
amazon
web services™
EC2

Table of Content (EC2)

1) EC2 Introduction

2) Instance –windows, Linux, HTTP, Addding script, Filezilla(Transfer files from local system to cloud server) software, Enable termination protection, Change shutdown behavior, Enable Password Authentication (Access through password)

3) Tagging

4) Launch template

5) Elastic IP

6) Security group

7) Keypair

8) ENI

9) Customize AMI – Backup Restore, Migration

AWS 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'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.

Features of Amazon EC2-Slide 1

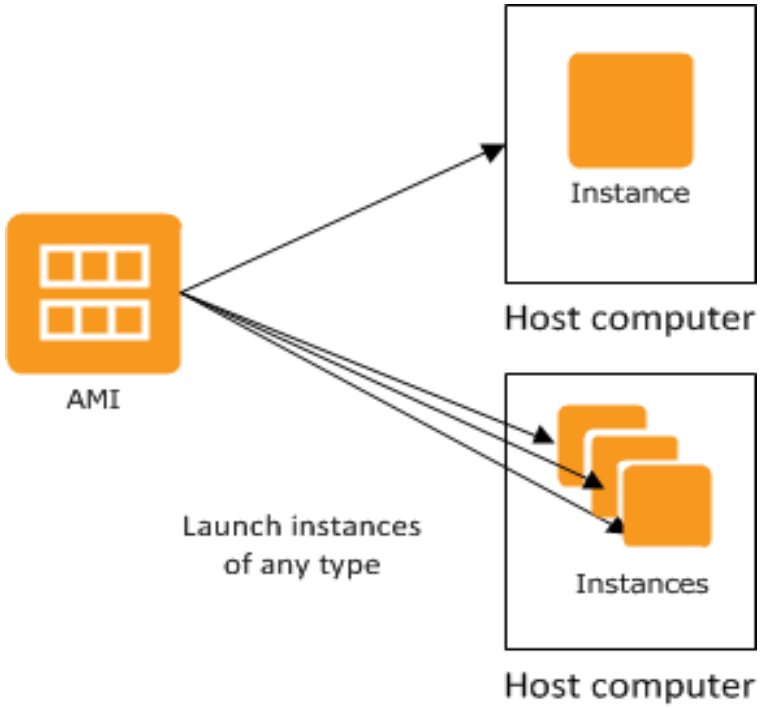
- Virtual computing environments, known as ***instances.***
- Preconfigured templates for your instances, known as *Amazon Machine Images (**AMIs**)*, that package the bits you need for your server (including the operating system and additional software).
- Various configurations of CPU, memory, storage, and networking capacity for your instances, known as ***instance types***
- Secure login information for your instances using ***key pairs*** (AWS stores the public key, and you store the private key in a secure place).
- Multiple physical locations for your resources, such as instances and Amazon EBS volumes, known as ***Regions*** **and** ***Availability Zones***

Features of Amazon EC2—Slide2

- ✓ Storage volumes for temporary data that's deleted when you stop or terminate your instance, known as ***instance store volumes***
- ✓ Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS), known as ***Amazon EBS volumes***
- ✓ A firewall that enables you to specify the protocols, ports, and source IP ranges that can reach your instances using ***security groups***
- ✓ Static IPv4 addresses for dynamic cloud computing, known as ***Elastic IP addresses***
- ✓ Virtual networks you can create that are logically isolated from the rest of the AWS cloud, and that you can optionally connect to your own network, known as ***virtual private clouds*** **(VPCs)**

Instances and AMIs

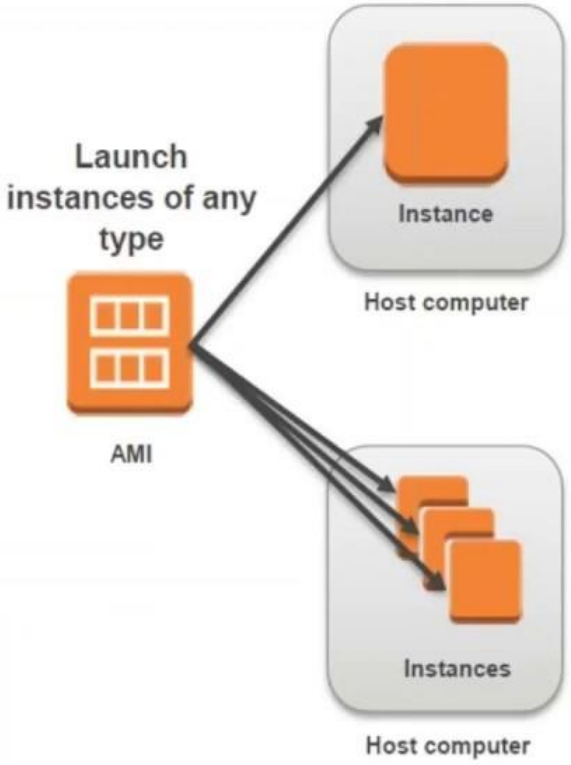
An *Amazon Machine Image (AMI)* is a template that contains a software configuration (for example, an operating system, an application server, and applications). From an AMI, you launch an *instance*, which is a copy of the AMI running as a virtual server in the cloud.



EC2: Amazon Machine Image - AMI

An AMI includes the following:

- A template for the ***root volume*** for the instance (for example, an operating system, an application server, and applications).
- Launch ***permissions*** that control which AWS accounts can use the AMI to launch instances.
- A block device mapping that specifies the ***volumes to attach*** to the instance when i is launched.



Amazon Machine Images (AMI)

An Amazon Machine Image (AMI) provides the information required to launch an instance. You must specify an AMI when you launch an instance.

You can launch multiple instances from a single AMI when you need multiple instances with the same configuration.

You can use different AMIs to launch instances when you need instances with different configurations.

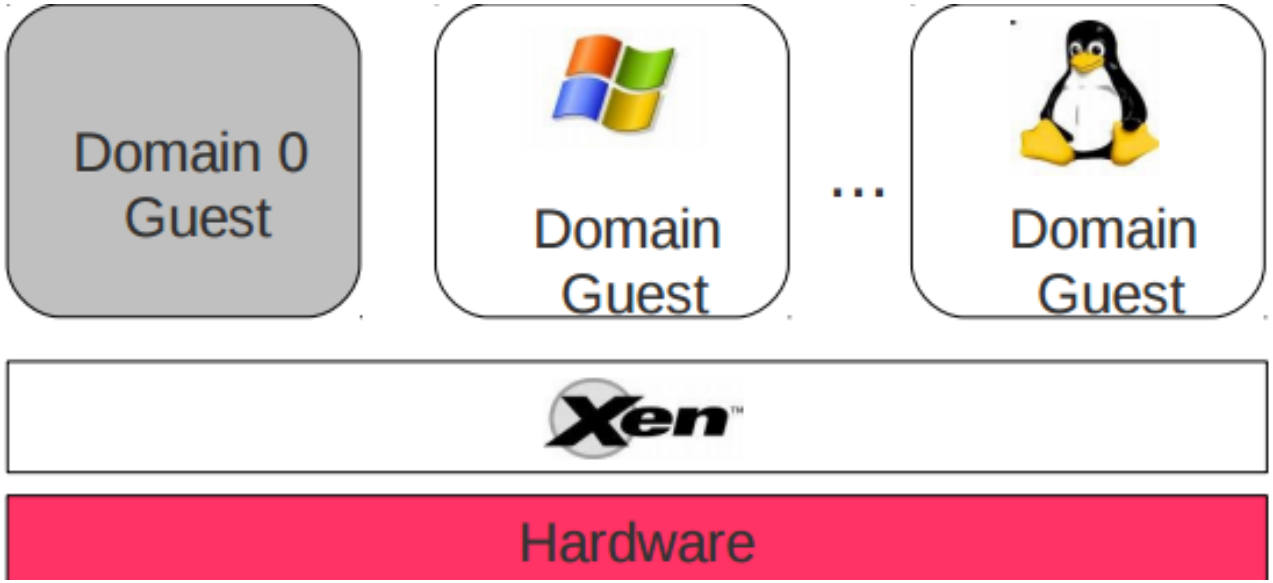
AWS Market Place and Community AMI

- **Community AMIs:** Whenever you create an AMI, you can add permissions to it to make it public. In that case, it goes to "community AMIs". These are AMIs that comes from AWS users, and are not verified by AWS
- **Maketplace:** This is a whole service at AWS, and all AMIs here are verified by AWS. It is basically used for software vendors to sell their products through AWS. The customers will be billed by AWS only, but then AWS will pay the AMI owner in return. For Centos this is a bit particular, as this is a free distribution. But been in the marketplace conforts the users that the AMI is safe

AWS AMI and the Xen hypervisor

Every AWS AMI uses the Xen hypervisor on bare metal. Xen offers two kinds of virtualization: *HVM (Hardware Virtual Machine) and PV (Paravirtualization).*

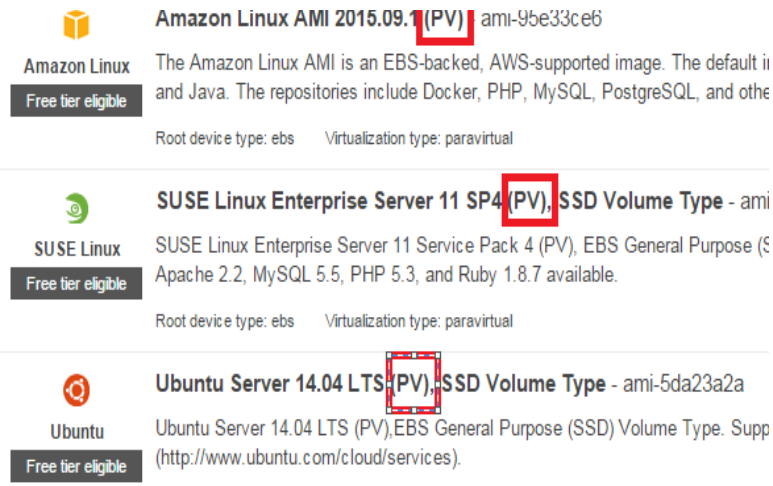
Virtual machines (also known as guests) run on top of a hypervisor. The hypervisor takes care of CPU scheduling and memory partitioning.



Linux AMI virtualization

Linux Amazon Machine Images use one of two types of virtualization: paravirtual (PV) or hardware virtual machine (HVM).

For the best performance, we recommend that you use current generation instance types and HVM AMIs when you launch your instances.



Instance types

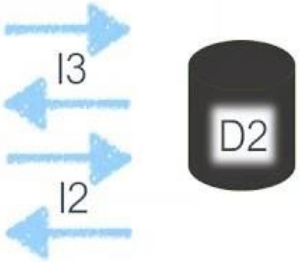
- ✓ When you launch an instance, the *instance type* that you specify determines the hardware of the host computer used for your instance.
- ✓ Each instance type offers different compute, memory, and storage capabilities and are grouped in instance families based on these capabilities.
- ✓ Select an instance type based on the requirements of the application or software that you plan to run on your instance.
- ✓ Free tier Eligible Instance type : t2.micro

EC2: Instance Types

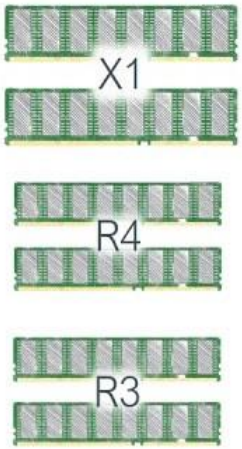
General purpose

Compute optimized

Storage and I/O optimized



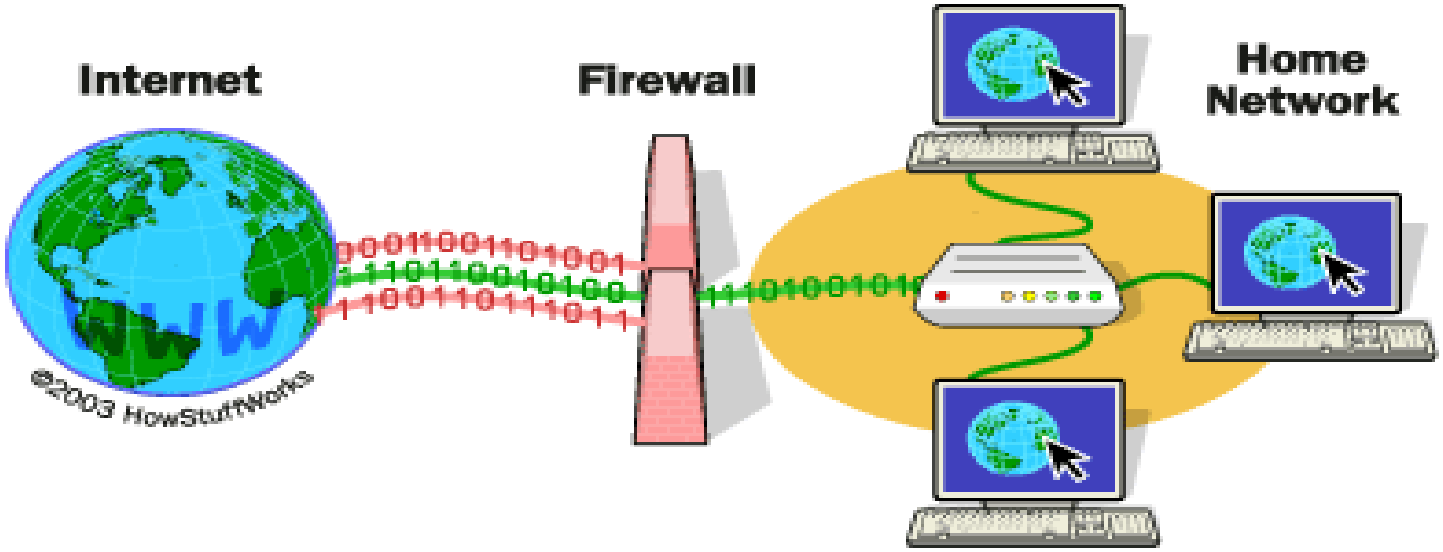
Memory optimized

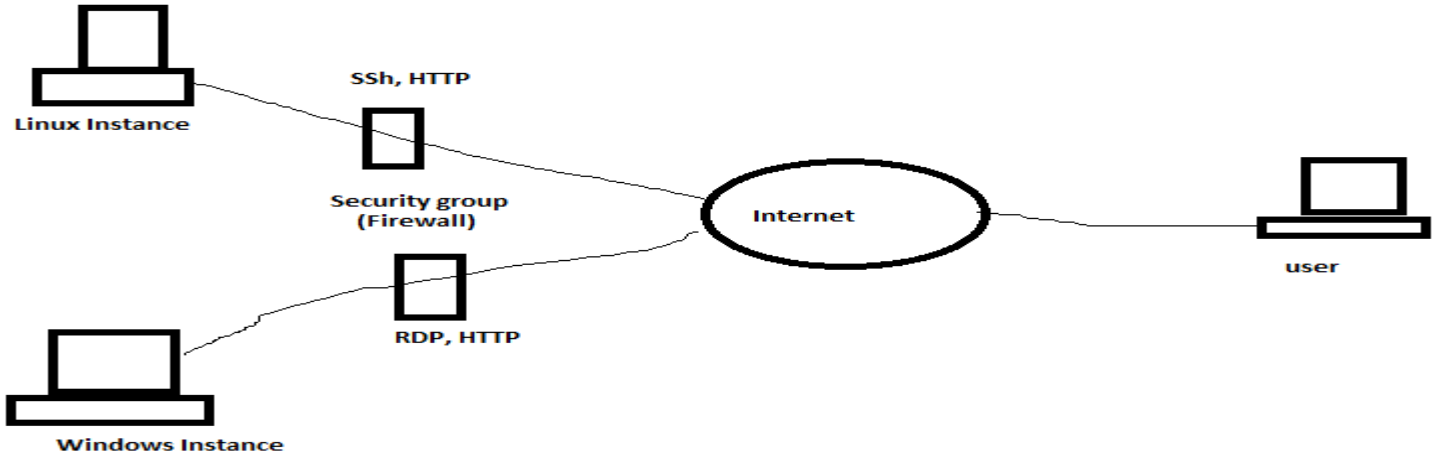


GPU or FPGA enabled

Amazon EC2 security groups

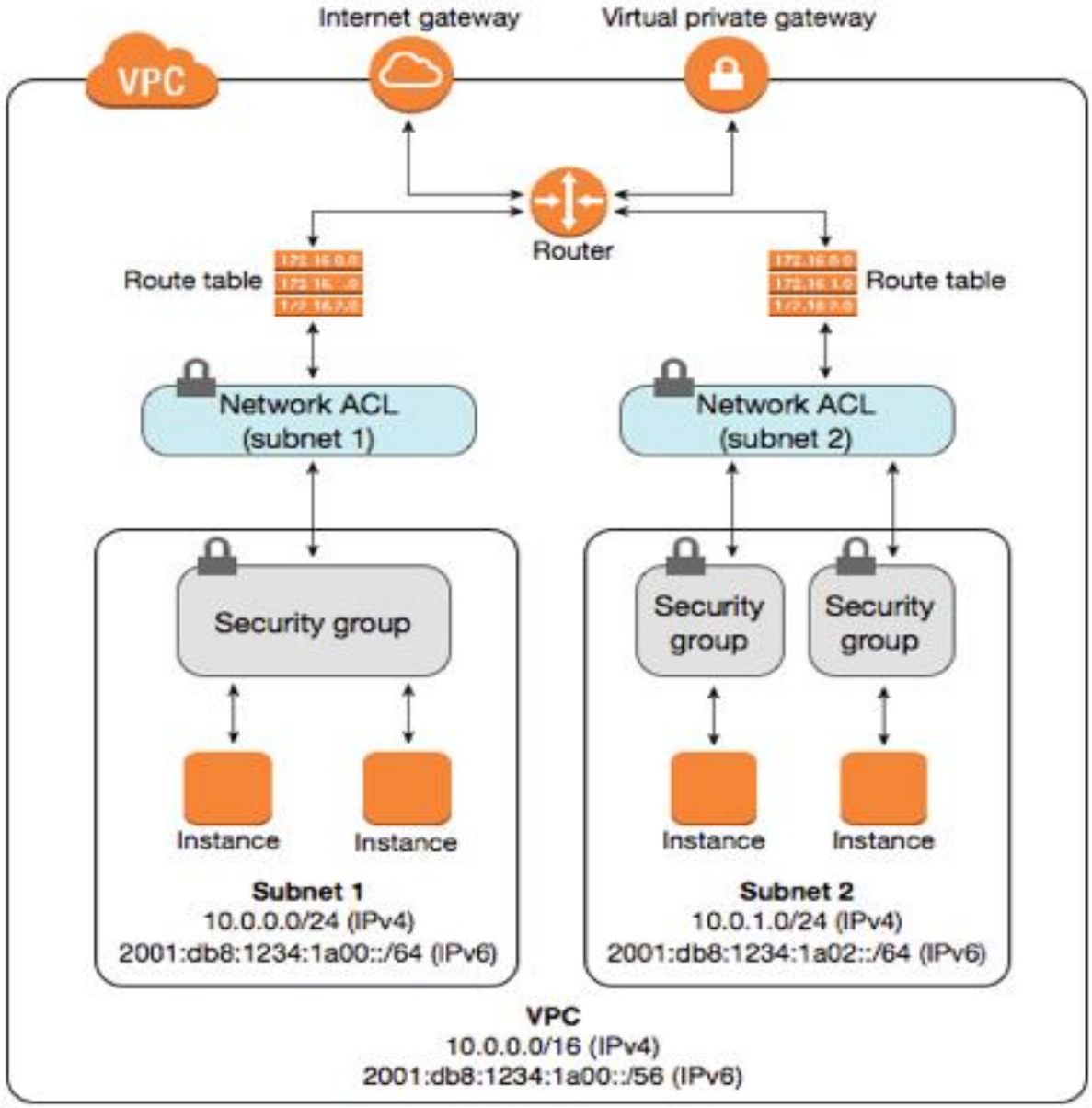
- ✓ A *security group* acts as a virtual firewall for your EC2 instances to control incoming and outgoing traffic.
- ✓ Inbound rules control the incoming traffic to your instance, and outbound rules control the outgoing traffic from your instance.
- ✓ When you launch an instance, you can specify one or more security groups.
- ✓ If you don't specify a security group, Amazon EC2 uses the default security group.
- ✓ You can add rules to each security group that allow traffic to or from its associated instances.
- ✓ You can modify the rules for a security group at any time. New and modified rules are automatically applied to all instances that are associated with the security group.


Internet
Firewall
Home Network
WWW
©2003 HowStuffWorks


Linux Instance
SSh, HTTP
Security group (Firewall)
Internet
user
RDP, HTTP
Windows Instance

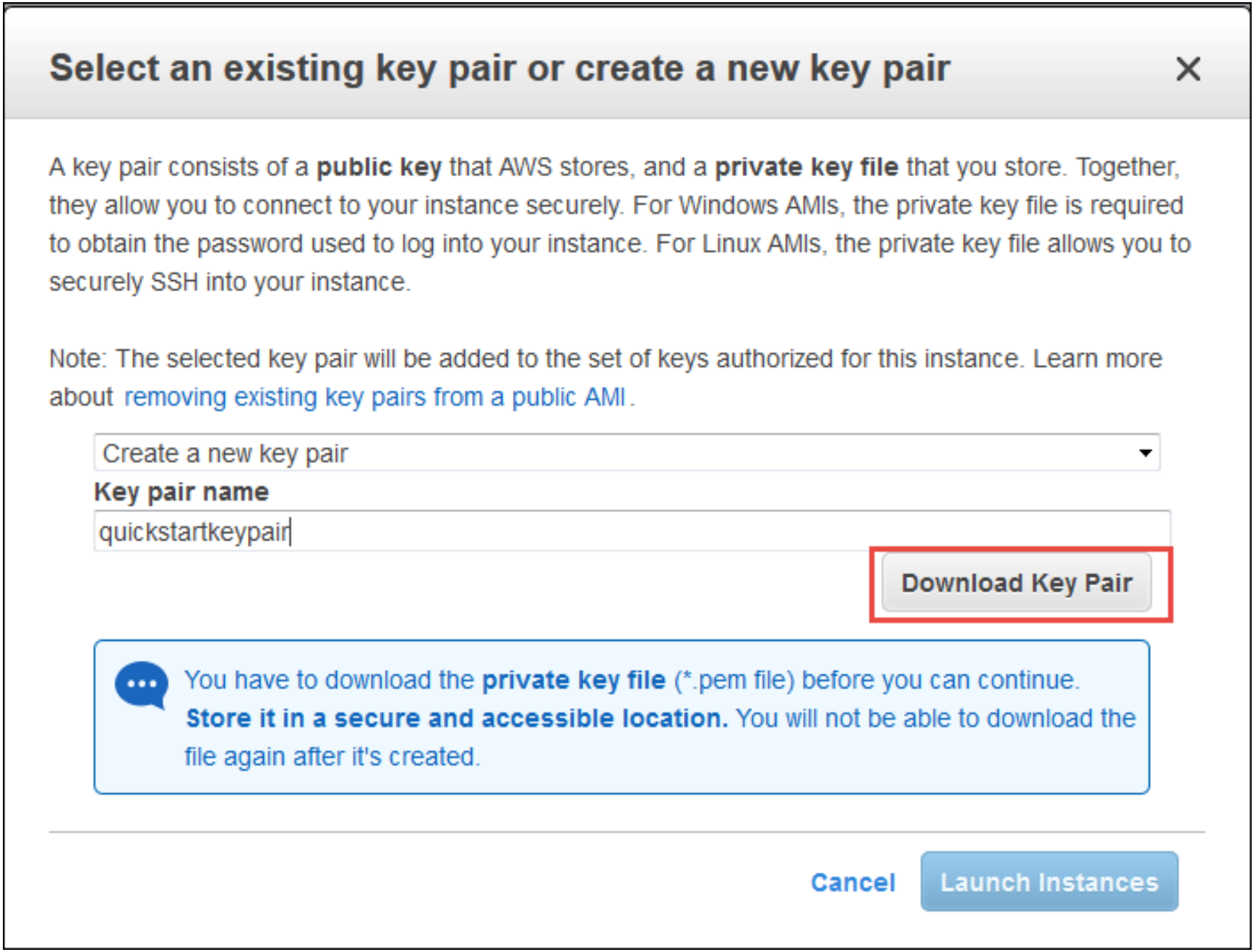
Well known Port number

1) HTTP ---80
2) HTTPS --- 443
3) RDP --- 3389
4) SSH --- 22
5) NFS -- 2049
6) MSSQL– 1433
7) MySQL/Aurora – 3306
8) Redshift -- 5439


Internet gateway
Virtual private gateway
VPC
Router
Route table
Route table
Network ACL
(subnet 1)
Network ACL
(subnet 2)
Security group
Security group
Security group
Instance
Instance
Instance
Instance
Subnet 1
10.0.0.0/24 (IPv4)
2001:db8:1234:1a00::/64 (IPv6)
Subnet 2
10.0.1.0/24 (IPv4)
2001:db8:1234:1a02::/64 (IPv6)
VPC
10.0.0.0/16 (IPv4)
2001:db8:1234:1a00::/56 (IPv6)

Amazon EC2 key pairs

- ✓ A key pair, consisting of a private key and a public key, is a set of security credentials that you use to prove your identity when connecting to an instance.
- ✓ Amazon EC2 stores the public key, and you store the private key. You use the private key, instead of a password, to securely access your instances.
- ✓ Anyone who possesses your private keys can connect to your instances, so it's important that you store your private keys in a secure place.


Select an existing key pair or create a new key pair
A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.
Create a new key pair
Key pair name
quickstartkeypair
Download Key Pair
You have to download the private key file (*.pem file) before you can continue. Store it in a secure and accessible location. You will not be able to download the file again after it's created.
Cancel
Launch Instances

Create launch Template

1) Open EC2---Launch template –create template –Full the required detail –create.

2) Select the created template –Action –launch from template—create—view launched instance.