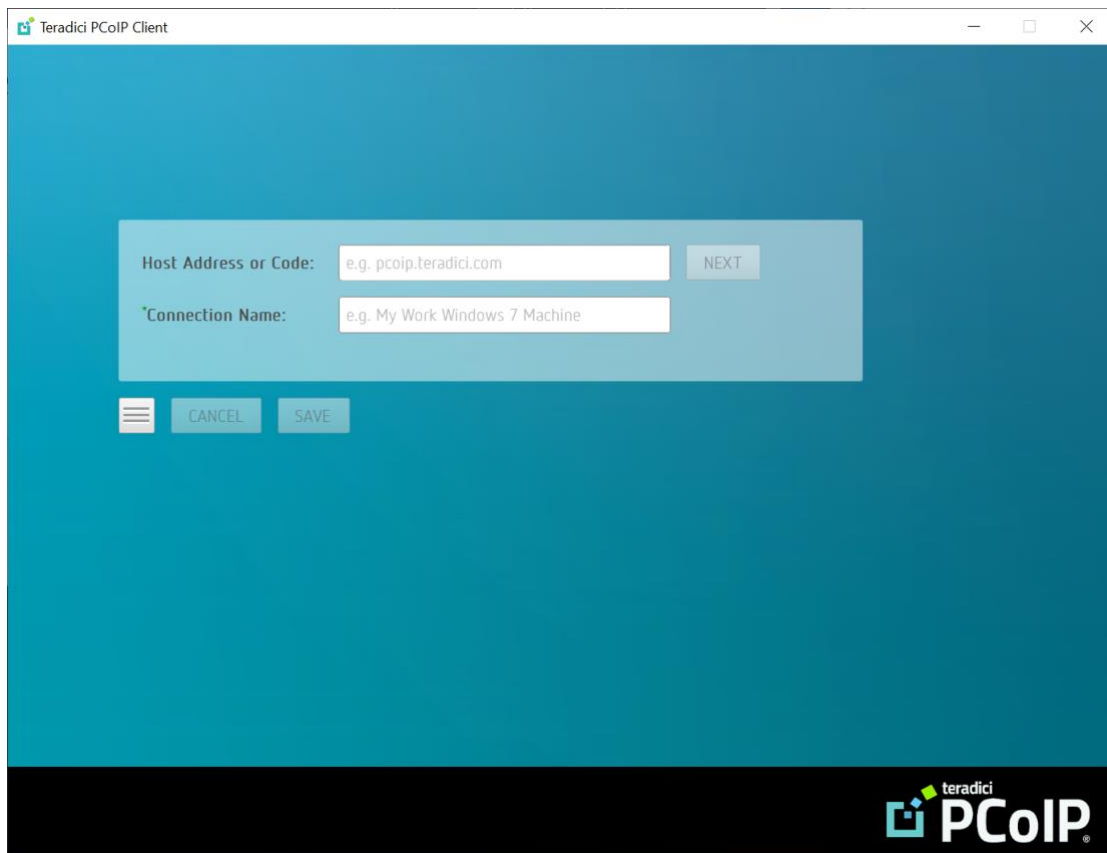
Instance Type	Instance State	Status Checks
g4dn.4xlarge	 running	 2/2 checks passed

- If your instance is still initializing, please wait a few more minutes for it to finish before proceeding.


### Connect using Teradici

Now that you have the client software installed locally and your instance has finished initializing, you're all ready to connect to it!

- Run the client on your local computer by double-clicking the **desktop shortcut** or by going to **Start**→**Teradici**→**PCoIP Client** (Windows)

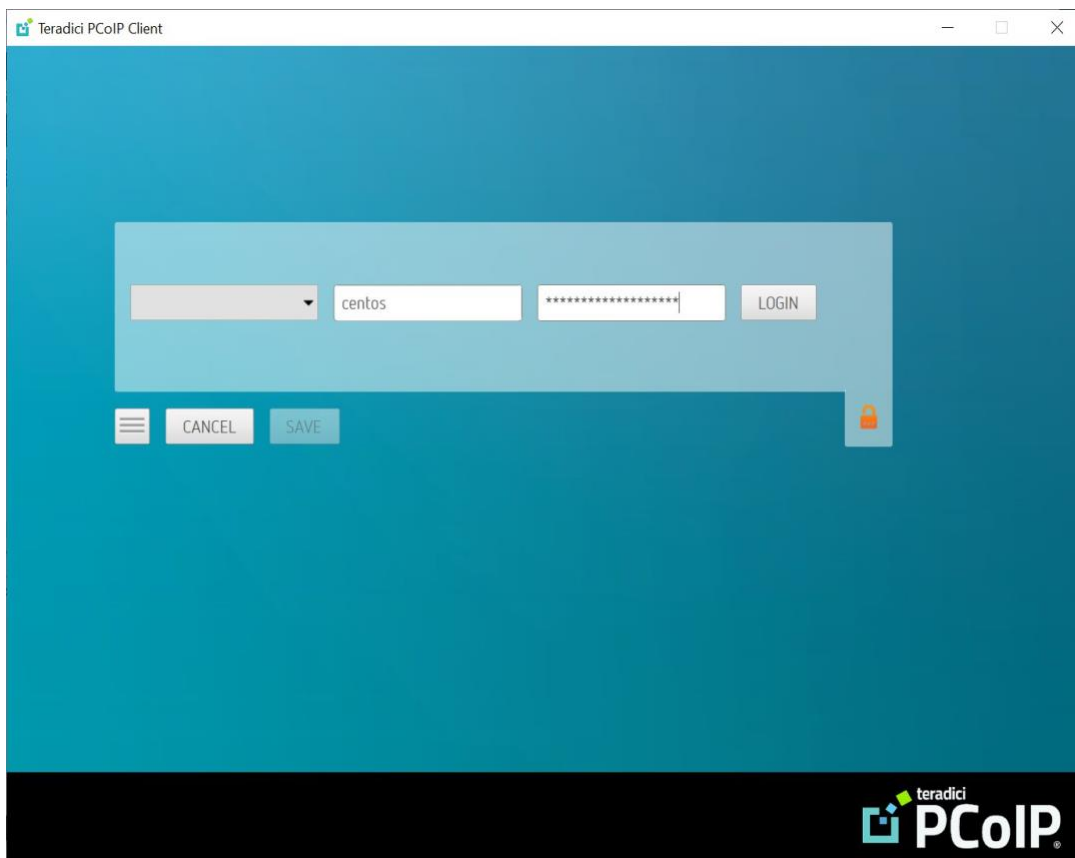


- In the **Host Address or Code** field, enter the **IPv4 Public IP** of your workstation instance.
  - *If you entered the IPv4 Public IP on your Important Information Cheat Sheet, you can cut and paste it from there.*
- In **Connection Name**, enter a name for this connection (e.g., Workstation)
- Click **Next**
- If you get a popup that says “Cannot verify your connection”, click **Connect Insecurely** to continue.
  - Note: This connection is seen as insecure, but because the PCoIP protocol used by Teradici is inherently secure, your connection is still completely safe. From a [Teradici support page](#):

 **Note: About insecure connections**

Windows sees this connection as insecure because the PCoIP Agent uses a self-signed certificate and not one signed by a trusted certificate authority. **The PCoIP session is inherently secure, so connecting this way is safe.** If you would like to avoid this step in the future, you can create your own certificate and install the appropriate files on the remote machine and your PCoIP clients.

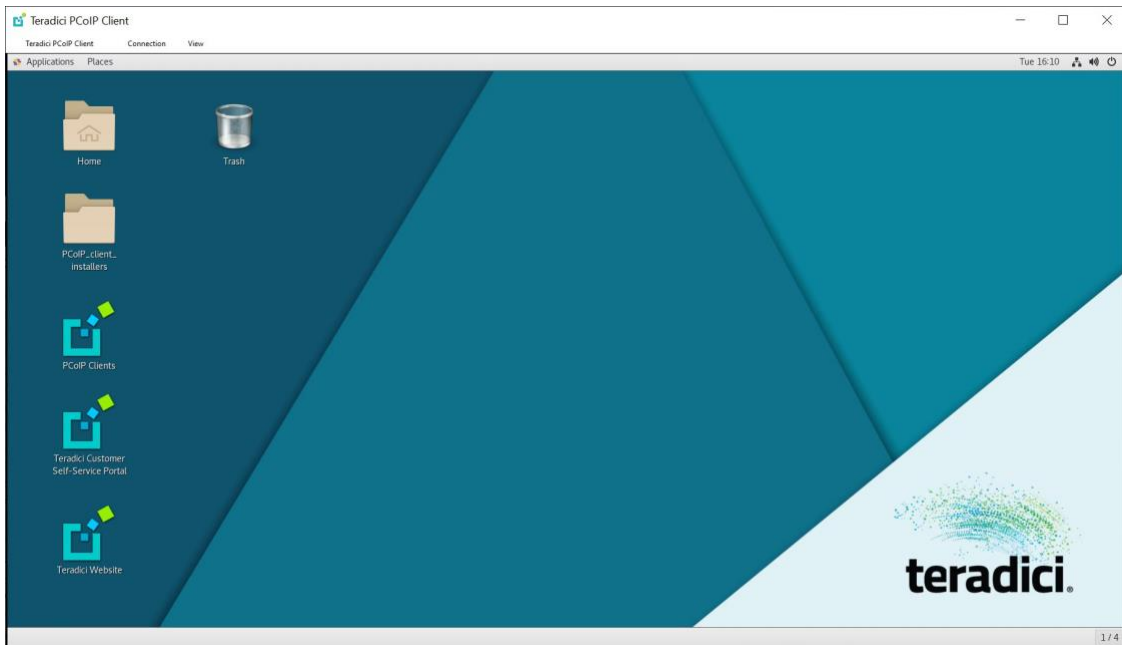
- For links to more information about Teradici security, please see the Appendix.
- For **Username** enter **centos**
  - *You may also want to note this username on the Important Information cheat sheet under “Workstation Username”.*
- For **Password**, use the **Instance ID** for your workstation
  - *You can cut and paste your workstation’s Instance ID from the cheat sheet.*



- Click **Login** to connect to your instance

## Your first EC2 instance

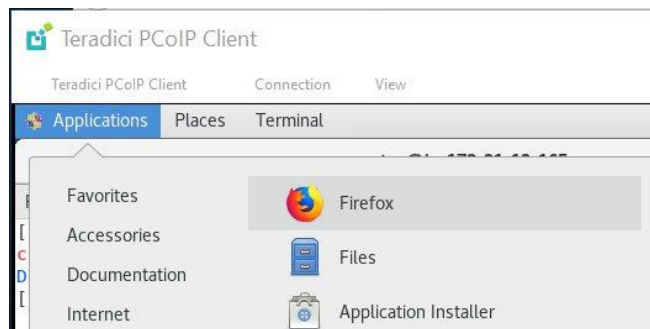
Congratulations! You have launched your first AWS EC2 instance and connected to it with **Teradici**! Feel free to click around and explore your instance. As you can see, there's not much there...but never fear, we're about to add some software.



## Install Blender

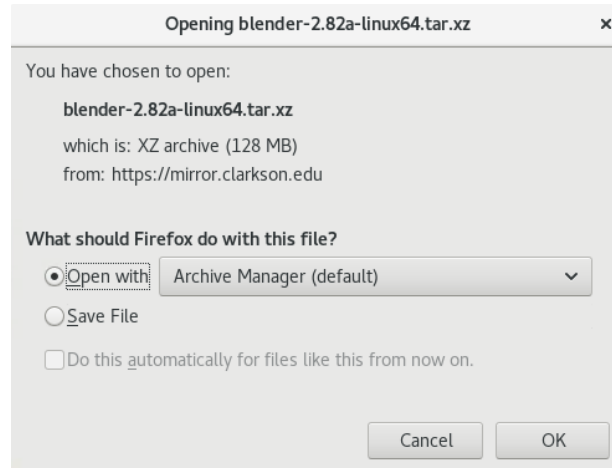
### Download Blender

- Launch Firefox by using the **Applications** menu at the top left of the desktop. You can find Firefox under Favorites.



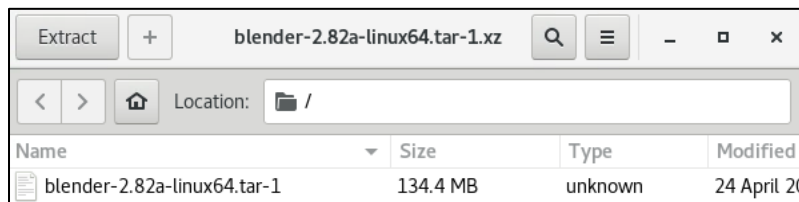
- In Firefox, navigate to <https://www.blender.org/download/>

- Click **Download Blender 2.82a** (the latest version at the time of publication). The version number you see may be different.
- When the window pops up asking what you would like to do with **blender-2.82a-linux64.tar.xz**, leave **Open with Archive Manager (default)** selected and click **OK**.

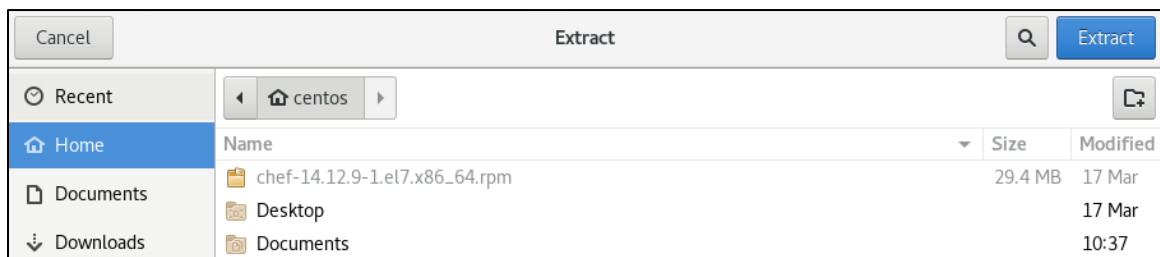


## Install Blender

- When the download is complete, the **Archive Manager** window will appear. In the upper left corner of the window, click the **Extract** button.



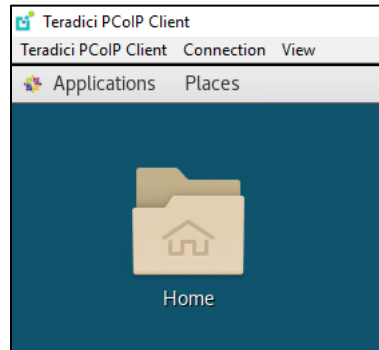
- Next in the **Extract** window, click the **Extract** button in the upper right corner of the window to begin extracting the Blender archive file.



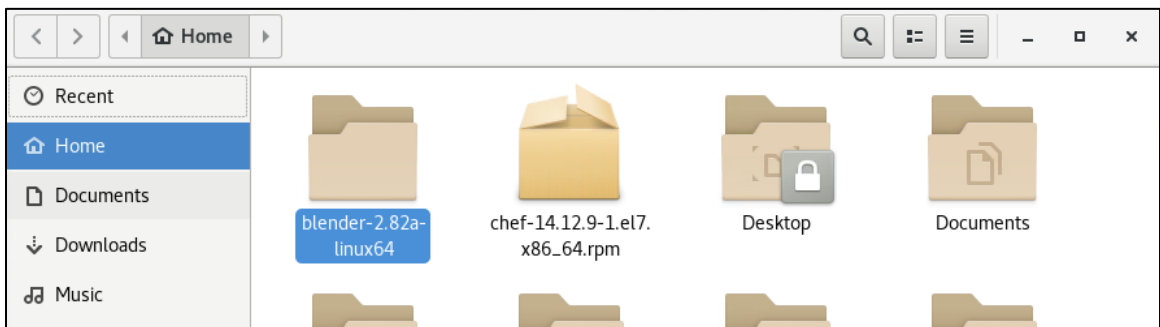
- Finally, click the **Close** button to exit the Archive Manager.

## Launch Blender

- Open the **Home** folder by double-clicking on its icon in the upper left corner of the desktop



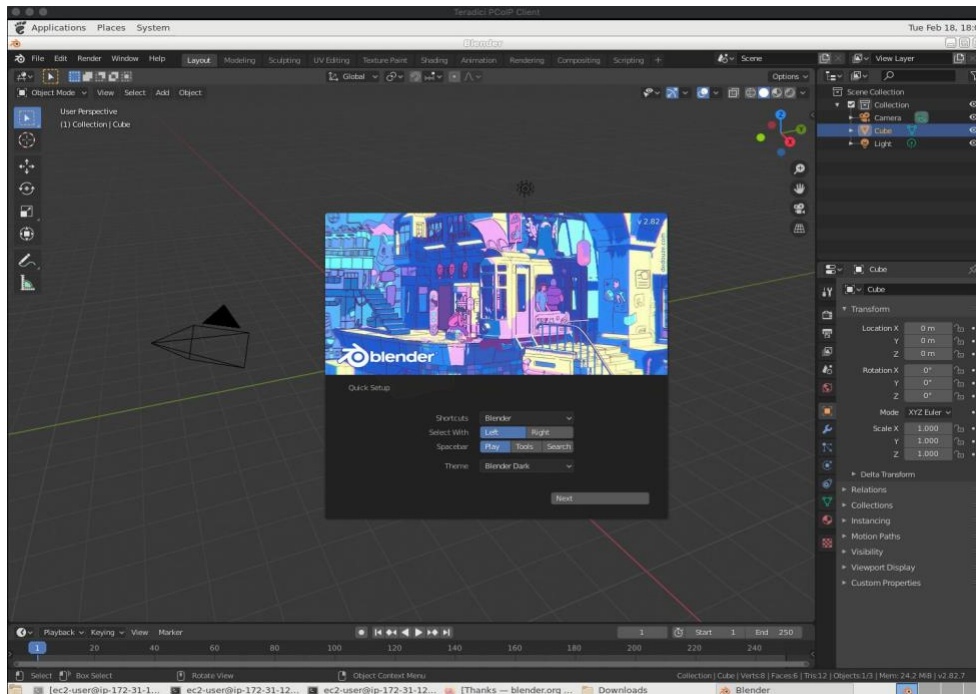
- From the file browser window, open the **blender-2.82a-linux64** folder (your version may be different)



- Then double-click on the **Blender executable**:







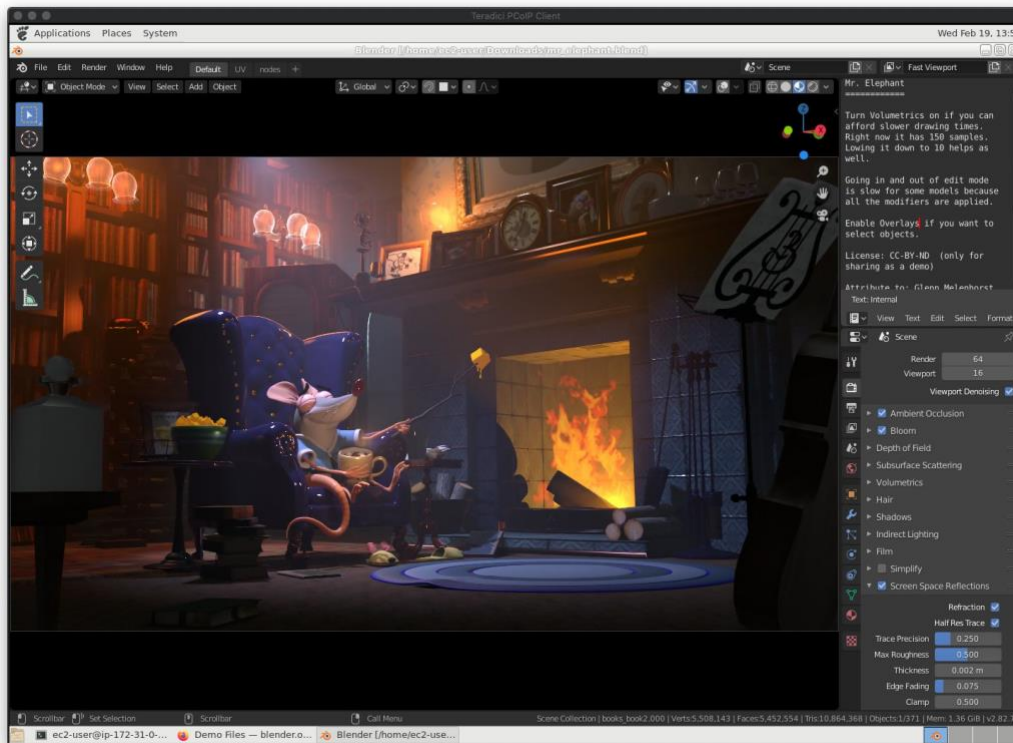
- Next, you will see the Blender window appear:

## Get Example Files

To explore Blender and see how well the instance is working, it's helpful to use real-world data. Wonderfully, Blender offers a number of free files to download and explore.

- On your instance, navigate to the **demo files** site:  
<https://www.blender.org/download/demo-files/>
- Scroll down until you find the **EEVEE** section.
- Choose one of the files to download.
- Once it's downloaded, jump back to Blender and choose **File** → **Open**.
- Select your downloaded file.

- Once the scene is open, use the **middle mouse button** to tumble your view and explore Blender from your streaming instance!



## Cost Optimization

In an effort to keep your costs down as much as possible, we recommend stopping or terminating this instance when you're not using it.

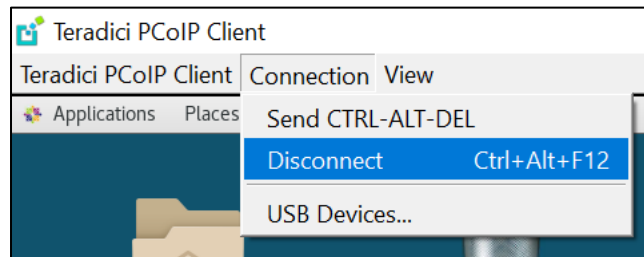
**Stopping** an instance is similar to shutting down your home or work computer. All running applications quit, but whatever data was stored on persistent storage (i.e., the instance's hard drive) is preserved. Once the instance is stopped, you are no longer charged any hourly fees for its use (although you are charged a small amount to store the data that was on the instance's hard drive). A stopped instance will still be listed in the EC2 instance list in the AWS Console and can be restarted at any time.

**Terminating** the instance is more final than stopping it. When an instance is terminated, it is stopped like above, but all the data that was stored in persistent storage is also deleted, so not only do you not incur any hourly charges, you also do not incur any charges for storage of data. You can think of it as first shutting down your computer, but instead of keeping it around for later, you send it to be recycled. Terminated instances

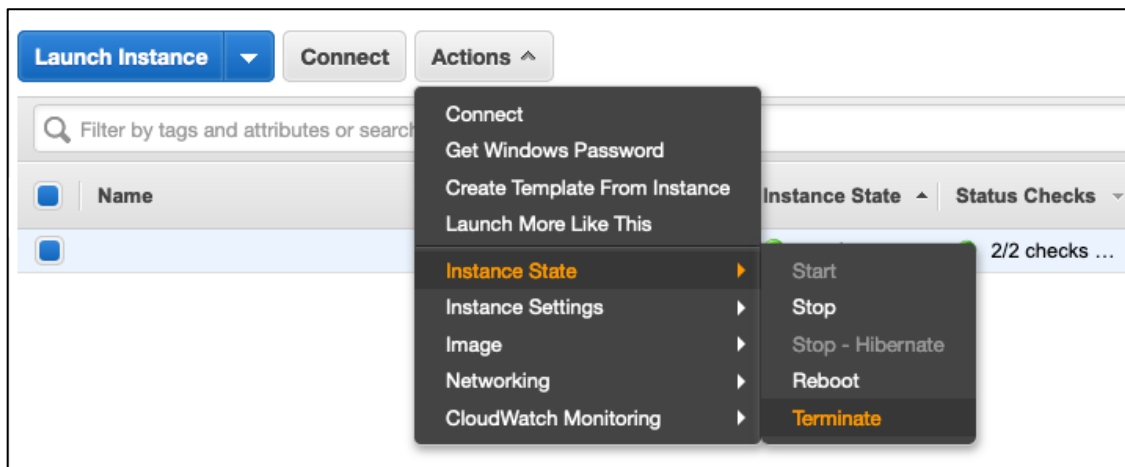
will remain visible in the console for a little while, but unlike stopped instances, they cannot be restarted.

### Terminate your instance

- From the **Teradici Client menu bar** on your instance, click the **Connection** menu, then select **Disconnect**. The Teradici app will close.



- Next, back in the **AWS Console**, select your instance.
- Then click the **Actions** button at the top of the list of instances and choose **Instance State**→**Terminate**. Then click the **Yes, Terminate** button.



The steps for stopping your instance are the same as above, except instead of **Instance State**→**Terminate**, you select **Instance State**→**Stop**. And if you ever want to restart a stopped instance, you can select **Instance State**→**Start**.

## Next Steps

We hope you've enjoyed learning how to launch your own Virtual Workstation on AWS. This tutorial covers just one example of how you can leverage the power of AWS to meet your content creation needs. To access more tutorials that cover additional topics related to cloud solutions for content creation, please visit [AWSThinkbox.com](http://AWSThinkbox.com).

---

## Appendix

### Links to AWS documentation

- [AWS Services By Region](#)
- [AWS Instance Types \(all Regions but Beijing & Ningxia\)](#)
- [AWS Instance Types \(Beijing & Ningxia only\)](#)
- [Consolidated Billing Process](#)
- [AWS Pricing Info](#)
- [AWS Regions and Availability Zones](#)
- [What is Service Quotas?](#)
- [Launching an Instance Using the Launch Instance Wizard](#)
- [Stop and Start Your Instance](#)
- [Terminate Your Instance](#)

### Links to other resources

- [ping.psa.fun](http://ping.psa.fun) - Useful website for estimating latency to different AWS Regions
- [Teradici Cloud Access Software](#)
- [Getting Started Guide - Connecting with a PCoIP Client](#)
- [What is PCoIP Technology?](#)
- [Cloud Access Software Security Features](#)
- [PCoIP Software Client Security Modes](#)
- [Installing Certificates on PCoIP Client for Windows](#)
- [Teradici Graphics Agent Configuration Guide](#)

### Downloads

- [Important Information Cheat Sheet](#)

## Required services for creating a virtual workstation

AWS Service	Availability (as of 11/26/19)	Purpose	Usage for Tutorials
EC2 On Demand Instances	all regions	virtual computers in the cloud	<i>see instance type table below</i>
VPC	all regions	virtual network that contains your virtual workstation	<i>no usage charges for tutorial</i>
IAM	all regions	grant/restrict access to AWS services	<i>no usage charges for tutorial</i>

## Required instance Type FOR THIS TUTORIAL

AWS Workstation Type	Availability (as of 11/26/19)	Purpose	Usage for Tutorials
g4dn.4xlarge - Linux	limited regions	GPU-enabled Linux workstation	about 2 hours

## Tracking availability in your region

You can track the availability of the required services and instance types in your Region using the websites below:

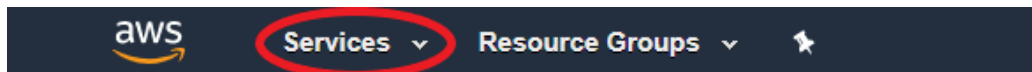
[AWS Services By Region](#)

[AWS Instance Types By Region \(all regions but Beijing & Ningxia\)](#)

[AWS Instance Types By Region \(Beijing & Ningxia only\)](#)

## Creating an IAM user

- Once logged in as the root user, go to the top left of the webpage click the **Services** drop down menu, then under **Security, Identity, & Compliance**, choose **IAM** or simply search for “IAM” in the search field



- In the navigation pane on the left, select **Users**.
- Next click the **Add user** button.
- Enter a **User name** for your new IAM user (e.g., bob).

- Check the check boxes next to both **Programmatic access** and **AWS Management Console access**.
- Select **Custom password** and enter a password in the field.
- Uncheck the check box next to **Require password reset**.
- Click the **Next: Permissions** button.
  - Click on **Attach existing policies directly**.
  - Click the **Create policy** button. This will open a new browser tab.
  - Next click on the **JSON** tab.
  - <shift>+click on the image below to open a new browser tab with the text that needs to be entered into the JSON entry field:

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "ec2:*",
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "elasticloadbalancing:*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2messages:AcknowledgeMessage",
        "ec2messages:DeleteMessage",
        "ec2messages:FailMessage",
        "ec2messages:GetEndpoint",
        "ec2messages:GetMessages",
        "ec2messages:SendReply"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "iam:AddRoleToInstanceProfile"
      ]
    }
  ]
}

```

**streaming\_workstation\_iam\_policy.json** - <shift>+click on the image above to open the JSON file in a new tab

- Replace the existing text in the JSON entry field with the text from the file above. Note: The image above doesn't show the complete file, please make sure you cut and paste the whole thing.
- When you're all done, the JSON entry field should look like this:

Create policy 1 2

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor JSON Import managed policy

```

80     "aws-marketplace:ViewSubscriptions",
81     "aws-marketplace:Subscribe",
82     "aws-marketplace:Unsubscribe"
83   ],
84   "Effect": "Allow",
85   "Resource": "*"
86 },
87 {
88   "Action": [
89     "aws-marketplace:CreatePrivateMarketplaceRequests",
90     "aws-marketplace:ListPrivateMarketplaceRequests",
91     "aws-marketplace:DescribePrivateMarketplaceRequests"
92   ],
93   "Effect": "Allow",
94   "Resource": "*"
95 }
96 ]
97 }

```

Character count: 1,415 of 6,144 Cancel **Review policy**

- Click the **Review policy** button.
- For **Name** enter **Streaming-Workstation-Policy**.
- For **Description** enter **Custom IAM policy with all permissions to complete the Streaming Workstation tutorial**.
- Click the **Create policy** button.
- Return to the browser tab that you were using to create your user.
- In the policies search field, enter **Streaming-Workstation-Policy** and then click the checkbox next to it to select it.
  - If you don't see the policy, click the **refresh** button.

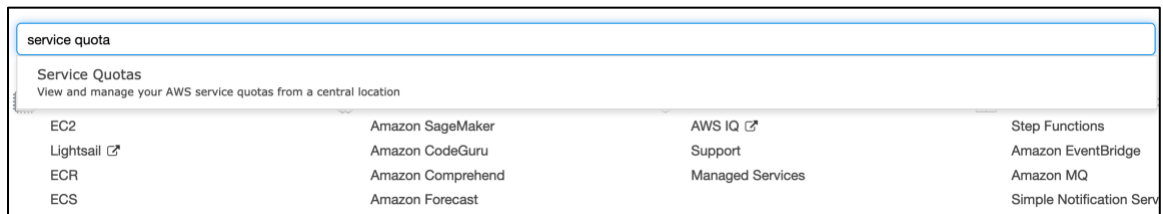


- Click the **Next: Tags** button.
- Click the **Next: Review** button.
- Review the information for the user and then click the **Create User** button.
- On the next page, you'll be presented with the security credentials for your user. You'll need these credentials later in the tutorials, so it is important that you download them now by clicking the **Download .csv** button.
- Make note of the location of the .csv file so that you can find it later. *Also enter the location of the .csv file on the [Important Information Cheat Sheet](#) that we've provided.*

- When you're done, login as your new user.

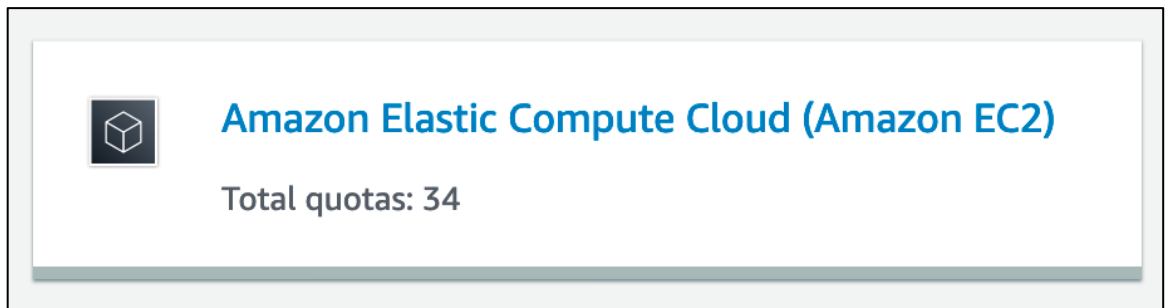
### Checking Your G Instance Quota

- If you haven't already, check that your Region is set correctly. For instructions, please see the [Set Your Region](#) section above.
- Click the **Services** drop down menu next to the AWS logo at the top left corner of

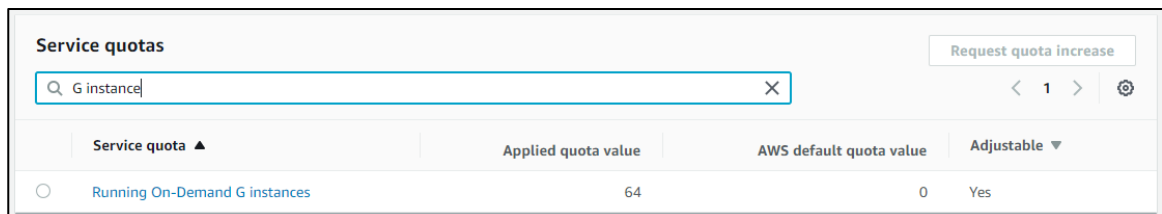


the AWS Console and type **service quota** in the search field.

- Select **Service Quotas** to go to the Service Quotas Dashboard.
- Click the **Amazon Elastic Cloud Compute (Amazon EC2)** card.



- In the "Filter by..." field, type in **G instance**
- Now you can review the quota you have available for your G instances. In order to complete these tutorials, you will need an **Applied quota value** of at least **16** because the **g4dn.4xlarge** instance we will be using in part of the tutorial uses **16**



### vCPUs.

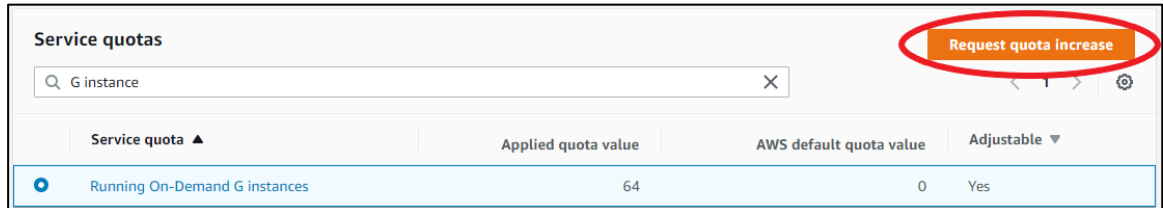
- To see how many vCPUs are used by other G4 instance types, please see the [G4 Product Details](#) chart.



- To request an increase in your quota value, please follow the instructions below.

### Requesting a Quota Increase

- Click the **circle** to the left of **Running On-Demand G instances**.
- Then click the **Request quota increase** button.



- Enter the new **Quota value** you would like to receive. Note: Make sure to enter a value of at least **16**.
- Click **Request**.
- Within a few minutes you will receive a new **Support Case** for your request. It can take anywhere from 12-48 hours to return a limit increase request.
- You can track the status of your request in the Service Quotas Dashboard, by clicking on **Quota request history** in the navigation panel on the left.
- You will see the status of your request in the list on the right.
- To see more details, click on the **status** value. (Immediately after submitting the request, it will say "Quota requested"). Then in the window that pops up, click on the **Support Center case number**
- If you have a contact at AWS who supports your account, you should also notify them that you have submitted a quota increase request.
- Refer this [AWS Documentation](#) to learn more about service quotas.