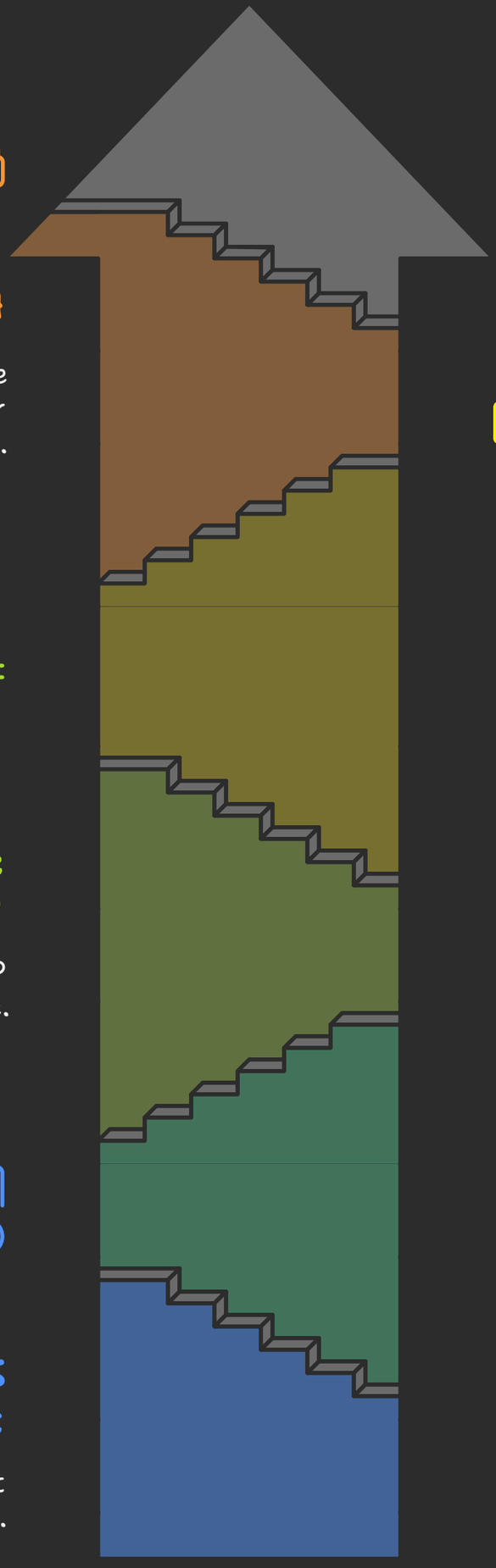
Hosting on AWS: A Step-by-Step Guide

This document provides a comprehensive guide to hosting on Amazon Web Services (AWS). It covers the essential steps, from creating an AWS account to connecting to your EC2 instance via SSH. This guide aims to simplify the process and equip you with the knowledge to get your server up and running on AWS.

Steps to Host on AWS



Connect via SSH

Establish a secure connection to your server.

Modify Inbound Rules

Configure security settings for your instance.

Allocate Elastic IP

Assign a static IP address to your instance.

Launch EC2 Instance

Start your virtual server on AWS.

Create AWS Account

Set up your AWS account to begin hosting.

1. Create an AWS Account

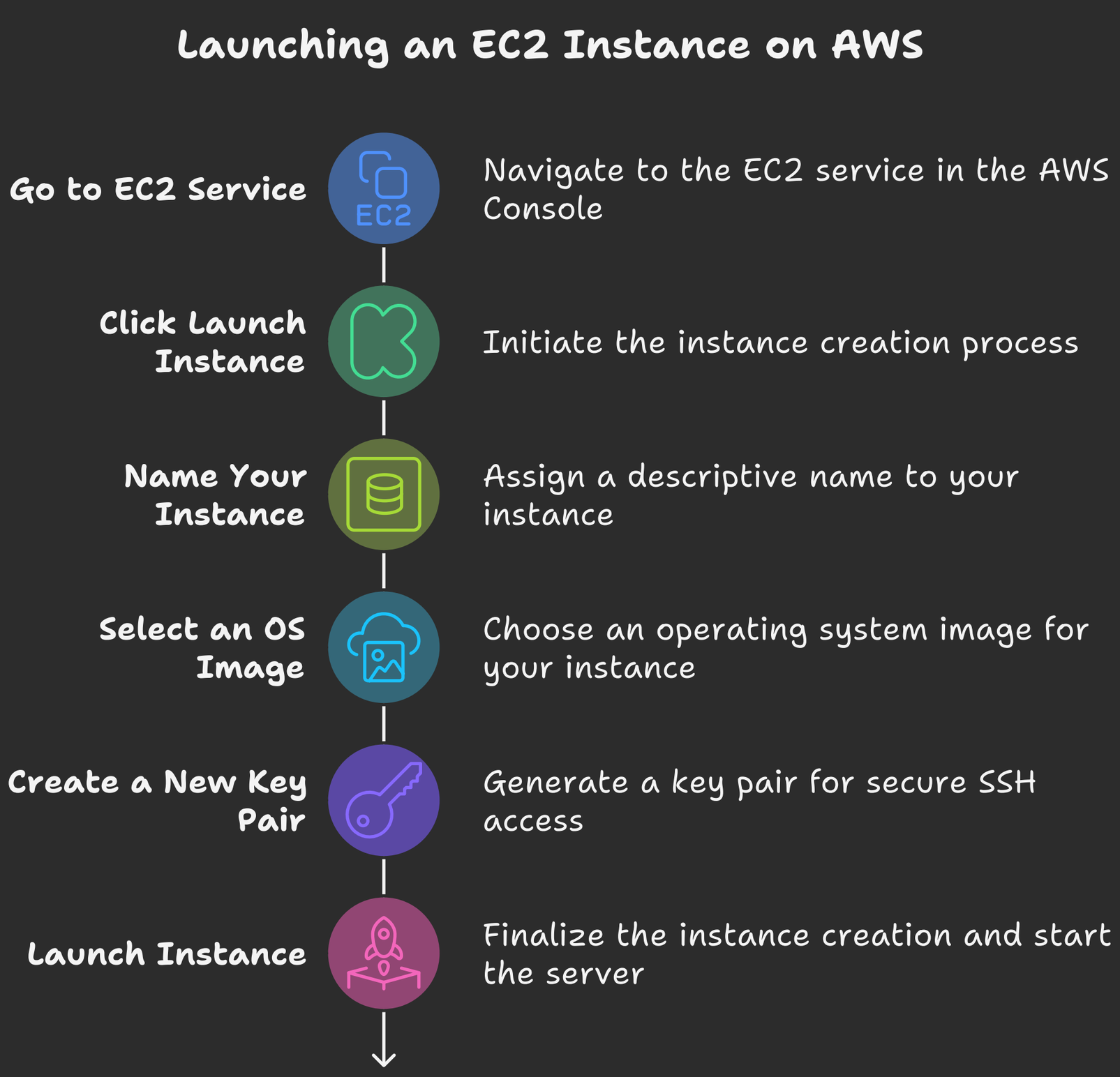
The first step to hosting on AWS is creating an account. Follow these instructions:

- **Go to the official AWS website:** Navigate to https://aws.amazon.com in your web browser.
- **Click Sign Up:** Locate and click the "Sign Up" button to begin the registration process.
- **Choose Root User:** Select "Root User" as the account type. This provides full access to all AWS services and resources.
- **Enter and Verify Email:** Enter your email address and follow the instructions to verify it. AWS will send a verification code to your inbox.
- **Complete CAPTCHA:** Complete the CAPTCHA verification to confirm you are not a bot.
- **Fill in Personal and Billing Information:** Provide your personal and billing information, including your name, address, and payment details.
    - **Note:** To activate the free tier, your credit/debit card must have at least $1 available. AWS uses this to verify your identity.
- **Enter and Verify Phone Number:** Enter your phone number and verify it via SMS. AWS will send a verification code to your phone.
- **Complete Registration:** After completing all the steps, you will be redirected to the AWS Management Console.

2. Launch an EC2 Instance

An EC2 instance is a virtual server in the AWS cloud. Here's how to launch one:

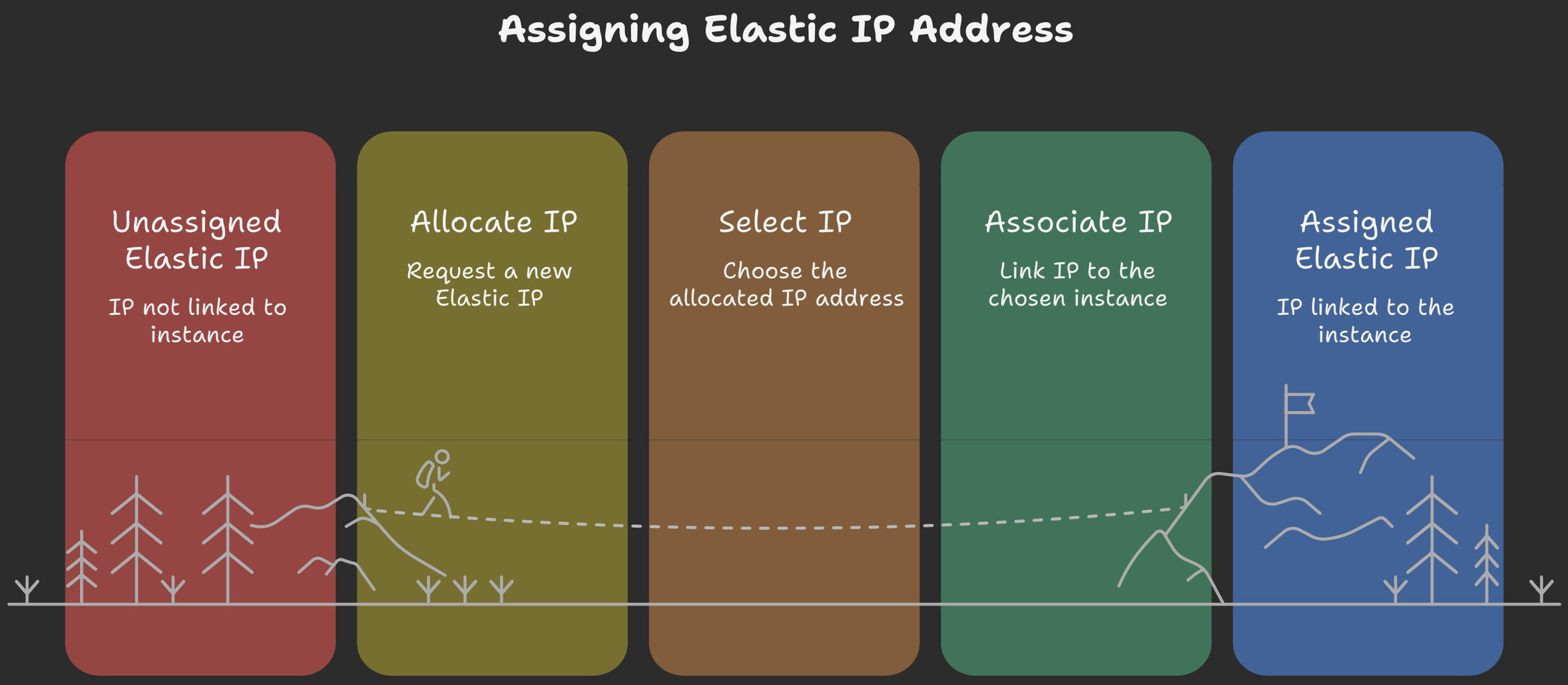
- **Go to EC2 Service:** From the AWS Console, search for and navigate to the EC2 service.
- **Click Launch Instance:** Click on "Launch Instance" from either the dashboard or the Instances page.
- **Name Your Instance:** Give your instance a descriptive name (e.g., "MyServer").
- **Select an OS Image:** Choose an operating system (OS) image, also known as an Amazon Machine Image (AMI). Ubuntu is a popular Linux distribution and a good choice for many applications.
- **Create a New Key Pair:** Create a new Key Pair for secure SSH access to your instance.
    - Choose the .pem format.
    - Download and securely store the .pem file on your computer. This file is crucial for connecting to your instance. **Keep it safe and never share it.**
- **Launch Instance:** Click "Launch Instance" to create the server. AWS will provision the resources and start your instance.



3. Allocate an Elastic IP Address

An Elastic IP address is a static, public IP address that you can associate with your EC2 instance. This ensures that your instance has the same IP address even if it's stopped and restarted.

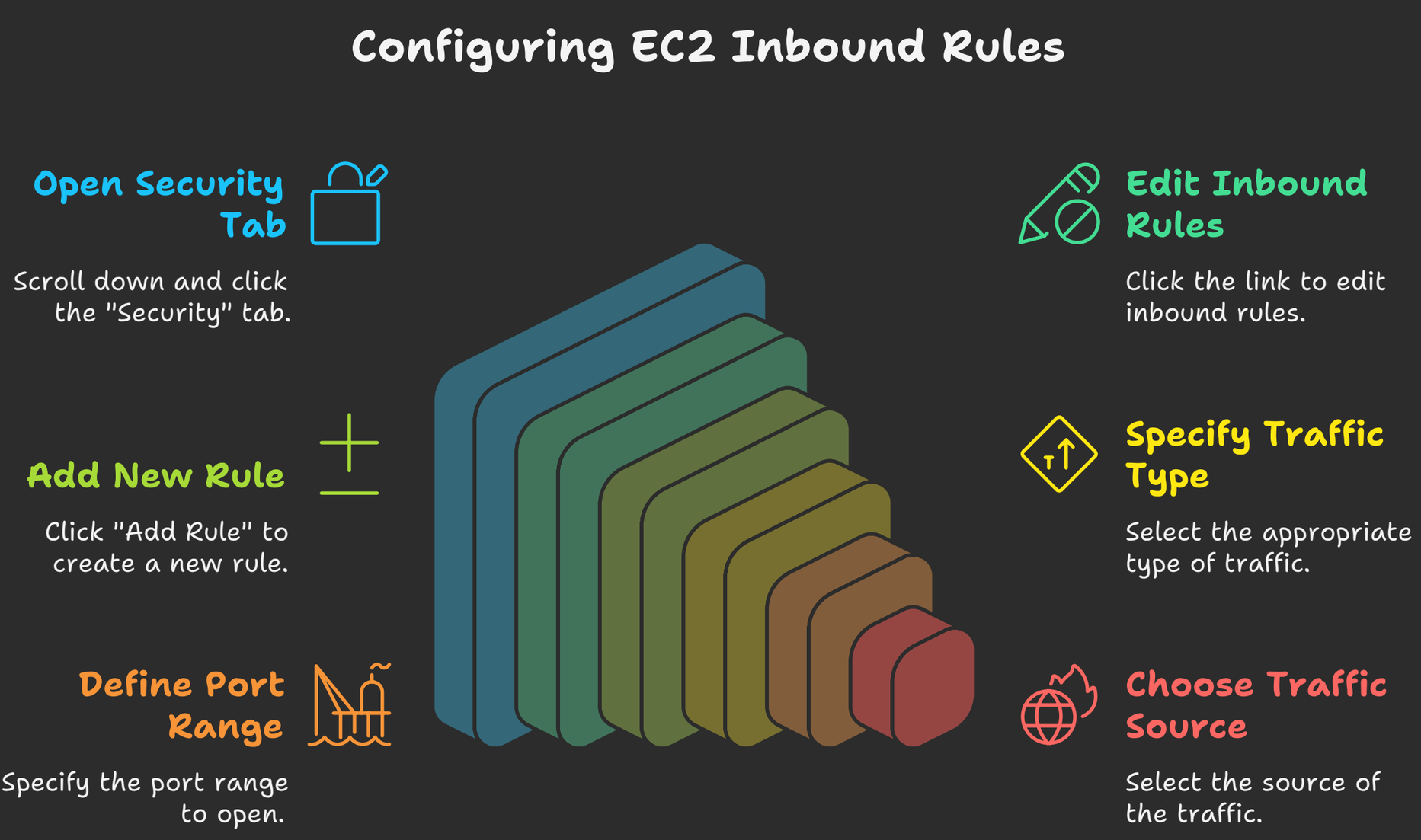
- **Go to Elastic IPs:** From the EC2 dashboard, navigate to "Elastic IPs."
- **Allocate Elastic IP Address:** Click "Allocate Elastic IP address."
- **Press Allocate:** Click the "Allocate" button to generate a new Elastic IP address.
- **Associate Elastic IP Address:** Click on the allocated IP address under the "Allocated IPv4 address" column.
- **Select Associate Elastic IP address:** Choose "Associate Elastic IP address" to link the IP to your instance.
- **Choose Instance:** Select the appropriate instance from the dropdown menu, then click "Associate."



4. Modifying Inbound Rules

Inbound rules control the traffic that is allowed to reach your EC2 instance. You need to configure these rules to open specific ports for SSH, HTTP, or any other services you want to run.

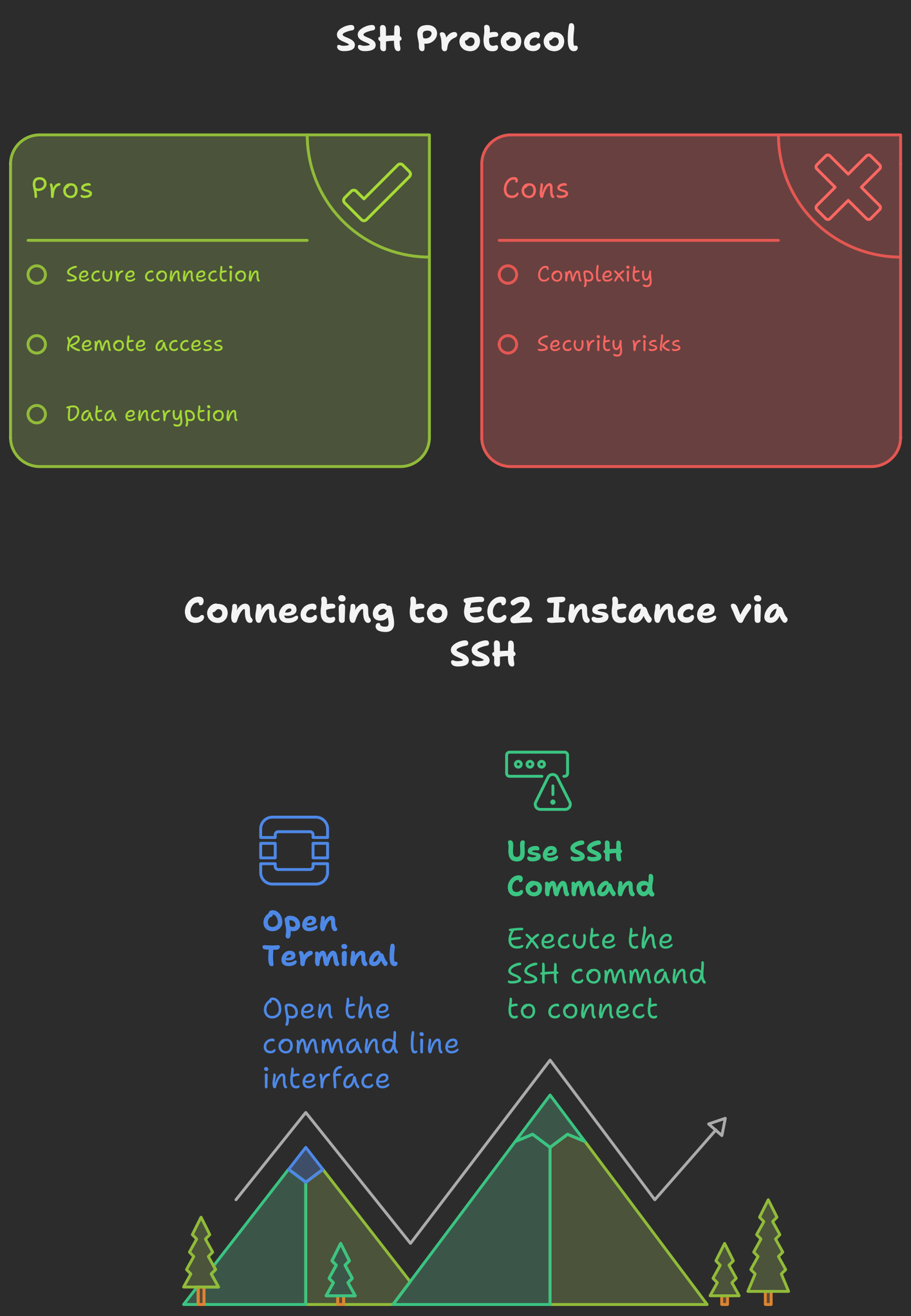
- **Go to Instances:** Navigate to "Instances" and select your instance.
- **Go to Security Tab:** Scroll down and click the "Security" tab.
- **Click Security Groups Link:** Click the link next to "Security groups."
- **Choose Edit Inbound Rules:** Select "Edit inbound rules."
- **Add Rule:** Click "Add Rule" to create a new rule.
    - **Type:** Select the appropriate type of traffic (e.g., "Custom TCP" for specific ports).
    - **Port Range:** Specify the port range you want to open (e.g., 22 for SSH, 80 for HTTP, 3000 for a Node.js application).
    - **Source:** Choose the source of the traffic. "Anywhere – IPv4" allows traffic from any IP address. For increased security, you can restrict the source to specific IP addresses or CIDR blocks.
- **Save Rules:** Click "Save rules" to apply the changes.



5. Connecting via SSH

SSH (Secure Shell) is a protocol used to securely connect to your EC2 instance from your local computer.

- **Open Terminal:** Open CMD (on Windows) or Terminal (on Linux/macOS).
- **Use SSH Command:** Use the following command to connect:




```
ssh -i "path-to-key.pem" ubuntu@YOUR_ELASTIC_IP
```

Example:

```
ssh -i "C:/Users/YourName/Downloads/my-key.pem" ubuntu@13.62.30.17
```

- **Explanation:**
  - ssh: The SSH command.
  - -i "path-to-key.pem": Specifies the path to your .pem key file.
  - ubuntu@YOUR_ELASTIC_IP: Specifies the username (ubuntu for Ubuntu AMIs) and the Elastic IP address of your instance.

- **Ensure Correctness:** Make sure the key path and the Elastic IP are correct.
- **Security Alert:** The first time you connect, you may see a security alert asking if you want to continue connecting. Type yes and press Enter.

Best Practices & Notes

- **Secure Your Key Pair:** Keep your .pem file secure and never share it. If someone gains access to your key pair, they can access your instance.
- **Enable Multi-Factor Authentication (MFA):** Enable MFA for your AWS account for extra security. This adds an extra layer of protection to your account.
- **Monitor Your Usage:** Monitor your AWS usage regularly to avoid unexpected charges. AWS provides tools to track your spending and set up alerts.
- **Stop Instances When Not in Use:** You can stop the instance when not in use to reduce costs. However, make sure to retain the Elastic IP to avoid losing it. If you release the Elastic IP, it may be assigned to another user.