**AWS Login**

* **URL:** To log in to AWS, you use the URL aws.amazon.com.
* **Root User:** This is the account with full administrative access to all resources in your AWS account.
* **IAM User:** IAM (Identity and Access Management) users are created to manage access permissions. They have restricted access based on their assigned policies.
* **Account ID:** A unique identifier for your AWS account (e.g., 717279735377).

**AWS Infrastructure**

* **Availability Zone:** A distinct location within a region that is engineered to be isolated from failures in other Availability Zones. AWS has 108 Availability Zones worldwide.
* **Region:** A geographic area where AWS has multiple data centers (Availability Zones). AWS has 34 regions globally.
* **Cloudping.info:** A website that helps you measure the latency between your location and various AWS regions.

**Creating a Server on AWS**

**On-Premises vs. Cloud:**

* **On-Premises:** Refers to local servers and infrastructure within your own physical location.
* **Cloud:** Refers to servers and infrastructure hosted by AWS.

**Corresponding Terms:**

* **Server (On-Premises) = Instance or EC2 (Elastic Compute Cloud) (Cloud):** An EC2 instance is a virtual server in AWS.
* **ISO Image (On-Premises) = AMI (Amazon Machine Image) (Cloud):** AMI is a pre-configured template for instances that includes the operating system and applications.
* **Hardware (On-Premises) = Instance Type (Cloud):** Instance types define the hardware configurations (e.g., CPU, memory) for instances.
* **Password (On-Premises) = Key Pair (Cloud):** A key pair is used to securely connect to instances. It consists of a public key (stored in AWS) and a private key (stored on your local machine).
* **Firewall (On-Premises) = Security Group (SG) (Cloud):** Security groups act as virtual firewalls for instances, controlling inbound and outbound traffic.
* **Network (On-Premises) = VPC (Virtual Private Cloud) (Cloud):** A VPC is a logically isolated network within AWS.
* **Hard Disk (On-Premises) = Storage or Volume (Cloud):** Storage in AWS refers to services like EBS (Elastic Block Store), which provides persistent block storage.

**Creating a Windows Instance**

When creating a Windows instance in AWS, you typically follow these steps:

1. **Login to AWS:** Go to aws.amazon.com and log in.
2. **Navigate to EC2 Dashboard:** Use the search bar or navigate to the EC2 service.
3. **Launch Instance:** Click "Launch Instance" to start the process.
4. **Choose an Amazon Machine Image (AMI):** Select a Windows AMI.
5. **Choose an Instance Type:** Select the desired instance type based on your requirements.
6. **Configure Instance Details:** Customize settings like the number of instances, network settings (VPC and subnets), and IAM role.
7. **Add Storage:** Specify the storage requirements for the instance.
8. **Add Tags (Optional):** Assign tags for easy identification.
9. **Configure Security Group:** Set up security group rules to allow necessary inbound and outbound traffic.
10. **Review and Launch:** Review your settings and launch the instance.
11. **Create Key Pair:** Create a new key pair or use an existing one. Download the private key file (.pem) as it will be needed to access the instance.
12. **Connect to Instance:** Once the instance is running, use an SSH client (e.g., PuTTY) to connect using the private key.

**Instance Management**

* **Administrator:** For Windows instances, the default administrator username is typically Administrator.

**Windows Instance**

* **User:**
  + The default administrator user for a Windows instance is usually named Administrator.

**Taking the Console**

You can connect to the console (graphical interface) of a Windows instance in two ways:

1. **Remote Desktop Connection (RDP):**
   * Use the Remote Desktop Protocol to connect to the instance.

**Making Users**

* You can create new users on your instance, both in Windows and Linux.

**Making a Linux Instance**

* **Linux Distributions Supported:**
  + AWS Linux
  + RHEL (Red Hat Enterprise Linux)
  + CentOS
  + Ubuntu
* **Default Users:**
  + **AWS Linux:** The default user is ec2-user.
  + **Ubuntu:** The default user is ubuntu.

**Privileged vs. Non-Privileged Users**

* **Privileged User (Root):** Has full administrative access to the system.
* **Non-Privileged User:** Has limited access and permissions.

**Taking the Console of Linux**

You can connect to a Linux instance using three ways:

1. **Command Line (SSH):**
   * Change directory to where your private key is stored:

cd downloads

* + Use SSH to connect to the instance:

ssh ec2-user@3.22.224.93 -i keyname.pem

1. **Third-Party Tools (PuTTY):**
   * Use PuTTY to connect to the instance.
   * .pem files can be converted to .ppk files using PuTTYgen for use with PuTTY.

**Enabling Root User in SSH**

To enable the root user to log in via SSH:

1. Edit the SSH configuration file:

vim /etc/ssh/sshd\_config

1. Change the following settings:

PermitRootLogin yes

PasswordAuthentication yes

1. Save and exit the file:

:wq

1. Restart the SSH service:

systemctl restart sshd

1. Set a password for the root user:

passwd root

**Making a Webserver**

**Website**

* **Windows:** Use IIS (Internet Information Services) to set up a web server.
* **Linux:** Use the Apache server to set up a web server.

**Setting Up a Webserver in Windows**

1. **IIS (Internet Information Services):**
   * The default directory for web content is usually C:/inetpub/wwwroot.

**Setting Up an Apache Server**

1. **Install Apache on a Linux Instance:**
   * Commands may vary based on the Linux distribution you're using.
2. **Configure Users and Permissions:**
   * Create necessary user accounts and set appropriate permissions.
3. **Deploy Your Website:**
   * Place your website files in the appropriate directory (e.g., /var/www/html for Apache).

**Summary**

* **Instances:** Virtual servers in AWS, including Windows and Linux instances.
* **Users:** Default users for different Linux distributions, and privileged vs. non-privileged users.
* **Connecting to Instances:** Using SSH or third-party tools like PuTTY.
* **Webservers:** Setting up IIS for Windows and Apache for Linux.