

Amazon Web Services

Web Development
Foundations

Amazon Web Services

Amazon offers a variety of services related to cloud hosting, database management, content delivery, and more.

Visit [their website](#) for details on their various offerings, and [how to get started](#).

Thankfully for us, at the time of this writing, they have some very basic options that are free that we can use for practice.

The screenshot shows the official AWS website homepage. At the top, there's a navigation bar with links for Products, Solutions, Pricing, Documentation, Learn, Partner Network, AWS Marketplace, Customer Enablement, Events, Explore More, Contact Us, Support, English, My Account, and a prominent orange 'Create an AWS Account' button. Below the navigation is a dark banner with the heading 'Start Building on AWS Today' and a subtext about building sophisticated applications with increased flexibility, scalability, and reliability. A 'Get Started for Free' button is visible. The main content area features several cards: 'Start Building With Free Tier' (using Amazon EC2, S3, and more—free for a full year), 'Launch Your First App in Minutes' (learning AWS fundamentals and starting building with short step-by-step tutorials), 'Enable Remote Work & Learning' (supporting remote employees, students, and contact center agents), and 'Amazon Lightail' (everything you need to get started on AWS—for a low, predictable price). Below this is a section titled 'Explore Our Solutions' with a 'Industry' and 'Technology Category' filter. It displays four cards: 'Advertising & Marketing' (achieving cost-efficiency for petabyte-scale analytics and single-digit millisecond latency workloads), 'Financial Services' (reducing costs, increasing resiliency, exploring AWS solutions across banking, payments, capital markets, and insurance), 'Game Tech' (creating computationally ridiculous games across all genres and platforms), and 'Healthcare & Life Sciences' (from benchtop to bedside, innovating faster to improve patient outcomes and lower costs).

Getting Started

Create an Account

Firstly, if you don't already have one, you'll need to make an account.

Click "Get Started for Free" on the homepage and sign up.

Please note that you **will** require a credit card, VISA debit card, a VISA/MasterCard gift card, or an equivalent during the sign-up process.



Sign up for AWS

Email address
You will use this email address to sign in to your new AWS account.

Password

Confirm password

AWS account name
Choose a name for your account. You can change this name in your account settings after you sign up.

Continue (step 1 of 5)

[Sign in to an existing AWS account](#)



Congratulations!

Thank you for signing up with AWS.

We are activating your account, which should take a few minutes. You will receive an email when this is complete.

[Go to the AWS Management Console](#)

[Sign up for another account or Contact Sales](#)



Sign up for AWS

Select a support plan

Choose a support plan for your business or personal account. [Compare plans and pricing examples](#)

Basic support - Free
Recommended for new users just getting started with AWS

- 24x7 self-service access to AWS resources
- For account and billing only
- Access to Personal Health Dashboard & Trusted Advisor

Developer support - From \$29/month
Recommended for developers experimenting with AWS

- Email access to AWS Support during business hours
- 12 (Business)-hour response times

Business support - From \$100/month
Recommended for running production workloads on AWS

- 24x7 tech support via email, phone, and chat
- 4-hour response times
- Full set of Trusted Advisor best-practice recommendations

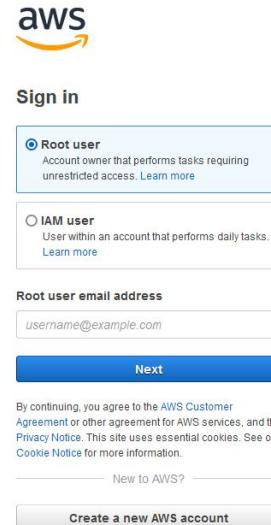
[Need Enterprise level support?](#)
From \$15,000 a month you will receive 15-minute response times and concierge-style experience with an assigned Technical Account Manager. [Learn more](#)

Complete sign up

Sign In

[Sign in to your account](#) and access the [console](#).

We'll often need "root" (administrator) access in order to make changes to the base configuration and install new features in our installation. Note, however, that you only want to use this sort of access when you absolutely need to, as the commands you have available can cause permanent damage to your project if you're not careful. It is also recommended you never share root user access or accounts with anyone for this reason.



The image shows the AWS sign-in interface. At the top is the AWS logo. Below it is a "Sign in" section. Under "Sign in", there are two radio button options: "Root user" (selected) and "IAM user". The "Root user" option is described as "Account owner that performs tasks requiring unrestricted access." Below this is a "Root user email address" input field containing "username@example.com". A large blue "Next" button is centered below the input fields. At the bottom of the form, a small note states: "By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information." There are also "New to AWS?" and "Create a new AWS account" links at the bottom.



The AWS Management Console

From the AWS Management Console, we can spin up virtual machines, databases, and more. Make a note of the “All services” dropdown near the top of the page, as well as the guides and options available beneath.

The screenshot shows the AWS Management Console homepage. At the top, there's a search bar and a "Services" dropdown. Below the search bar, the title "AWS Management Console" is displayed. A sidebar on the left lists "AWS services" and "Build a solution". The main content area features several service guides with icons and estimated times:

- Launch a virtual machine** (With EC2) - 2-5 minutes
- Build a web app** (With Elastic Beanstalk) - 6 minutes
- Build using virtual servers** (With Lightsail) - 1-2 minutes
- Register a domain** (With Route 53) - 3 minutes
- Connect an IoT device** (With AWS IoT) - 3 minutes
- Start migrating to AWS** (With Amazon MGN) - 1-2 hours
- Start a development project** (With CodeStar) - 3 minutes
- Deploy a serverless microservice** (With Lambda, API Gateway) - 2 minutes

Below these, there are sections for "Getting Started with AWS", "Free AWS Training", and "Have feedback?". The "Free AWS Training" section includes links to "AWS Certified Practitioner Essentials" and "AWS Cloud Practitioner Essentials". The "Have feedback?" section has a feedback form.

Choosing a Hosting Instance

Preparing a Hosting Solution

Let's create a [free tier solution](#).

AWS Free Tier

Filter by:

- Free Tiers
- Upcoming
- Amazon Free Tier
- Cloud

Product Categories:

- Analytics
- Application integration
- Compute
- Container management
- Databases
- Dev/Test
- Edge Compute
- Front End Web & Mobile
- Games
- Internet of Things
- Machine Learning
- Management & Governance
- Media Services
- Mobile & Device
- Networking & Content Delivery
- Storage
- Security, Identity, & Compliance
- Services
- Virtualization

Free Tier details

Service	Free Tier	Allocation	Allocation Type
Amazon DynamoDB 25 GB	Fast and flexible NoSQL database with seamless scalability.	Amazon Free Tier	Amazon Free Tier
Amazon CloudWatch Metrics 10	Monitoring for AWS cloud resources and applications.	Amazon Free Tier	Amazon Free Tier
Amazon Glacier (Glacier API only) 10 GB	Long-term, secure, durable object storage.	Amazon Free Tier	Amazon Free Tier
Amazon Macie 1 GB	Discover, classify, and protect your data.	Amazon Free Tier	Amazon Free Tier
Amazon SES 62,000	Cost effective email service in the Cloud.	Amazon Free Tier	Amazon Free Tier

13 Month Free: These free offers are only available to new AWS customers, and are available for 13 months following your AWS sign-up date. When you 13 month free usage term ends, or if you've already used up your 13 month free usage term, you'll be charged at the standard rate for the AWS services you're using.

About free: These free offers do not provide credits or a credit of 13 months for the time that you've already used up your 13 month free usage term.

AWS Products Solutions Pricing Documentation Learn Partner Network AWS Marketplace Customer Enablement Events Explore More

Web Hosting

Amazon Web Services offers cloud web hosting solutions that provide businesses, non-profits, and governmental organizations with low-cost ways to deliver their websites and web applications. Whether you're looking for a marketing, rich-media, or ecommerce website, AWS offers a wide-range of website hosting options, and we'll help you select the one that is right for you.

Why use AWS for web hosting?

Broad platform support

Your customers can be anywhere in the world. With AWS you can have a datacenter or CDN hosting your website in any geography you choose with just a few mouse clicks.

Datacenters worldwide

Website traffic can fluctuate a lot. From quiet times in the middle of the night, to campaign drives, social media sharing, or traffic spikes, AWS infrastructure that can grow and shrink to meet your needs.

Scalable from day one

AWS only charges you for the resources you use, with no up-front costs or long-term contracts. AWS has web hosting options that offer pay-as-you-go pricing or fixed monthly pricing.

Flexible pricing models

“Always Free” Services

There are a variety of options available that are free for different periods of time and for trial periods.

There is a filter available for “always free” selections, which will help you narrow them down to exclusively completely free offerings.

You may also need to make use of “12 months free” options if no “always free” options for the target service are available.

The screenshot shows the AWS Free Tier landing page. At the top, there's a navigation bar with links for Products, Solutions, Pricing, Documentation, Learn, Partner Network, AWS Marketplace, Customer Enablement, Events, Explore More, and a search bar. Below the navigation is a "Free Tier details" section with a sidebar for filtering products by tier type, including "Always Free" (which is selected and highlighted with a red box). The main content area displays several free services in a grid:

- DATABASE**: Amazon DynamoDB (25 GB) - Described as a fast and flexible NoSQL database with seamless scalability.
- COMPUTE**: AWS Lambda (1 Million invocations per month) - Described as a compute service that runs your code in response to events and automatically manages the compute resources.
- MOBILE**: Amazon SNS (1 Million publications) - Described as a fast, flexible, fully managed push messaging service.
- DEVELOPER TOOLS**: Amazon CloudWatch (10 custom metrics and alarms) - Described as monitoring for AWS cloud resources and applications.
- BUSINESS PRODUCTIVITY**: Amazon Chime (Basic) - Described as a modern unified communications service that offers frustration-free meetings with video, audio, and screen sharing.
- MOBILE**: Amazon Cognito (50,000 users/month) - Described as mobile user identity and synchronization.

Each service entry includes a brief description, its tier (Free Tier or Always Free), and usage limits or details.

All-Purpose Hosting

It is important to explore the available options, their strengths, their weaknesses, and their pricing. Sometimes different configurations may serve your project better, worse, or not at all.

A very flexible, and familiar for those used to more traditional hosting platforms, solution is the [Amazon EC2](#) compute option. Amazon EC2 stands for “Amazon Elastic Compute Cloud.” That the server resources can adjust based on the load placed on it by user requests, is where it gets its elastic and cloud reputation.

This particular type of hosting will allow you to set up a dedicated virtual machine capable of serving website files, running server-side scripts, and running a database management system.

They offer the following article to help you get started: “[Tutorial: Get started with Amazon EC2 Linux instances](#).”

Select EC2

Sort by “12 Months Free” and “Always Free,” or use their search feature, to locate the Amazon EC2 offering. On the page that follows, click the “Get started” button.

The screenshot shows the AWS EC2 landing page. The main heading is "Amazon EC2" with the subtext "Secure and resizable compute capacity to support virtually any workload". Below this is a large image of a futuristic city skyline at night. A prominent orange "Get started with Amazon EC2" button is at the bottom left. The page footer contains sections for "Nearly 400 instances for virtually every business need", "Only cloud provider that supports macOS", "25 regions and 80 availability zones globally", and "Choice of Intel, AMD, and Arm-based processors".

The screenshot shows the AWS Free Tier page under the "Terms and Conditions" tab. The "COMPUTE" section is highlighted with a red box. It lists two services: "Amazon EC2" (Free Tier, 12 MONTHS FREE, 750 Hours per month) and "Amazon DynamoDB" (Free Tier, ALWAYS FREE, 25 GB of storage). Other services listed include Lambda, Lambda@Edge, and Step Functions. The "STORAGE" and "DATABASE" sections are also visible on the right.

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'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 offers the broadest and deepest compute platform with choice of processor, storage, networking, operating system, and purchase model. We offer the fastest processors in the cloud and we are the only cloud with 400 Gbps ethernet networking. We have the most powerful GPU instances for machine learning training and graphics workloads, as well as the lowest cost-per-inference instances in the cloud. More SAP, HPC, Machine Learning, and Windows workloads run on AWS than any other cloud. Click [here](#) to learn What's New with Amazon EC2.

Nearly 400 instances for virtually every business need

Only cloud provider that supports macOS

25 regions and 80 availability zones globally

Choice of Intel, AMD, and Arm-based processors

Reliable, scalable infrastructure on demand

- Increase or decrease capacity within minutes, not hours or days
- SLA commitment of 99.99% availability for each Amazon EC2 region. Each region consists of at least 3 availability zones
- The AWS Region/AZ model has been recognized by Gartner as the #1 cloud provider for regional resilience and footprint size



*Since March, we have seen market volumes increase by 2.3x...We

Setting up an EC2 Instance

EC2 Dashboard and Console

Welcome to the [EC2 Dashboard and Console](#)! These screens and options allow us to create, customize, and monitor any virtual machines we will work on.

In the top-right corner of the screen, there is a drop down beside your name. It is a good idea to set your region to an area close to you, if it has not already defaulted to this. You'll see faster load times if your server is closer to you.



The screenshot shows the EC2 Dashboard and the new EC2 console side-by-side. The dashboard on the left has a sidebar with region selection, while the new console on the right has a sidebar with account attributes and explore options. Both sides show resource counts and a 'Launch instance' button.

Launching an Instance

Click the “Launch Instance” button, and we can begin setting up our instance.



For this example, we'll go with the default “Amazon Linux 2 AMI (HVM), SSD Volume Type.”

The screenshot shows the AWS Management Console interface for launching an EC2 instance. The top navigation bar includes 'Services', a search bar, and tabs for 'AMI', 'Configure Instance', 'Add Storage', 'Add Tags', 'Configure Security Group', and 'Review'. Below the tabs, a note states: 'An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. You can select an AMI provided by AWS, our user community, or the AWS Marketplace; or you can select one of your own AMIs.' A search bar allows users to search for AMIs by name. The main content area is titled 'Quick Start' and lists several AMI options:

- Amazon Linux 2 AMI (HVM), SSD Volume Type** - ami-07d95749ec4b20a3 (64-bit x86) / ami-0c3eda36031f1578 (64-bit Arm)
Description: 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, Brutus 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is approaching end of life on December 31, 2020 and was removed from this wizard.
Virtualization type: HVM
EBS Optimized: Yes
Select button (highlighted with a red box)
- Red Hat Enterprise Linux 8 (HVM), SSD Volume Type** - ami-0277bce2afabaa3ad (64-bit x86) / ami-0ee0ff629139ee1481 (64-bit Arm)
Description: Red Hat Enterprise Linux version 8 (HVM), EBS General Purpose (SSD) Volume Type
Virtualization type: HVM
EBS Optimized: Yes
Select button
- SUSE Linux Enterprise Server 15 SP2 (HVM), SSD Volume Type** - ami-0d8c979549f51c0 (64-bit x86) / ami-07a449590fa1e3715 (64-bit Arm)
Description: SUSE Linux Enterprise Server 15 Service Pack 2 (HVM), EBS General Purpose (SSD) Volume Type. Amazon EC2 AMI Tools preinstalled. Apache 2.2, MySQL 5.6, PHP 5.3, and Ruby 1.8.7 available.
Virtualization type: HVM
EBS Optimized: Yes
Select button
- Ubuntu Server 20.04 LTS (HVM), SSD Volume Type** - ami-0801628222c2e96d5 (64-bit x86) / ami-0994658be3d2178e0 (64-bit Arm)
Description: Ubuntu Server 20.04 LTS (HVM) EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
Virtualization type: HVM
EBS Optimized: Yes
Select button
- Microsoft Windows Server 2019 Base** - ami-05ffadfe032c3d88
Description: Microsoft Windows 2019 Datacenter edition [English]
Virtualization type: HVM
EBS Optimized: Yes
Select button
- Amazon RDS**
Description: Are you launching a database instance? Try Amazon RDS. Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale your database on AWS by automating time-consuming database management tasks. With RDS, you can easily deploy Amazon Aurora, MariaDB, MySQL, Oracle, PostgreSQL, and SQL Server databases on AWS. Aurora is a MySQL- and PostgreSQL-compatible, enterprise-class database at 1/10th the cost of commercial databases. Learn more about RDS
Launch a database using RDS button

Choose an Instance Type

As we're hoping to experiment in a free sandbox, we'll select the t2.micro type. This particular option is eligible for Amazon's web tier.

When using any service, be careful to review what you may or may not be charged for before agreeing to anything.

Once selected, click "Review and Launch" to continue.

The screenshot shows the AWS EC2 console interface. The top navigation bar includes 'Services', a search bar, and tabs for '1. Choose AMI', '2. Choose Instance Type' (which is highlighted in blue), '3. Configure Instance', '4. Add Storage', '5. Add Tags', '6. Configure Security Group', and '7. Review'. Below the tabs, there are filters for 'All instance families' (selected), 'Current generation' (selected), and 'Show/Hide Columns'. A message at the top states: '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.' A table lists various instance types, including t2.nano, t2.micro (selected and highlighted with a red box), t2.small, t2.medium, t2.large, t2.xlarge, t3.nano, t3.micro, t3.small, t3.medium, t3.large, t3.xlarge, t3a.nano, and t3a.micro. The t2.micro row has a tooltip: 'Micro instances are eligible for the AWS Free Usage Tier. For the first 12 months following your AWS sign-up date, you get up to 750 hours of micro instances each month. When your free usage tier expires or if your usage exceeds the free tier restrictions, you pay standard, pay-as-you-go service rates. Learn more about free usage tier eligibility and restrictions'.

Review Instance Launch

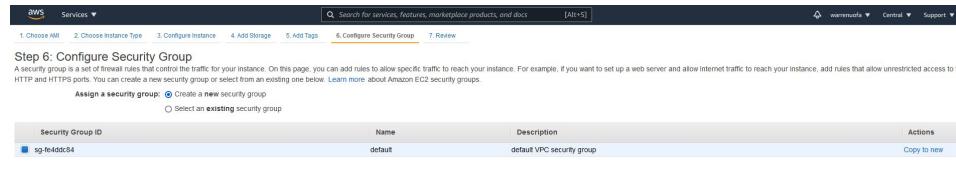
In this step, we'll have a look at editing the security groups.

The screenshot shows the AWS Step 7: Review Instance Launch wizard. At the top, there are tabs: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The 7. Review tab is selected. Below the tabs, the page title is "Step 7: Review Instance Launch". A sub-instruction says "Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process." A warning message in a yellow box states: "⚠ Improve your instances' security. Your security group, launch-wizard-1, is open to the world. Your instances may be accessible from any IP address. We recommend that you update your security group rules to allow access from known IP addresses only. 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". Below this, the "AMI Details" section shows "Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-07d9874fec4b20a3" and indicates it's a "Free tier" AMI. The "Instance Type" section shows a table with one row: t2.micro, 1 ECUs, 1 vCPUs, 1 GiB Memory, EBS only storage, and Low to Moderate network performance. To the right of the table are three buttons: "Edit instance type", "Edit security groups" (which is highlighted with a red box), and "Edit instance details". Below the table are sections for "Security Groups", "Instance Details", "Storage", and "Tags". At the bottom of the page are buttons for "Cancel", "Previous", "Launch" (which is highlighted with a blue box), and "Next". The footer includes links for "Feedback", "English (US)", "Privacy Policy", "Terms of Use", and "Cookie preferences".

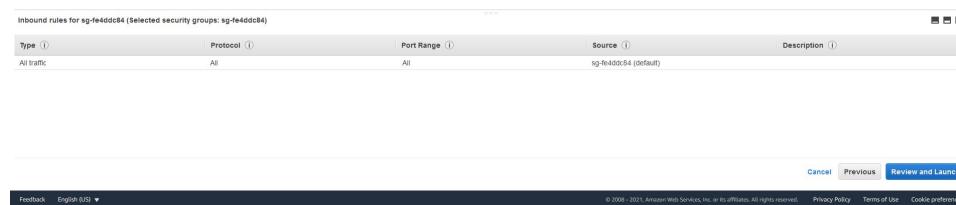
Assigning a Security Group

For detailed information on how security groups work and what your options are in-full, see the article: "[Amazon EC2 security groups for Linux instances.](#)"

Click the radio button that reads “Create a new security group”, and we’ll proceed to the next step.



Security Group ID	Name	Description	Actions
sg-fe4ddc84	default	default VPC security group	Copy to new



Type	Protocol	Port Range	Source	Description
All traffic	All	All	sg-fe4ddc84 (default)	

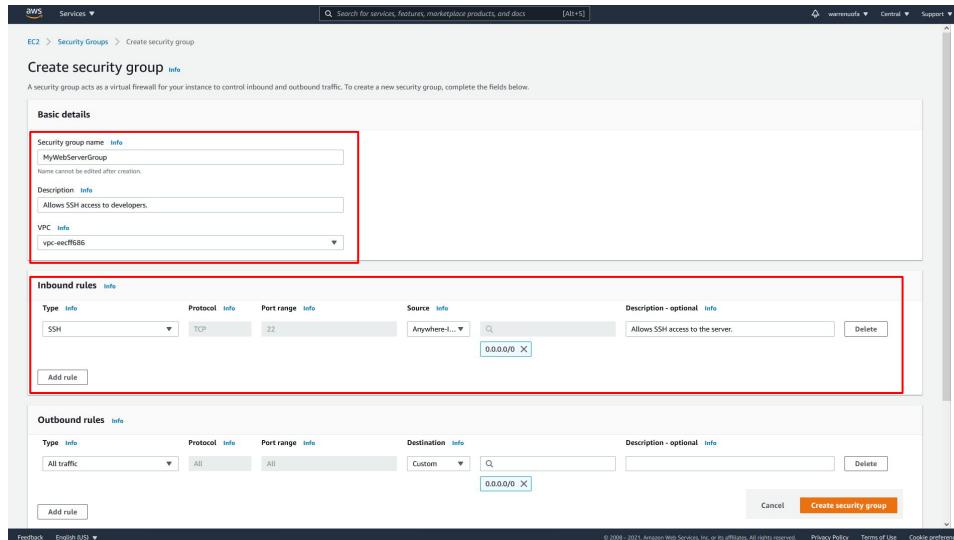
Adding a Security Group

We'll add a very basic Security Group to ensure that our server will allow SSH traffic. SSH is the type of access we'll need in order to log into our virtual machine and run commands.

There will be some more steps to SSH configuration, but we'll get to those soon! For now, name your security group, add a description, and add the following inbound rule:

- Type: SSH
- Source: Anywhere (IPv4)

Once this information is added, click the "Create security group" button to proceed.



Review the Instance

You now have a chance to review any details related to the instance you're creating. Feel free to expand the details for "Instance Details," "Storage," and "Tags" to see how it is currently configured.

Once you're ready to proceed, click the "Launch" button.

The screenshot shows the AWS Step 7: Review Instance Launch page. At the top, there's a navigation bar with links: 1. Choose AMI, 2. Choose Instance Type, 3. Configure Instance, 4. Add Storage, 5. Add Tags, 6. Configure Security Group, and 7. Review. The 'Review' link is underlined. Below the navigation is a search bar and a help link [Alt+G]. On the right side of the header are links for warrenfa, Central, and Support.

Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details
Amazon Linux 2 AMI (HVM), SSD Volume Type - ami-07d95749ec4b20a3
Free tier: 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, Braswell 2.29.1, and the latest software packages through extras. This AMI is the successor of the Amazon Linux AMI that is a...
Edit AMI

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

Security Groups
Edit security groups
Security Group ID: sg-fe40d64 | Name: default | Description: default VPC security group
All selected security groups inbound rules
Type: All traffic | Protocol: All | Port Range: All | Source: sg-fe40d64 (default) | Description: (empty)

Instance Details
Edit instance details
Storage
Tags

At the bottom right are buttons for Cancel, Previous, and Launch. The Launch button is highlighted in blue. The footer includes links for Feedback, English (US), Privacy Policy, Terms of Use, and Cookie preferences, along with copyright information: © 2006–2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

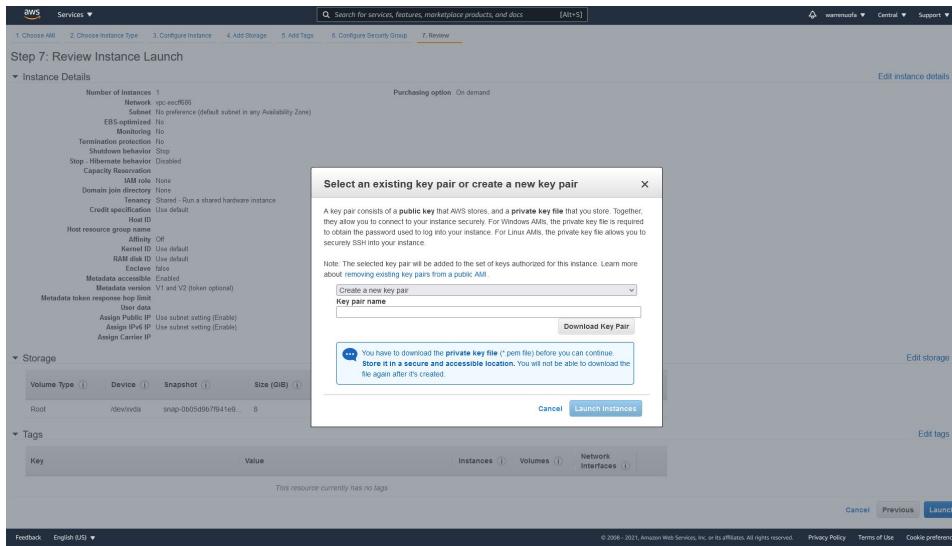
Create a New Key Pair

In order to connect to your instance, you'll need a key pair. Key pairs are used for securing and encrypting connections between devices.

Because we have not generated any pairs yet, you'll need to name your pair and download it.

Ensure that you download your .pem file to a safe place, as you will not be able to access this again. It is a security measure, that you will be the only one possessing a copy of this file.

Once downloaded, you can click the “Launch Instances” button.



Launching!

Now the instance should be launched, or will be launching shortly. Note that there are links on this screen for [billing alerts](#), as well as [guides](#) and [forums](#) to assist you in your journey.

When you're ready to proceed to working with your instance, click the "View Instances" button.

The screenshot shows the AWS Launch Status page. At the top, there's a navigation bar with the AWS logo, a search bar, and links for 'Services', 'Search for services, features, marketplace products, and docs [Alt+S]', 'Warrooms', 'Central', and 'Support'. Below the header, the main content area is titled 'Launch Status'. It displays a green success message: 'Your instances are now launching' with a note that they have been initiated and a link to 'View launch log'. There's also a blue info message about 'Get notified of estimated charges' with a note about creating billing alerts. Underneath, a section titled 'How to connect to your instances' provides tips and links to 'How to connect to your Linux instance', 'Amazon EC2 User Guide', 'Learn about AWS Free Usage Tier', and 'Amazon EC2 Discussion Forum'. Another section titled 'While your instances are launching you can also' lists options like 'Create status check alarms to be notified when these instances fail status checks. (Additional charges may apply)', 'Create and attach additional EBS volumes (Additional charges may apply)', and 'Manage security groups'. At the bottom right, there's a blue 'View Instances' button. The footer contains links for 'Feedback', 'English (US) ▾', '© 2006–2011, Amazon Web Services, Inc. or its affiliates. All rights reserved.', 'Privacy Policy', 'Terms of Use', and 'Cookie preferences'.

An EC2 Instance

Our instance can now be viewed in our console!
Note that the state should be “Running” once it is started.

The screenshot shows the AWS EC2 Instances page. At the top, there is a search bar and several filter options. Below the search bar, a table displays one instance:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP
-	i-0c27e074ef02d32ad	Running	t2.micro	2/2 checks passed	No alarms	ca-central-1b	ec2-15-223-119-196.ca...	15.223.119.196	-

Below the table, there is a detailed view for the selected instance (i-0c27e074ef02d32ad). The details include:

- Instance summary:** Public IPv4 address (15.223.119.196), Private IPv4 address (172.31.14.178), Public IPv6 DNS (ip-172-31-14-178.ca-central-1.compute.internal), Private IPv6 DNS (ip-172-31-14-178.ca-central-1.compute.internal), VPC ID (vpc-eccff666), Subnet ID (subnet-2e443f54).
- Details tab:** Contains sections for Instance details, Platform, Platform details, Launch time, Stop-hibernate behavior, State transition reason, State transition message, AMI ID, AMI name, AMI location, AMI launch index, Credit specification, and RAM disk ID.
- Security tab:** Contains IAM Role information.
- Networking tab:** Contains Elastic IP addresses information.
- Storage tab:** Contains information about the volume attached to the instance.
- Status checks tab:** Contains information about the instance's health.
- Monitoring tab:** Contains monitoring and termination protection settings.
- Tags tab:** Contains any tags assigned to the instance.

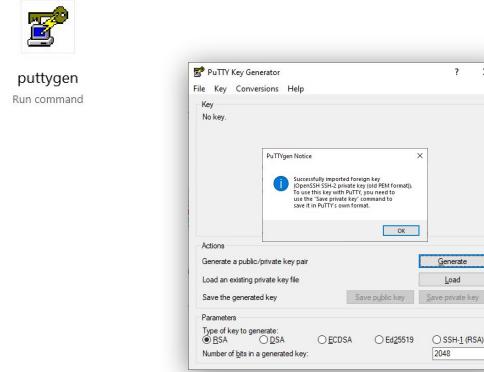
Connecting to your EC2 Instance

Connecting to your Linux Instance

Time to [connect to our new instance](#)! We'll need our key file. If using Windows, please [download PuTTY](#).

We'll need to generate keys in a format that PuTTY can understand, so we'll need to run its generator before we can connect to our server securely. When you install the software, two programs should be installed, please open PuTTYgen.

Click "Load," choose to show all file types, and find the keys file we generated earlier. Once loaded, save a private key file. It is heavily recommended you add a passphrase to increase the security of your key, ensure you will not forget it, or have it written down in a secure location.

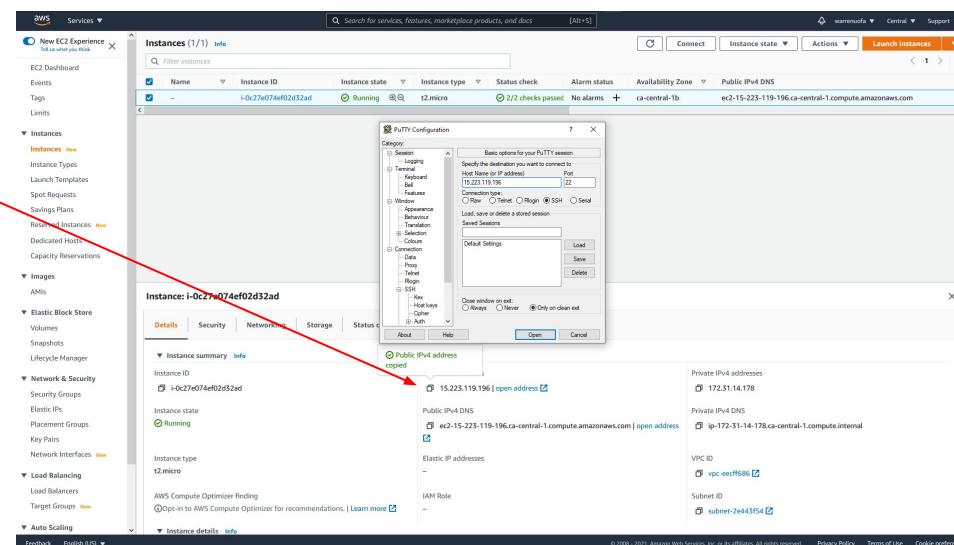


Connecting to your Instance

We'll need our instance public IP address. Check your console for this information.

Open the PuTTY application and enter the IP address. In the "Host Name (or IP address)" field.

Note there are [various prerequisites and default usernames](#) that you may want to be aware of before proceeding.

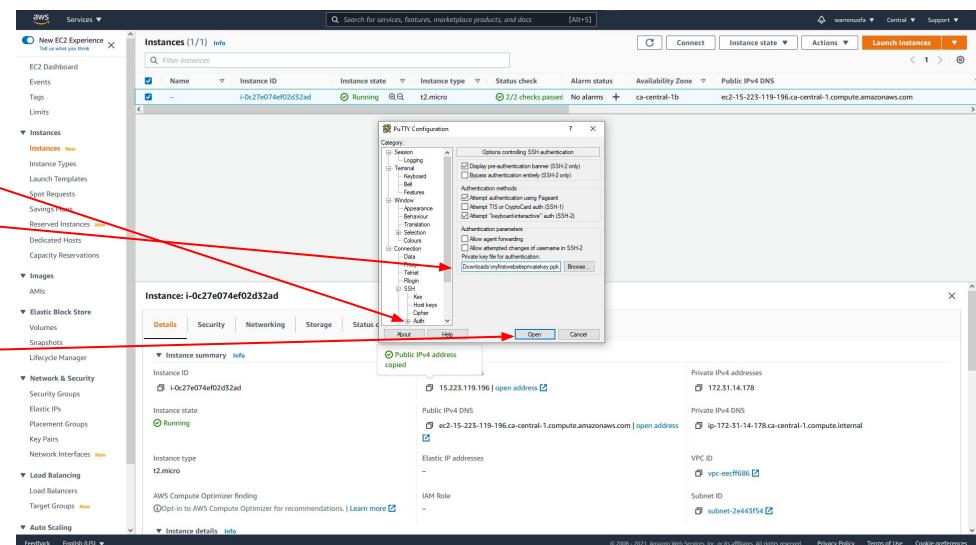


Add your Key(s) to the Session

Navigate to the “Connection,” then to the “SSH,” and then to the “Auth” category.

In this category, you have an option to browse for your key file.

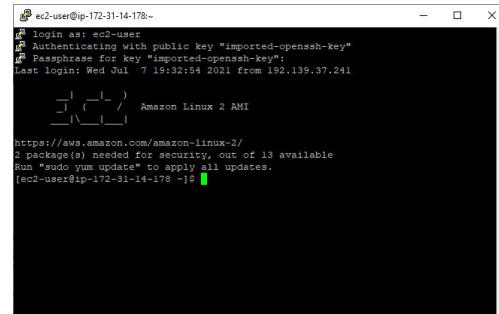
Once these steps are completed, you can click the open button to begin your session with the server instance.



SSH Terminal Access

Once you click “Open,” PuTTY will attempt to connect to the server. There may be a warning, confirming that you’re sure you’d like to proceed despite the server fingerprint not yet being familiar to your computer. If you’re sure you’re connecting to your server, proceed.

A terminal should soon appear asking for a username. Enter the username (the default is “ec2-user”, then (if you had entered a passphrase when generating your key) your passphrase.



The screenshot shows a PuTTY terminal window titled "ec2-user@ip-172-31-14-178:~". The session has been authenticated using a public key ("imported-openssh-key"). The last login was on Wednesday, July 7, 2021, at 19:32:54. The user is connected from IP address 152.139.37.241. The prompt shows the user's name and the server type ("Amazon Linux 2 AMI"). Below the prompt, there is a message about security updates: "https://aws.amazon.com/amazon-linux-2/" followed by "2 package(s) needed for security, out of 13 available". It also suggests running "sudo yum update" to apply all updates. The command history shows the user has run "ls" and is currently at the prompt "(ec2-user@ip-172-31-14-178 ~)".

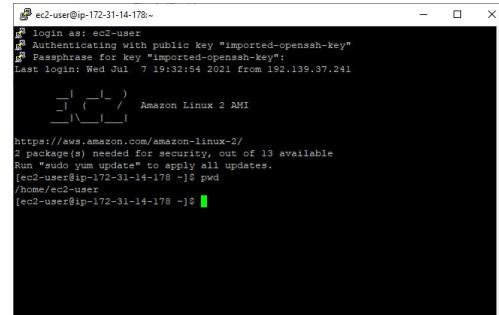
SSH Terminal

Once you are successfully connected, you are able to enter commands as if you were really sitting at the Linux machine itself!

This space can be used to create, edit, and delete files and directories, as well as install and update software on your EC2 instance.

Congratulations: “you’re in!”

Try running a simple command like “pwd” to check the current directory and prove that the text works.



The screenshot shows an SSH terminal window with the following text:

```
ec2-user@ip-172-31-14-178:~  
login as: ec2-user  
Authenticating with public key "imported-openssh-key"  
Passphrase for key "imported-openssh-key":  
Last login: Wed Jul  7 19:32:54 2021 from 192.139.37.241  
[ec2-user@ip-172-31-14-178 ~] $ ls  
[ec2-user@ip-172-31-14-178 ~] $ cd /home/ec2-user  
[ec2-user@ip-172-31-14-178 ~] $ pwd  
/home/ec2-user  
[ec2-user@ip-172-31-14-178 ~] $
```

Server Stack

Type of Website we'll Launch

A common and affordable “stack” for serving dynamic or static websites and pages is the “LAMP” stack. This consists of the following technologies:

- **Linux** (an operating system like CentOS, Ubuntu, etc.)
- **Apache** (an open-source server software)
- **MariaDB** (a database management system following MySQL syntax)
- **PHP** (a server-side scripting language capable of performing logical operations and interacting with a database)

This will be the example stack we work with today.

When setting up any type of hosting with a service like AWS, it is a good idea to first check the documentation. Many large companies like Amazon and Google offer tutorials, information, suggestions, and/or suggested resources to assist you in configuring an appropriate environment.

In this case, it would be a good idea to have a look at Amazon's “[Tutorial: Install a LAMP web server on the Amazon Linux 2](#)” article. It provides steps to help in this process.

Note there are many other popular stacks that are also supported.

Setting up the Stack and Preparing for Traffic

Update Packages

Firstly, we'll need to ensure that our software is up to date. We can run a command in our prompt:

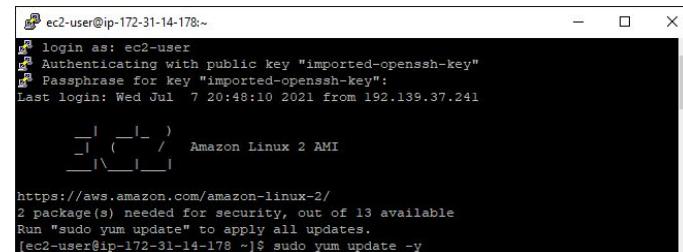
```
sudo yum update -y
```

Next, we'll want to ensure our server software is installed. We'll start with PHP and MariaDB (MySQL):

```
sudo amazon-linux-extras install -y lamp-mariadb10.2-php7.2 php7.2
```

Now for Apache and the server software for MariaDB:

```
sudo yum install -y httpd mariadb-server
```



The screenshot shows a terminal window titled "ec2-user@ip-172-31-14-178:~". It displays the following text:

```
[ec2-user@ip-172-31-14-178:~]$ login as: ec2-user
[ec2-user@ip-172-31-14-178:~]$ Authenticating with public key "imported-openssh-key"
[ec2-user@ip-172-31-14-178:~]$ Passphrase for key "imported-openssh-key":
Last login: Wed Jul  7 20:48:10 2021 from 192.139.37.241
[ec2-user@ip-172-31-14-178:~]$ Amazon Linux 2 AMI
[ec2-user@ip-172-31-14-178:~]$ https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-172-31-14-178:~]$ 2 package(s) needed for security, out of 13 available
[ec2-user@ip-172-31-14-178:~]$ Run "sudo yum update" to apply all updates.
[ec2-user@ip-172-31-14-178:~]$ sudo yum update -y
```

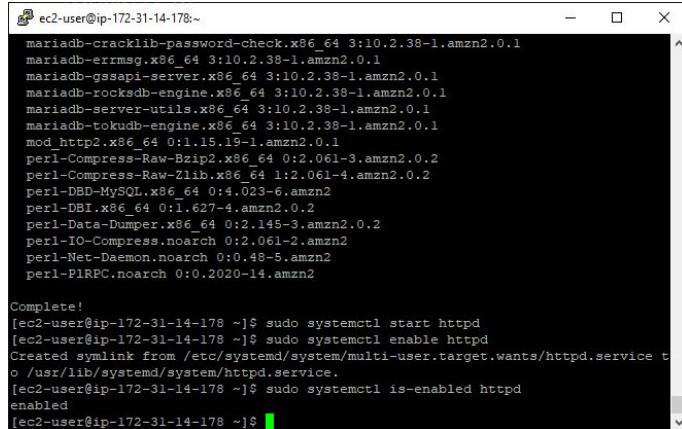
Start Web Server

To start your Apache server, run:

```
sudo systemctl start httpd
```

For our convenience, we can tell Apache to run each time the system boots as well:

```
sudo systemctl enable httpd
```



```
[ec2-user@ip-172-31-14-178 ~]$ sudo systemctl start httpd
[ec2-user@ip-172-31-14-178 ~]$ sudo systemctl enable httpd
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[ec2-user@ip-172-31-14-178 ~]$ sudo systemctl is-enabled httpd
enabled
[ec2-user@ip-172-31-14-178 ~]$
```

Open Port 80 for web Browser Traffic

Navigate to your console, find the Security tab, and open your security group. Edit the inbound rules, and add a new one.

Set its type to “HTTP” traffic (port 80), this will allow users in internet browsers to view web pages from your server.

If all is done correctly, navigate in your web browser to the IPv4 address assigned to your instance and you should be greeted with the Apache test page!

Test Page

This page is used to test the proper operation of the Apache HTTP server after it has been installed. If you can read this page, it means that the Apache HTTP server installed at this site is working properly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to tell the administrators of this website know that you've seen this page instead of the page you expected, you should send them an e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

If you are the website administrator:

You may now add content to the directory /var/www/html/. Note that until you do so, people visiting your website will see this page, and not your content. To prevent this page from ever being used, follow the instructions in the file /etc/httpd/conf.d/welcome.conf.

You are free to use the image below on web sites powered by the Apache HTTP Server:



AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

AWS CloudWatch Metrics interface showing a metric named 'CPU Utilization' over time. The chart displays data points for various instances, with a prominent peak around July 2nd, 2021, reaching approximately 100% utilization.

Populating your Website with Files and Pages

Populating your Website

As the test page states: content, like web pages (HTML files), can be added to the /var/www/html directory in your instance. Once added they will be displayed *instead* of this test page.

It is these files that make up your live website!

We can write files directly in the terminal, or we can use a tool like WinSCP or FileZilla to connect and transfer files via SFTP. Many people prefer GUI SFTP programs as they are typically easier to use and understand than their command-line counterparts.

FileZilla is multi-platform (works on Linux, Mac, and Windows) so we'll be having a look at this option.

[Navigate to the website and install the software to your machine.](#)



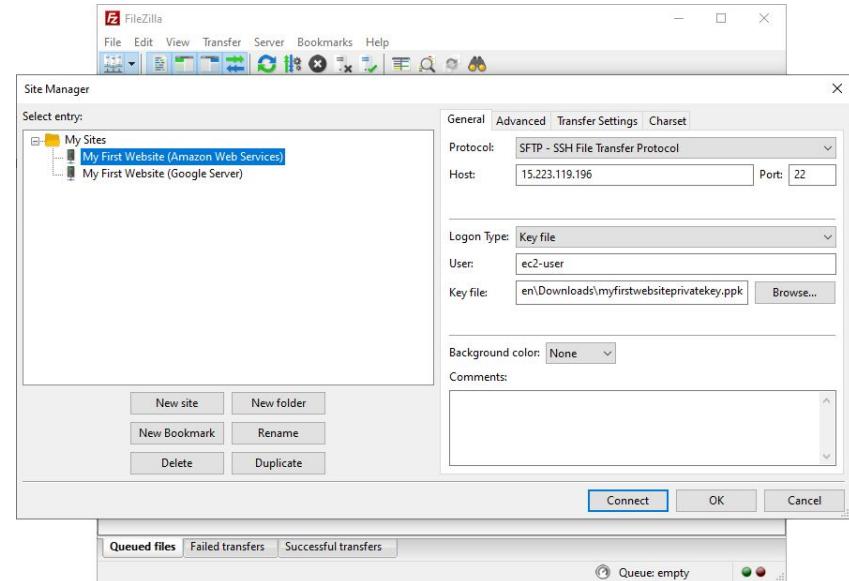
Connecting via FileZilla

Once installed, open the FileZilla application. Click “File→Site Manager” so that we may set up our connection information.

Click the “New site” button, and we can begin setting up our connection information. Select the “SFTP - SSH File Transfer Protocol” option, enter your host IP address, and port 22.

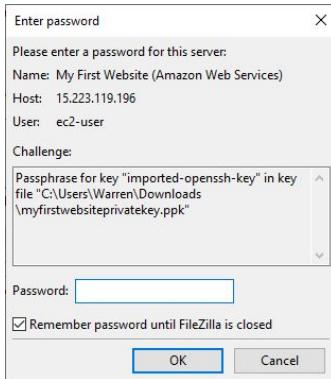
For “Logon Type,” select “Key file.” Enter your username (default “ec2-user”.) Browse and select your “Key file.”

Once this information is all entered and/or selected, click the “Connect” button.

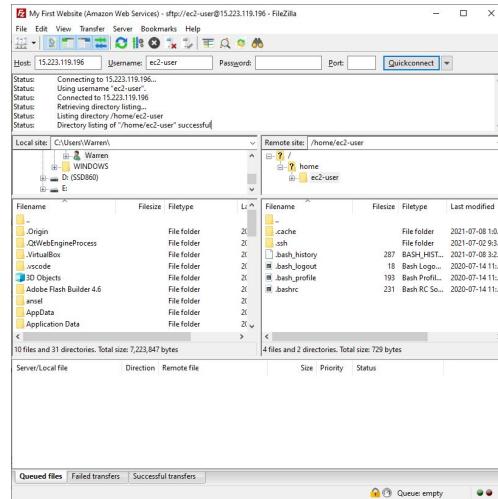


Connect to the Instance

If you had entered a password for your key, there may be a dialogue at this step that asks for it. Enter it, if applicable, and press “OK.”



Assuming your information was entered correctly, you should now be connected to your server:

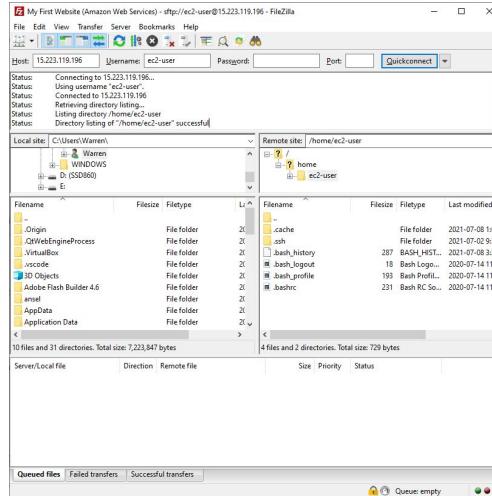


Navigating Filesystems

Note that once connected, two panes populate with folders and/or files:

- **Left Pane**
Your local filesystem (the files and folders on your personal computer.)
- **Right Pane**
The external/foreign filesystem (the files and folders on your EC2 instance.)

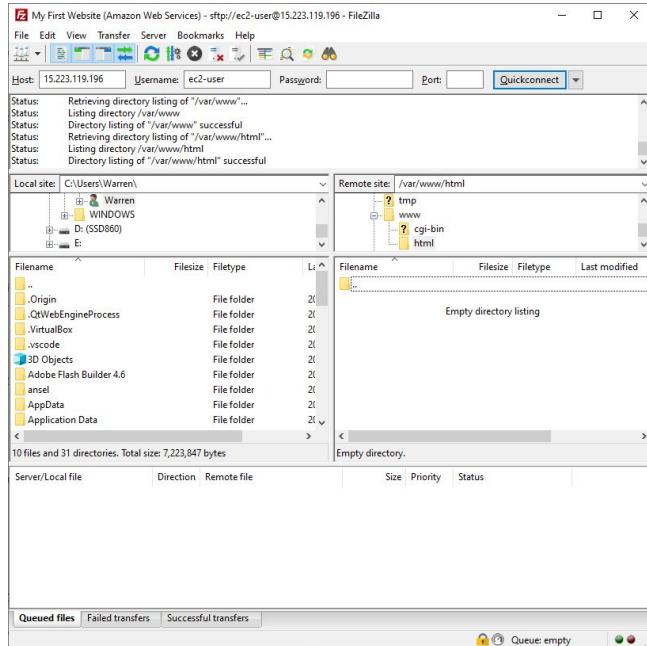
Careful, in this application, clicking on a file will automatically copy it from the active location in one pane to the other.



Find the Website Folder

In the right pane navigate to /var/www/html

This is the folder where we can add web assets. Once web pages, stylesheets, scripts, images, and more are added to this folder and/or subfolders in this location, we'll have a website that is accessible to the world!



Start Small

Let's create a document on our local computer called "index.html". **Make a note of where you place this!** This will act as the default document loaded when you either load a domain attached to your IP address, or the IP address itself (depending on your server configuration.) We'll populate the document like so:

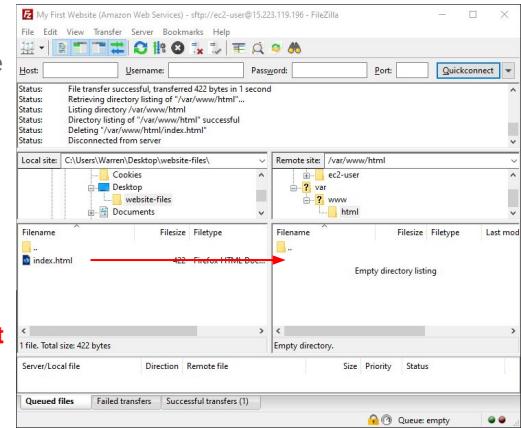
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>My First Website!</title>
</head>
<body>
<h1>My First Website!</h1>
<p>
Hello, my name is Warren, nice to meet you!
This is my brand new website!
</p>
</body>
</html>
```

Fill in the file:

```
<!-- index.html -->
<html lang="en">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>My First Website!</title>
</head>
<body>
<h1>My First Website!</h1>
<p>
Hello, my name is Warren, nice to meet you!
This is my brand new website!
</p>
</body>
</html>
```

Transfer it to the server:

*Don't forget to save the file before trying to transfer!



Did you encounter a permissions error during transfer? See the next slide!

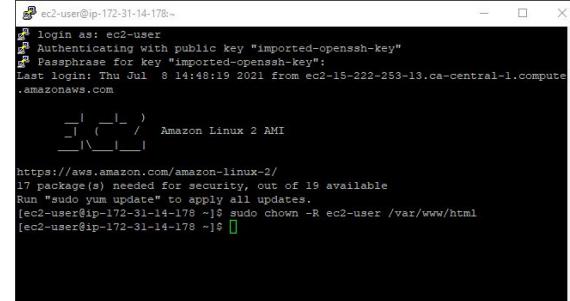
File and Folder Ownership

Linux (and most any operating system) will decide who or what “owns” certain files and folders on a computer in order to protect them. This is both a security measure, and assists in preventing human error (like accidentally changing or deleting a file you didn’t mean to!)

If there is a protected folder, and we have appropriate access with our user, we can redefine or add to the owner list when we need to. In the case that your /var/www/html folder is protected from write and delete access in regards to your “ec2-user” user, we’ll send our instance the following command (use PuTTY or an equivalent for this.)

Enter the following (and recall that “sudo” is a command that we should be very careful about):

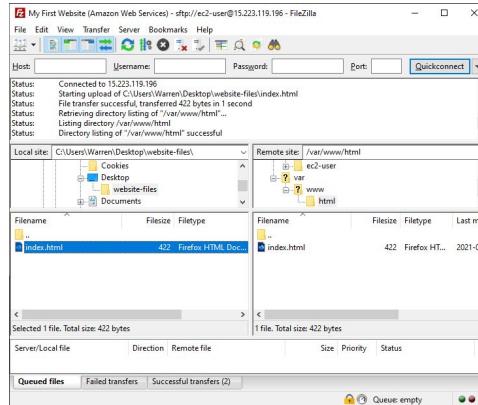
```
sudo chown -R ec2-user /var/www/html
```



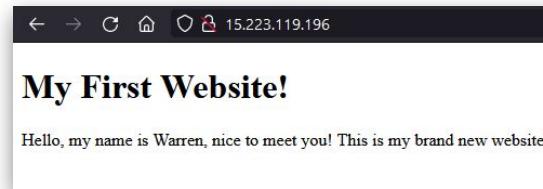
The screenshot shows a terminal window titled "ec2-user@ip-172-31-14-178:~". The session is authenticated via SSH using a public key. The terminal displays the command "sudo chown -R ec2-user /var/www/html" being run, which changes the ownership of the specified directory to the "ec2-user" account. The terminal is running on an Amazon Linux 2 AMI.

Transfer the File and Test in the Browser

Once the user is an owner, you should have full access to upload your “index.html” file in your SFTP client. Drag it over (if it hadn’t worked before.)



Once you’re done... check your website in the browser again (type or paste your instance IP address into the address bar.) If all was successful, you should see our new website!



As you learn more about web development, you can begin adding more and more to the site.

Congratulations!

Helpful Tutorials and Documentation

Ensure you familiarise yourself with the documentation for the tools offered by Amazon (or any other host you try out) as you'll likely need to refer to these if you want to try new things, forget steps you've carried out before, or want to explore what the product is capable of.

Some articles good to keep on hand to get started with include...

- [Getting Started with AWS](#)
- [Pricing](#)
- [AWS Pricing Calculator](#)
- [AWS Free Tier](#)
- [Security Groups](#)
- [Connecting Using Third-Party Tools](#)
- [Installing a LAMP web server on Amazon Linux 2](#)
- Additional Reading:
 - [Configure SSL/TLS](#)
 - [Route a Domain to an EC2 Instance](#)

Recommended Readings

If you'd like a more in-depth look at hosting and configuration:

- [Geewax, J. J. \(September 2018\). Google Cloud Platform in Action. Manning Publications.](#)