

# AWS VPC CIDR & Subnet Design with Enterprise Route Table Architecture

## Project Overview

This project focuses on designing and deploying a **robust, enterprise-grade AWS VPC** that can support both current workloads and future expansion. The network architecture is built to:

- Use **different subnet sizes** tailored to workload requirements
- Enforce **deliberate and restricted internet connectivity**
- Apply **structured CIDR planning** aligned with industry standards
- Maintain **strong security, scalability, and traceability**

## Importance of This Approach

- Network design decisions are **difficult to change later**
- Poor IP planning leads to **address exhaustion and rework**
- Clearly defined routing improves **security control and visibility**

## Business Use Case

The organization is establishing a **centralized cloud networking layer** to host multiple types of services, including:

- Administrative and operational systems
- Internet-facing web applications
- Backend application services
- Shared internal utilities
- Containerized, high-growth workloads

## Rationale for a Shared VPC Model

- Enables **central policy enforcement**
- Simplifies **logging, auditing, and monitoring**
- Reduces operational overhead and **optimizes costs**

## TASK 1: VPC CIDR & Capacity Planning

1. Choose a private IP range suitable for enterprise workloads.
2. Allocate a CIDR block **not smaller than /16** to allow future growth.
3. Create a VPC with the following CIDR:
  - **VPC CIDR:** 10.0.0.0/16
4. Enable DNS resolution and DNS hostnames for internal communication.

### Result:

A VPC with 65,536 IP addresses, sufficient for current and future subnet expansion.

You successfully created vpc-0b8ff6608453e7172a / devops.vpc

Your VPCs

VPCs | VPC encryption controls

Your VPCs (3) info

Name	VPC ID	State	Encryption c...	Encryption control ...	Block Publicic...	IPv4 CIDR	IPv6 CIDR
terraform-vpc	vpc-01ccdf19852f2e2e7	Available	-	-	Off	10.0.0/16	-
-	vpc-0a87f14f260bbb259	Available	-	-	Off	172.31.0.0/16	-
devops.vpc	vpc-0b8ff6608453e7172a	Available	-	-	Off	10.0.0/16	-

Last updated 2 minutes ago | Actions | Create VPC

You have successfully created 1 subnet: subnet-0d64556390a374d02

Subnets (1) Info

Find subnets by attribute or tag

Subnet ID : subnet-0d64556390a374d02 | Clear filters

Name	Subnet ID	State	VPC	Block Publicic...	IPv4 CIDR	IPv6 CIDR
shared.sh	subnet-0d64556390a374d02	Available	vpc-0b8ff6608453e7172a   dev...	Off	10.0.0.0/19	-

Last updated less than a minute ago | Actions | Create subnet

## STEP 2: Plan Subnet Allocation (Largest First)

1. List subnet requirements in descending order of IP capacity.
2. Start allocating CIDRs from the beginning of the VPC range.
3. Ensure:
  - o No CIDR overlap
  - o Correct network boundaries
  - o All subnets remain within 10.0.0.0/16

## STEP 3: Create Subnets

Create the following six subnets in order:

1. **Shared Subnet**
  - o CIDR: 10.0.0.0/19
  - o Purpose: Large internal services
2. **Platform Subnet**
  - o CIDR: 10.0.32.0/20
  - o Purpose: Containers and internal tools
3. **App Subnet**
  - o CIDR: 10.0.48.0/21
  - o Purpose: Application tier
4. **Web Subnet**
  - o CIDR: 10.0.56.0/22
  - o Purpose: Web tier
5. **Edge Subnet**
  - o CIDR: 10.0.60.0/23
  - o Purpose: Load balancers / ingress
6. **Admin Subnet**
  - o CIDR: 10.0.62.0/24
  - o Purpose: Bastion hosts and operations

### Result:

Six non-overlapping subnets with different capacities, all inside the VPC.

**VPC****VPC ID**

Create subnets in this VPC.

vpc-0b8f6608453e7172a (devops.vpc)

**Associated VPC CIDRs****IPv4 CIDRs**  
10.0.0.0/16**Subnet settings**

Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 1****Subnet name**

Create a tag with a key of 'Name' and a value that you specify.

plantform.sh

The name can be up to 256 characters long.

**Availability Zone** Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

No preference

**IPv4 VPC CIDR block** Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.0.0.0/16

**IPv4 subnet CIDR block**

10.0.32.0/20

4,096 IPs

**IPv4 subnet CIDR block**

10.0.32.0/20

4,096 IPs

▼ **Tags - optional****Key**

Name

**Value - optional**

plantform.sh

X

Remove

Add new tag

You can add 49 more tags.

Remove

Add new subnet

Cancel

Create subnet

**Subnet settings**

Specify the CIDR blocks and Availability Zone for the subnet.

**Subnet 1 of 3****Subnet name**

Create a tag with a key of 'Name' and a value that you specify.

App

The name can be up to 256 characters long.

**Availability Zone** Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.

No preference

**IPv4 VPC CIDR block** Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

10.0.0.0/16

**IPv4 subnet CIDR block**

10.0.48.0/21

2,048 IPs

&lt; &gt; ^ v

▼ **Tags - optional****Key**

Name

**Value - optional**

App

X

Remove

Add new tag

You can add 49 more tags.

Remove

### Subnet 2 of 3

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.



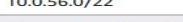
#### IPv4 VPC CIDR block Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.



#### IPv4 subnet CIDR block

1,024 IPs



#### ▼ Tags - optional

##### Key

##### Value - optional



You can add 49 more tags.

### Subnet 3 of 4

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.



#### IPv4 VPC CIDR block Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.



#### IPv4 subnet CIDR block

1,024 IPs



#### ▼ Tags - optional

##### Key

##### Value - optional



You can add 49 more tags.

### Subnet 4 of 5

#### Subnet name

Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

#### Availability Zone Info

Choose the zone in which your subnet will reside, or let Amazon choose one for you.



#### IPv4 VPC CIDR block Info

Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.



#### IPv4 subnet CIDR block

512 IPs



#### ▼ Tags - optional

##### Key

##### Value - optional



You can add 49 more tags.

**Subnet 5 of 5**

**Subnet name**  
Create a tag with a key of 'Name' and a value that you specify.  
  
The name can be up to 256 characters long.

**Availability Zone** [Info](#)  
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

**IPv4 VPC CIDR block** [Info](#)  
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

**IPv4 subnet CIDR block**  
 256 IPs

**Tags - optional**

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="Admin"/> <a href="#">Remove</a>

[Add new tag](#)  
You can add 49 more tags.  
[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

**Subnets (10) [Info](#)**

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 CIDR	IPv6 CIC
public-subnet	subnet-0b05a66742abb6b1	Available	vpc-01cc719852f2e2e7   terra...	Off	10.0.1.0/24	-	-
-	subnet-0894d389b25de35af	Available	vpc-0a87f14f260bbb259	Off	172.31.0.0/20	-	-
-	subnet-034db967b19a14116	Available	vpc-0a87f14f260bbb259	Off	172.31.16.0/20	-	-
-	subnet-0e247592c0f8b6848	Available	vpc-0a87f14f260bbb259	Off	172.31.32.0/20	-	-
shared.sh	subnet-0fd45f6390a374d02	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.0.0/19	-	-
platform.sh	subnet-0bdcfb920f70ea457c4	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.32.0/20	-	-
edge	subnet-04919cbfa042427b4	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.60.0/23	-	-
App	subnet-0fafc9d581b162dc1	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.48.0/21	-	-
web	subnet-0633b400060efdfdad	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.56.0/22	-	-
Admin	subnet-00eaf9c92e7bac97	Available	vpc-0b8f6608453e7172a   dev...	Off	10.0.62.0/24	-	-

## STEP 4: Create and Attach Internet Gateway

Internet gateway creations (dev.igw)

**Internet gateway igw-0d6c4d9c52a149c4c successfully attached to vpc-0b8f6608453e7172a**

**Internet gateways (1/3) [Info](#)**

Name	Internet gateway ID	State
-	igw-05896acd27a0c72c2	Attached
terraform-igw	igw-062a1525f4eea1fcb	Attached
dev.IGW	igw-0d6c4d9c52a149c4c	Attached

### Why IGW is required:

- Enables communication between VPC and the internet
- Does nothing unless routing explicitly allows it
- Central control point for outbound/inbound traffic

**Create internet gateway** Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

**Internet gateway settings**

**Name tag**  
Creates a tag with a key of 'Name' and a value that you specify.

**Tags - optional**  
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="dev.IGW"/> <span style="color: red;">X</span> <span style="color: blue;">Remove</span>

Add new tag

You can add 49 more tags.

Cancel Create

## ➤ Internet gateway Attached to vpc

**Internet gateways (3) Info**

Actions Create internet

<input type="checkbox"/>	Name	Internet gateway ID	State	VPC ID	Owner
<input type="checkbox"/>	-	igw-05896acd27a0c72c2	Attached	vpc-0a87f1f260bbb259	773184566684
<input type="checkbox"/>	terraform-igw	igw-062a1525f4eea1fcf	Attached	vpc-01cc7d719852f2e2e71   terraform-vpc	773184566684
<input type="checkbox"/>	dev.IGW	igw-0d6c4d9c52a149c4c	Attached	vpc-0b8f6608453e7172a   devops.vpc	773184566684

igw-0d6c4d9c52a149c4c / dev.IGW

Details Tags

**Details**

Internet gateway ID <input type="text" value="igw-0d6c4d9c52a149c4c"/>	State <span style="color: green;">Attached</span>	VPC ID <input type="text" value="vpc-0b8f6608453e7172a   devops.vpc"/>	Owner <input type="text" value="773184566684"/>
---	--	---	--

## Why attachment matters:

- Without attachment, public subnets cannot access internet
- Ensures intentional connectivity

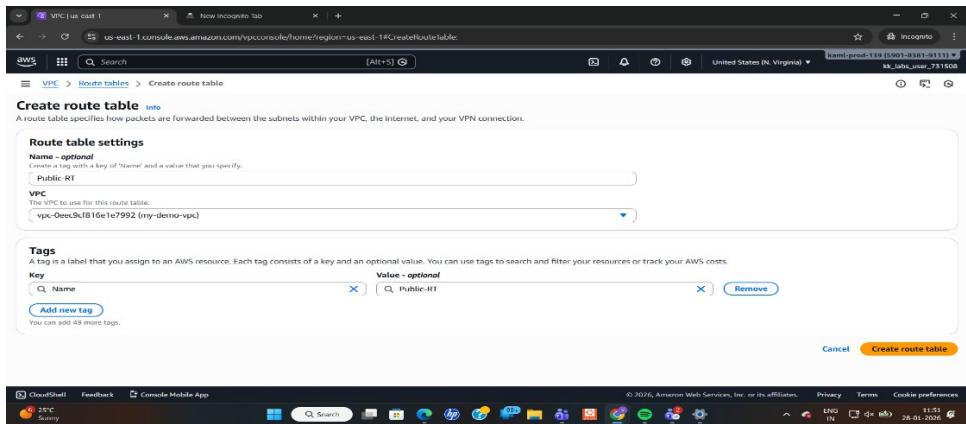
## STEP 5: Create Route Tables

### Step 5.1: Create Public Route Table (Public-RT)

- Create a route table named **Public-RT**.
- Add the following routes:
  - 10.0.0.0/16 → Local
  - 0.0.0.0/0 → Internet Gateway

## Why this is used:

- Allows controlled internet access
- Used only by Admin and Edge subnets



## Step 5.2: Create Private Route Table (Private-RT)

1. Create a route table named **Private-RT**.
2. Ensure only the default route exists:
  - o  $10.0.0.0/16 \rightarrow \text{Local}$

### Why no internet route:

- Prevents accidental internet exposure
- Enforces zero-trust networking

Route table rtb-08bc476852d1a8ee1 | route-pvt.rt was created successfully.

Details		Info	
Route table ID	rtb-08bc476852d1a8ee1	Main	-
VPC	vpc-0b8f6608453e7172a   devops.vpc	Owner ID	773184566684
		Explicit subnet associations	Edge associations
		-	-

**Routes** | Subnet associations | Edge associations | Route propagation | Tags

**Routes (1)**

Q Filter routes		Both			Edit routes	
Destination	Target	Status	Propagated	Route Origin	< 1 >	⋮
10.0.0.0/16	local	Active	No	Create Route Table		

## Route Table Associations

Subnet	Route Table	Why
Admin	Public-RT	Ops access
Edge	Public-RT	Ingress traffic
Web	Private-RT	Internal only
App	Private-RT	Backend isolation
Platform	Private-RT	Secure tooling
Shared	Private-RT	Internal services

Image-12: Route table associations

### Why explicit association:

- Prevents use of default main route table
- Improves audit visibility

## 8. Security-Driven Network Behavior (Task )

Subnet Type	Internet Access	Reason
Admin	Yes	IGW route
Edge	Yes	IGW route
Others	No	No default route

### Why this works:

- AWS local routing enables internal communication
- No security groups needed to prove isolation

## STEP 6: Associate Route Tables with Subnets

1. Explicitly associate **Admin** subnet with **Public-RT**.
2. Explicitly associate **Edge** subnet with **Public-RT**.
3. Explicitly associate the following subnets with **Private-RT**:
  - o Web
  - o App
  - o Platform
  - o Shared
4. Verify no subnet is using the **main route table**.

## Result:

Public and private access is strictly controlled.

The screenshot shows two main sections: 'Subnet associations' and 'Route tables'.

**Subnet associations:**

- Explicit subnet associations (0):** A table with columns: Name, Subnet ID, IPv4 CIDR, and IPv6 CIDR. It displays a message: "No subnet associations. You do not have any subnet associations."
- Subnets without explicit associations (4):** A table with columns: Name, Subnet ID, IPv4 CIDR, and IPv6 CIDR. It lists four subnets: shared.sh, plantform.sh, App, and web.

**Route tables:**

- rtb-08bc476852d1a8ee1 / route-pvt.rt:** A summary card with details like Route table ID, Main status (No), Owner ID, and Edge associations.
- Routes (1):** A table with columns: Destination, Target, Status, Propagated, and Route Origin. It shows one route: 10.0.0.0/16 to local, Active status, No propagated, and Create Route Table as the origin.

## STEP 7: Enforce Security-Driven Network Behavior

1. Confirm only Public-RT has a route to the Internet Gateway.
2. Verify private subnets have no 0.0.0.0/0 route.
3. Ensure all subnets retain the local VPC route.

## Outcome:

- Admin & Edge → Internet access allowed
- Web, App, Platform, Shared → Internet blocked
- Internal communication → Allowed

The screenshot shows three separate route table configurations:

- rtb-08bc476852d1a8ee1 / route-pvt.rt**: A table with one route to 10.0.0.0/16 target local, status Active, propagated No, and origin CreateRouteTable.
- Edit routes (1 route)**: A modal showing the same route configuration with options to Add route, Remove, Preview, or Save changes.
- Edit routes (1 route)**: Another modal showing the same route configuration, but with the target changed to Internet Gateway (igw-0d6c4d9c52a149c4c), status Active, propagated No, and origin CreateRouteTable. It also shows a dropdown menu for selecting the gateway.

## STEP 8: Validate Network Behavior

### Step 8.1: Test Public Subnets

1. Launch an EC2 instance in Admin or Edge subnet.
2. Attempt to access an external website.

#### Expected Result:

Internet access succeeds due to IGW route.

### Step 8.2: Test Private Subnets

1. Launch an EC2 instance in any private subnet.
2. Attempt to access the internet.

#### Expected Result:

Connection fails due to absence of internet route.

### Step 8.3: Test Internal Communication

1. Test connectivity between instances in different subnets.

#### Expected Result:

Communication succeeds via local VPC routing.

## **STEP 9: Failure & Audit Scenarios**

### **Scenario 1: Internet Gateway Detached**

1. Detach IGW from the VPC.

#### **Result:**

- All internet traffic fails
- Internal traffic remains functional

### **Scenario 2: Private Subnet Associated with Public-RT**

1. Associate a private subnet with Public-RT.

#### **Result:**

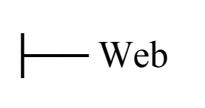
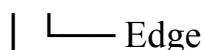
- Subnet becomes internet-accessible
- Violates security and compliance policies

\

## **Architecture Diagram (Logical)**

 **Image-15:** Logical VPC architecture diagram

VPC 10.0.0.0/16



## Conclusion

This VPC design follows **enterprise-grade networking principles**, ensures **strong security boundaries**, and provides a **future-proof IP addressing strategy**. The architecture is scalable, auditable, and suitable for long-term production use.

### Resources

You are using the following Amazon EC2 resources in the United States (Ohio) Region:

Instances (running)	0	Auto Scaling Groups	0
Dedicated Hosts	0	Elastic IPs	0
Key pairs	6	Load balancers	0
Security groups	38	Snapshots	0

### Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

[Launch instance](#)

[Migrate a server](#)

Note: Your instances will launch in the United States (Ohio) Region

### Instance alarms

⚠ 0 in alarm

✓ 0 OK

[View in CloudWatch](#)

0 insufficient data

[Instances in alarm](#)

### Scheduled events

**United States (Ohio)**

No scheduled events



[Mobile App](#)

### Service health

**Region**  
United States (Ohio)

### Zones

**Zone name**

us-east-2a

us-east-2b

us-east-2c

[Enable additional Zones](#)

ⓘ It seems like you may be new to launching instances in EC2. Take a walkthrough to learn about EC2, how to launch instances and about best practices

### Launch an instance Info

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

#### Name and tags Info

Name  
 [Add additional tags](#)

#### Application and OS Images (Amazon Machine Image) Info

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS images

[Recent](#) [Quick Start](#)



[Browse more AMIs](#)  
Including AMIs from AWS, Marketplace and the Community

#### Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type  
ami-06e6845c79fd65d6 (64-bit (x86)) / ami-01da1d3fgea3a6ee6 (64-bit (Arm))  
Virtualization: hvm ENA enabled: true Root device type: ebs

[Free tier eligible](#)

**Description**  
Ubuntu Server 24.04 LTS (HVM),EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).  
Canonical, Ubuntu, 24.04, amd64 noble image

Architecture	AMI ID	Publish Date	Username   ⓘ
64-bit (x86)	ami-06e3c045d79fd65d9	2025-12-12	ubuntu Verified provider

**Instance type** ⓘ | Get advice

**Instance type**

t3.micro	Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand RHEL base pricing: 0.0392 USD per Hour	On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour
On-Demand Windows base pricing: 0.0196 USD per Hour	On-Demand SUSE base pricing: 0.0104 USD per Hour
On-Demand Linux base pricing: 0.0104 USD per Hour	

All generations | Compare instance types

**Additional costs apply for AMIs with pre-installed software**

**Key pair (login)** ⓘ

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

**Key pair name - required**

sandeep	Create new key pair
---------	---------------------

**VPC - required** ⓘ

vpc-0b8f608453e7172a (devops.vpc)  
10.0.0.0/16

**Subnet - required**

subnet-0ccafaf0c02a7bac97	Admin
VPC: vpc-0b8f608453e7172a Owner: 773184566684 Availability Zone: us-east-2b (use2-az2)	Create new subnet
Zone type: Availability Zone IP addresses available: 251 CIDR: 10.0.6.0/24	

**Auto-assign public IP** ⓘ

Enable
--------

**Firewall (security groups)** ⓘ

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group	Select existing security group
-----------------------	--------------------------------

**Security group name - required**

Launch-wizard-36
------------------

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and \_-./@#%^&().

**Description - required** ⓘ

Launch-wizard-36 created 2026-01-28T07:31:09.948Z
---

**Inbound Security Group Rules**

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0)

Type   ⓘ	Protocol   ⓘ	Port range   ⓘ
ssh	TCP	22
Source   ⓘ	Add CIDR, prefix list or security group	Remove
Anywhere	e.g. SSH for admin desktop	
0.0.0.0/0		

**Connect** ⓘ

Connect to an instance using the browser-based client.

EC2 Instance Connect	Session Manager	SSH client	EC2 serial console
----------------------	-----------------	------------	--------------------

**Instance ID**

i-03348593fdf087a5 (devops)
-----------------------------

1. Open an SSH client.  
2. Locate your private key file. The key used to launch this instance is saikiran.pem  
3. Run this command, if necessary, to ensure your key is not publicly viewable:  
chmod 400 "saikiran.pem"  
4. Connect to your instance using its Public IP:  
3.23.114.78

Command copied

ssh -i "saikiran.pem" ubuntu@3.23.114.78

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

```
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1018-aws x86_64)

 * Documentation:   https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Wed Jan 28 09:06:57 UTC 2026

System load: 0.16           Temperature:      -273.1 C
Usage of /: 26.0% of 6.71GB Processes:          117
Memory usage: 24%           Users logged in:    0
Swap usage:  0%             IPv4 address for ens5: 10.0.62.115

Expanded Security Maintenance for Applications is not enabled.
0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-62-115:~$
```