

AWS in DevOps

Page No.	
Date	

* Cloud Computing : It provides on-demand virtual infrastructure
it is pay per use

* Vendors :

- AWS → 33%
- Azure → 22%
- Google Cloud → 8%

* AWS (Amazon Web Services)

Resources :

EC2 instance EC2 stand elastic cloud compute

S3 (simple storage service)

VPC (Virtual private cloud)

load balancer

Elastic Beanstalk

Lambda

Database (RDS, Dynamodly) etc

* Servers in data centre can be accessed remotely

In Windows server we use → RDP (Remote Desktop protocol)

In Linux Server we use → SSH (Secured shell)

* For data transfer :

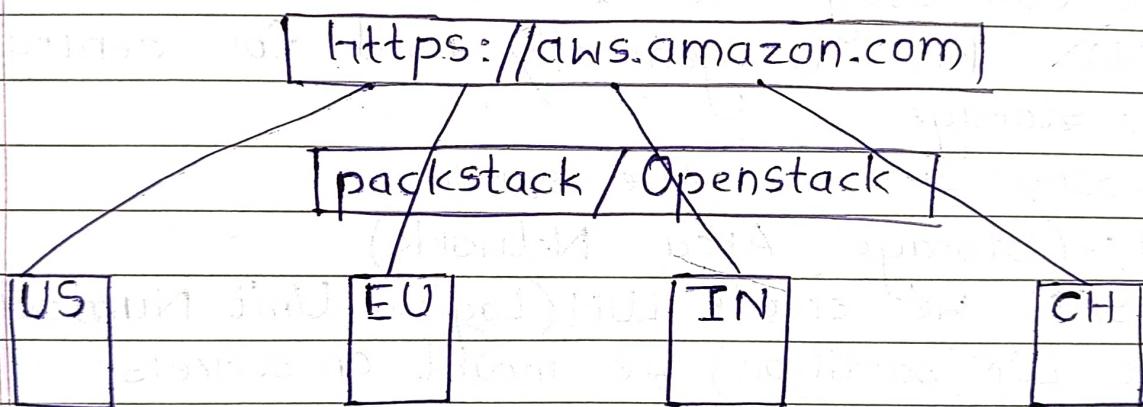
FTP (File Transfer Protocol) : using FTP we can transfer bulk data over the internet.

* Cloud Computing :

- In backend they have data centers
- e.g. AWS have data center in US, EU, India, China.

- b. They have integration this data center to this url <https://aws.amazon.com>
- c. They have integrated with help of software packstack/openstack these software through which we can integrate data centers using URL.
- d. packstack is a licensed by redhat protocol.
- e. Openstack is a open source.

Note: If you lean open stack it means you are able to create your own cloud environment



* Cloud Computing types

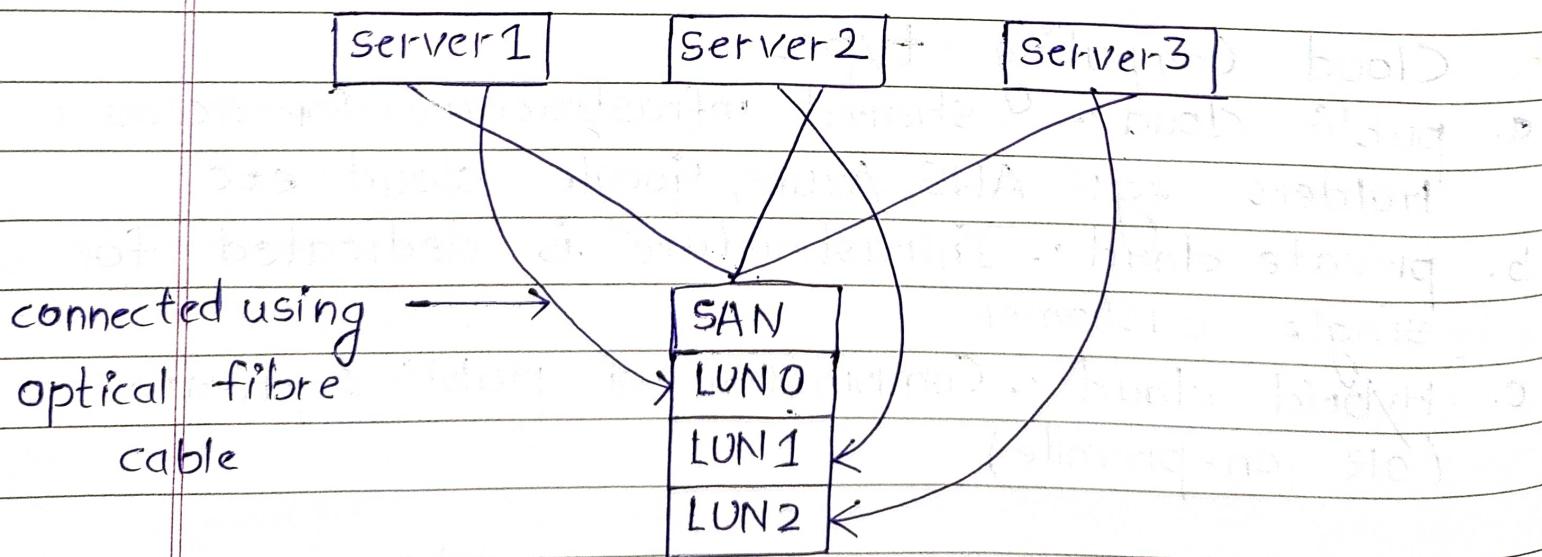
- a. public cloud :- shared infrastructure for account holders eg: AWS, Azure, Google cloud etc
- b. private cloud :- Infrastructure is dedicated for single customer
- c. Hybrid cloud :- Combination of public & private (or on-premize)

* Cloud Service Models

- a. IaaS (Infrastructures as a Service)
- b. PaaS (Platform as a Service)
- c. SaaS (Software as a Service)

* Storage :- Types

- a. DAS : (Directly Attached Storage)
⇒ Harddisk connected to system.
- b. NAS : (Network Attached Storage)
⇒ NAS box will connect to network
We can assign IP Address to NAS box.
⇒ NAS box is generally used for centralized data storage.
⇒ mostly act as file servers.
- c. SAN : (Storage Area Network)
In SAN we create LUN (Logical Unit Number) & those LUN (partition) we mount on servers



* Cloud Service Models

Traditional

IaaS

PaaS

SaaS

	Applications	Applications	Application	Application
you manage	Data	Data	Data	Data
you manage	Runtime	Runtime	Runtime	Runtime
you manage	Middleware	Middleware	Middleware	Middleware
you manage	O/S	O/S	O/S	O/S
you manage	Virtualization	Virtualization	Virtualization	Virtualization
you manage	Servers	Servers	Servers	Servers
you manage	Storage	Storage	Storage	Storage
you manage	Networking	Networking	Networking	Networking

* How to deploy EC2 instance on cloud?

If you have already account in AWS

⇒ aws.amazon.com → Sign in → Authentication code → Submit → EC2 [click] → Instances → Launch instances → Name : linuxec2 → Create new key pair → Key pair name: linuxec2 → Create key pair → Allow HTTP traffic from the internet → Launch instance → Instances → Q → i-079b4c416f10b078d → Connect → ssh [copy] →

Open command prompt:

> cd Downloads

> paste

yes

This is the virtual server we have created on AWS

This is example of IaaS

We create O/S

\$ sudo su -

yum install httpd [it is web server package
y it will install apache & create a data]

```
# cd /var/www/html  
# ls  
# echo "India is Great" > index.html  
# systemctl start httpd  
# systemctl enable httpd  
# curl localhost  
→  
>i-079b4c416f10b078d → public IPv4 address [Copy]  
→ New tab [Paste]
```

* How to create Linux EC2 instance?

EC2 stands for Elastic cloud compute

- a. Linux EC2 instances
- b. Will attach a persistence disk to EC2 instance.
- c. Will create two partition in the given disk
- d. Install httpd & configure the server as web server.

Steps:

aws.amazon.com free → sign in → Authentication → EC2 instances → N-Virginia [Select] → Launch instances → Name: linux1 → [Select] Amazon Linux → t2.micro by default [Select] → Create new key pair → Key pair name: Linux1 → Create key pair → Configure storage: Add new volume 1 x 20 → 4GB Magnetic Standard [Select] → Launch instance → Instances → Refresh

* Firmware : It's a small software stored in ROM (Read Only Memory)

- Works betn Computer hardware & operating system.

Types of firmware:

- a. BIOS :- Basic Input Output System
 - Introduced in 1986
 - Support for MBR (Master Boot Record) partitioning type
 - can support disk size of 2TB only.
- b. UEFI :- Unified Extended System Interface
 - Latest firmware
 - support for MBR as well as GPT (GUID partitioning table) partitioning scheme.
 - can support disk size of 9.4 million TB.

* MBR :- Master Boot Record

- command used in fdisk

* GPT :- GUID Partition Table

- command used in gdisk

Linux1 → Instance ID [Click] → Connect → SSH client →

Copy Example: ssh -i "...".pem ec2-18-215-129-115.us-east-2.compute.amazonaws.com

Command Prompt

> cd Downloads

> paste —

\$ sudo su -

[Provide admin privilege]

\$ yum install httpd [will install httpd package]

y

#cd /var/www/html [change directory]

echo "Welcome to IEC" > index.html

[create web page]

#systemctl start httpd } [Start & enable httpd service]

#systemctl enable httpd }

#curl localhost [will open web page]

Google chrome

free-css.com/free-css-templates → Download any template → ↴ → Download history → Copy link address

Command prompt

#Wget paste → Notepad → then type command

#ls

#unzip healet.zip

#ls

#cd Healet-HTML/

#ls

#cp -a * .. → Enter

#ls

#cd ..

#ls

#rm -rf Healet-HTML

#rm -rf Healet.zip

#ls

#systemctl start httpd

#systemctl enable httpd

#curl localhost

linux1 → Instance ID [click] → Public IPv4 address [copy]

→ Security → Security group [Click Link] →

Edit inbound rules → Add rule → HTTP [Select] →

0.0.0.0/0 → Save rules

* MBR partitioning

Steps:

1. Attach extra disk (persistant)
2. fdisk -l [show details of disks attached and partition with details]
 fdisk /dev/xvdb [create two partitions 10GB each]
 m [list menu options]
 n [create new partitions or volumes]
 p
 1
 2048
 +10G
 n
 p
 2
 Enter
 Enter
 w

fdisk -l → it will show 2 partition

3. Create file system/format the volume/partition

```
#mkfs -t xfs /dev/xvdb1  

#mkfs -t xfs /dev/xvdb2
```

4. Create two folder for volume mounting

```
#mkdir /media/part1 /media/part2
```

5. Mount it

```
Edit file # nano /etc/fstab
```

```
/dev/xvdb1 /media/part1 xfs defaults 0 0
/dev/xvdb2 /media/part2 xfs defaults 0 0
```

Ctrl + O = Save

Enter

Ctrl + X = Exit

#cat /etc/fstab [check/view file]

6. #mount -a [Read /etc/fstab file & mounts accordingly]

7. #mount [shows mount point]

#cd /media/part1

#ls

#touch t1

#ls

* GPT partitioning

Steps

1. Attach one more disk

eg. /dev/xvdc

#fdisk -l [list disk & file systems in it]

2. #gdisk /dev/xvdc

3. #mkfs -t /dev/xvdc1

How to attach disk

linux1 → Instance ID → Elastic Block Store → Volumes

→ Create volumes → Volume type: Magnetic (standard)

[Select] → Size: 20 → Create volume → Actions:

Attach volumes → [linux1 Select] → [Select volume]

/dev/sdc [select] → Attach volume

Command Prompt

```
fdisk -l
```

```
# gdisk /dev/xvdc  
:n
```

Enter

Enter

+10G

Enter

n

Enter

Enter

Enter

w

y

```
# fdisk -l
```

```
# mkdir /media/part3 /media/part4
```

```
# vi /etc/fstab
```

Esc → :wq! + Enter = Edit

```
# nano /etc/fstab
```

/dev/xvdc1 /media/part3 xfs defaults 0 0

/dev/xvdc2 /media/part4 xfs defaults 0 0

ctrl + O = Save

Enter

Ctrl + X = Exit

```
# cat /etc/fstab
```

```
# mount -a
```

```
# mkfs -t xfs /dev/xvdc1
```

```
# mkfs -t xfs /dev/xvdc2
```

```
# mount -d
```

```
# mount -t xfs /dev/xvdc1 /mnt
```

- * VPC: Virtual Private cloud
 - your private network on AWS cloud.
 - By default 1 vpc per region is already available (default)
 - And within default, no. of subnets also available (one subnet per availability zone is available)

aws.amazon.com free → sign in → VPC [Search] → vpc →
 VPC Create VPC → Name tag: VPC01 → IPv4 CIDR block:
 10.0.0.0/16 → Create VPC

Subnet Subnets → Create subnet → VPC ID: VPC02 [Select] →
 Subnet name: pubsubnet → IPv4 subnet CIDR block:
 10.0.0.0/24 → Create Subnet
 Create subnet → VPC02 [Select] → Subnet Name: pvtsub
 - net → IPv4 subnet CIDR block: 10.0.1.0/24 →
 Create subnet → Subnet ID: click

Route tables By default 2 route-tables available → Rename →
 PubRT & PvtRT → PubRT Route table ID [click] →
 Subnet associations → Edit Subnet associations →
 PubSubnet → Save associations → Route tables →
 Create route table → Name: PvtRT → VPC02 [Select] →
 Create route table → Subnet associations → Edit subnet
 associations → PvtSub → Save associations

Internet gateway Create internet gateway → Name: IGW01 →
 Create internet gateway → Actions: [Attach to VPC]
 → VPC02 [Select] → Attach internet gateway.
 Route table → PubRT → PubRT Route table ID
 [click] → Edit routes → Add routes → Destination:
 0.0.0.0/0 → ▽ → Internet Gateway [Select] →
 igw- [click] → Select long → Save changes

Create public & private ec2 instance
EC2 [Search] → Launch instance → name: pubec2 →
aws [Select] → Create new key pair [Select] → Network settings → Key pair
name: myec3 → Create key pair → Edit → VPC02
[Select] → Subnet: Pubsubnet [Select] → Auto-assign
public IP: Enable → Launch instance →
instances → Launch instance → name: pvtc2 →
aws [Select] → Create new key pair [Select] → myec3 →
Edit → VPC02 [Select] → Subnet: Pvtsubnet [Select] →
Select existing security group → Common security
groups: launch-wizard-3 → Launch instances →
instances → pubec2 → ID → Connect → ssh [copy]

Command prompt

> cd Downloads

> paste _____

yes

\$ ping google.com

Instances → pvtc2 → ID → Connect → ssh [copy]

Command prompt pvt

> cd Downloads

> paste _____

yes [No work]

> dir [it will show myec3]

> \$ ls

\$ vi myec3.pem

This pc → Downloads → ~~move~~ myec3.pem → More
apps [Notepad] Select → Copy paste Command
prompt → to edit click (i) → Esc → :wq! -

```
$ ls  
$ pvtec2 ssh paste  
$ ls  
$ ls -l  
$ chmod 400 myec3.pem  
$ ls  
$ ls -l  
$ pvtec2 ssh paste [worked]  
$ ping 10.0.0.135 [no work]  
$ ping 10.0.1.61 [no work]
```

instance → pvtec2 → Security → Security group
[Click] link → Edit inbound rules → Add rule →
All ICMP-IPv4 → 10.0.0.0/16 → Save rules

Command Prompt

```
$ ping 10.0.1.61 [worked]  
$ pvtec2 ssh paste [worked]  
$ ping 10.0.0.135 [worked]  
$ ping google.com [no worked]
```

NAT gateway VPC [Search] → VPCs → VPC02 ID → Nat Gateway →
Create Nat gateway → Name: NAT01 → pubsubnet →
Allocate Elastic IP [click] →
Create Nat gateway → Route table → Pvt RT ID →
Edit routes → Add route → 0.0.0.0/0 → NAT Gateway
[Select] → Save changes →

Command prompt

```
$ ping google.com
```

How to destroy infrastructure

ec2 → instances → pvtEc2 pubEc2 → Instance state : Terminate (delete) instance → Terminate instance

VPC → Route table → pvtRT ID → Edit routes → Remove → Save changes → subnet association → Selected subnets → Save association → route table → pubRT ID → Edit routes → Remove → Save changes → subnet association → Selected subnets → Save association

NAT Gateway → NAT01 → Action : Delete NAT Gate-way → delete → Delete →

Internet Gateway → IGW01 → Action : Detach from VPC → Delete Internet Gateway →

Elastic IP's → → Action : Release Elastic IP address → Release all →

Internet Gateway → IGW01 → Action : Detach from VPC → Delete Internet Gateway →

IGW01 → Delete Internet Gateway → delete → Delete

VPC01 → Action : Delete VPC → delete → Delete.

* S3 :- Simple Storage Service S3

- Its a object storage
 - you can store any type of data here
 - object storage is called unstructured data storage
 - we can store unlimited data.
 - maximum file size can be STB (Single file)
- To store data we need to create bucket name should be unique in entire AWS environment.
- Bucket name :- Name what you want + Random string

* How to delete bucket :

S3 [Search] → Empty ~~bucket~~ permanently delete →
 Empty → Exit → ~~payalsargar93~~ → Delete →
 Bucket name [paste] → Delete bucket

How to create bucket :

Create bucket : → Bucket name : payalsargar93 →
 • ACLs enabled → → Block off public access →
 Create buckets → payalsargar93 [link click] → Upload →
 Add files → upload any file → Upload → Uploaded file
 [Click] link → Object URL [Copy] → Paste New tab →
 No access

payalsargar93 → file → Action : Make public using
 ACL → Make public → Close
 Refresh page ←
 [accessible]

properties → Bucket Versioning → Edit → Bucket versioning: Enable → Save changes → Up

Create-text folder [Desktop] → Save → Upload → Add files → Select this file → Upload → Click document link → Object URL [Click] link → Close → Uploaded file → Actions: Make public using ACL → Make public → Refresh page → Accessable

Edit file → Save → Upload → Add files → Select file → Upload → Close → latest file shown → Show versions [on] → new file → click link → Object URL [copy] → paste newtab → No access → Object actions: Make public using ACL → Make public → Refresh page → Accessable

payalsargar93 → Select file → Delete → delete → Delete objects → close

- * Route 53 :- Its a DNS service provided by AWS
- * S3 :- Is port number of DNS
- * DNS :- Domain Name System
DNS resolve Host name (FQDN) to IP address & vice versa
eg. FQDN :- www.google.com
- * Route 53 : Store information in the forms of record
 - a. A Record: Host record, resolve hostname to IP address
 - b. PTR Record: Pointer Record, Resolve IP address to hostname

c. C Name Record: Canonical Name Record. We can provide alternate to FQDN.

e.g. your domain name :- a.google.com want access same application using portal.google.com

d. NS Record: Name Server Record. List no. of DNS (Name) servers in your network.

e. SOA Record: Start of Authority record. This record gets automatically created during creation of zone. It contains information about DNS server & several timer information.

* Route 53:-

a. Private Hosted zone: If we want to access ec2 instances with its name instead of IP address within VPC.

Requirement: domain name (iec.local)

Note: We can give any name. This is not accessible outside VPC

b. Public Hosted dns: We can access our application using publicly accessible domain name instead of IP address

Requirement: Real domain name (we need to purchase)

Create single VPC Here

- VPC [Search] → VPCs → Create VPC → Name tag

VPC name: VPC01 → IPv4 CIDR → 10.0.0.0/16 →

Create VPC →

- Subnet name: PublicSubnet → CIDR block: 10.0.0.0/24

→ Create subnet → Subnet name: PrivateSubnet →

IPv4 subnet CIDR block: 10.0.1.0/24 → Create subnet

Route tables → Public RT [Name] 2nd no. → Save → Click link in PublicRT → Edit subnet associations → Public Subnet → Save association → Route table → Create route table →

Name: PrivateRT → VPC: VPC01 [Select] → Create route table → Subnet associate → Edit subnet associations → PrivateSubnet → Save associations

Internet Gateways → Create internet gateway → Name: INTGW → Create internet gateway → Actions: Attach to VPC → VPC01 [Select] → Attach internet gateway →

Route tables → PublicRT → Link → Edit routes → Add route → 0.0.0.0/0 → Internet Gateway → igw-0. → Remove Save changes.

EC2 → Launch instances → publicec2 → Create new key pair → ec2my → Create key pair → Networks setting → Edit → [Select] VPC01 → Public Subnet [Select] → Enable: Auto public Assign IP → Launch instance → Launch instance → publicec2 → Create new key pair [Select] ec2my → ⚡ Select existing security group → Network setting → Edit → [Select] VPC01-Private Subnet [Select] → Disable → Common security group info [Select] → Launch Wizard -2 → Launch instance → instance → Refresh → publicec2 [click] link → Connect → SSH client → [Copy] ssh

Command prompt

> cd Downloads

> paste

yes

\$ ping google.com

\$ vi ec2my.pem

This PC → Downloads → ec2my.pem → Open With → Notepad → Copy [data] →

Command prompt

Click ↵ → paste → Esc → :wq!

\$ ls -l

\$ chmod 400 ec2my.pem

\$ ls -l

instances → private ec2 [link] → Connect → Copy [ssh]

Command Prompt

\$ paste

yes

\$ ping 10.0.0.15

↑ Start ping

instances → Security → Security groups: Link → Edit inbound rules → Add rule → All ICMP-IPv4 → Q 0.0.0.0/0 → Save rules]

Search route53 → Create hosted zone → google.com
My domain name → Private hosted zone →
Choose region: US East (N.Virginia) → VPC ID: VPC01
→ Create hosted zone → Create record → Record name: publicec2 → Record type: A-Routes traffic
→ Value: 10.0.0.15 → Create records → Create record
→ Record name: privateec2 → value: 10.0.1.195 →
Create records VPC → VPC01 [link] Click →
Actions → Edit VPC settings → Enable DNS
hostnames → Save →

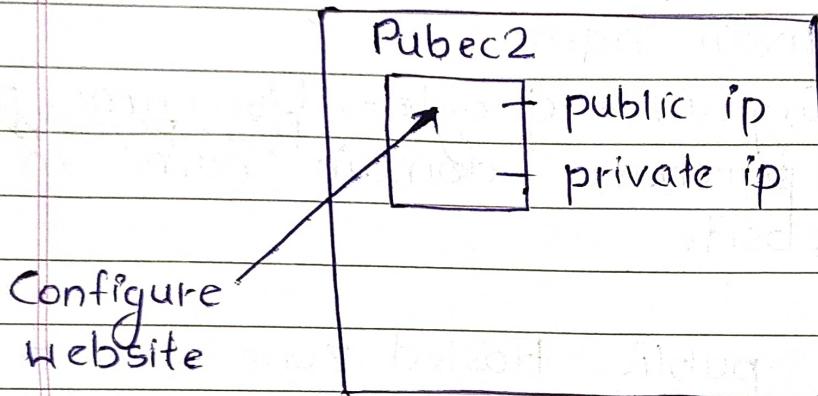
Command prompt

\$ ping 10.0.0.15

\$ ping pubec2.google.com

- * Public Hosted zone: We can access our application Web server using FQDN (Real) instead of IP address. VPC01

Public Subnet



Command prompt

\$ 10.0.1.195 link paste

\$ ping pubec2.google.com [able]

\$ exit

\$ sudo dnf install httpd

\$ y

\$ sudo su -

\$ cd /var/www/html

echo "welcome to Devops class" > index.html

systemctl start httpd

systemctl enable httpd

curl localhost

Security → Security groups → launch-wizard link
 [click] → Edit inbounded rules → Add rules →
 HTTP → Q: 0.0.0.0/0 → Save rules

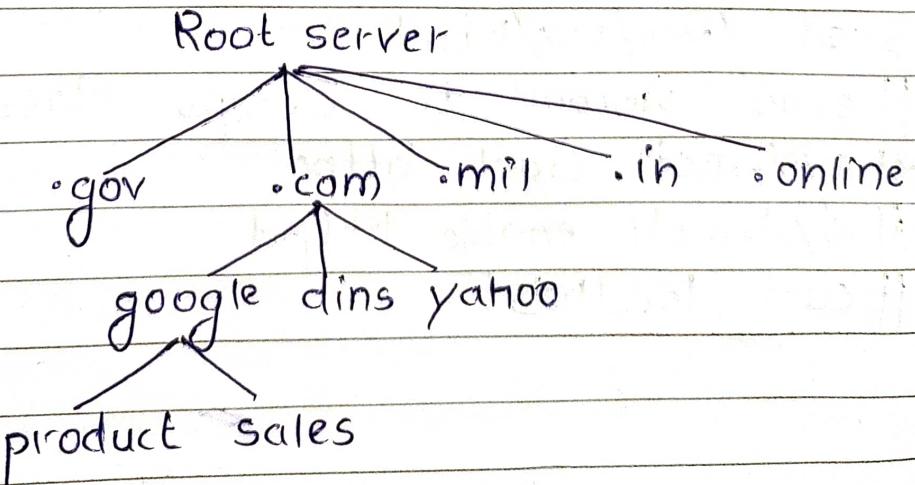
EC2 → instances → publicec2 [link] click →
 Public IPv4 address → Paste New tab

* Need public domain name:
 godaddy.com → Sign in → Google → Username: ^{P/W: Payalsagar} Payal
 you need to purchase domain name in
 godaddy.com website

1. On AWS Created public hosted zone with domain payalsagar ...

Route 53 → Hostedzone → Create hosted name →
 name: → Create hosted zone

2. Update godaddy name server list with NS Server available an public hosted zone you created



Route (2) → Value / Route traffic to → ns-54.ansdns
 [copy] 1st no. →

godaddy.com

→ Name servers → Change Name Servers → Select delete all
 → paste all together → 2nd no copy paste →
 3rd no copy paste → Add nameservers → 4th no.
 copy paste → Save → Continue

3. Create A record in public Hosted zone in Route53
 Create record → name: portal → Value: publsec2 →
 ipv4 address paste → Create records

domain name access hogya → use copy & paste in
 new tab

TTL: Time to Live

Command Prompt

> ipconfig /flushdns

> copy → paste new tab

> nslookup → paste nahi hua to →

> nslookup → paste

→ paste [New tab]

Delete zone → delete → delete

① google.com → delete → delete → delete
 [do not allow to delete]

② google.com [link] click → publsec2 pvtsec2 →
 Delete records → delete → delete zone → delete →

delete → Dashboard

VPC → Route tables → Public RT [link] → Edit rules → Remove → Save change → Subnet association → Edit subnet association → → ~~Subnet association~~ → → Route tables → Private RT [link] → Subnet association → Edit subnet association → → Save association

EC2 → Instances → Name → Instance state: Terminate [delete] instance → Terminate instance

VPC → Internet Gateway → INTGWM [click] link → Manage tags → Remove → Save → Action: Detach from VPC → Detach Internet Gateway → VPC → VPC01 → Action: Delete VPC → delete → delete



EC2 pricing options

1. On-demand: default options
2. Reserve Instance: - We need reserver capacity of EC2 instance for 1 or 3 year
 - Need to pay up-front cost. And can get discount upto 70%
 - a. Standard Reserver Instance
 - b. Convertable Reserver Instance
 - c. Schedule Reserver Instance
3. Spot instance: AWS want sell unused infrastructure at huge discount upto 90%
4. Dedicated Host:

10.0.0.0/24

10.0.0.0 :- network

10.0.0.1 :- default gateway : network exit

10.0.0.2

10.0.0.3

10.0.0.255 :- broadcast

10.0.0.0/28

$$32 - 28 = 4 \quad 2^4 = 16$$

$$16 - 5 = 11 \quad (\text{usable})$$

Console Home Info

[Reset to default layout](#)[+ Add widgets](#)

Recently visited Info



S3



EC2



IAM

Welcome to AWS



Getting started with AWS

Learn the fundamentals and find valuable information to get the most out of AWS.



Training and certification

Learn from AWS experts and advance your skills.

AWS Health Info

Open issues

0

Past 7 days

Scheduled changes

0

Upcoming and past 7 days

Other notifications

0

Past 7 days

Activate Windows
Go to Settings to activate Windows



Type here to search



Very humid



10:25 AM
07/07/2024

EC2 Dashboard

EC2 Global View

Events

Console-to-Code [Preview](#)

Instances

- Instances
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts
- Capacity Reservations [New](#)

Images

<https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Instances>

Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Instances (running)	4	Auto Scaling Groups	0	Capacity Reservations	0
Dedicated Hosts	0	Elastic IPs	1	Instances	4
Key pairs	3	Load balancers	0	Placement groups	0
Security groups	5	Snapshots	0	Volumes	5

Launch instance

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

[Launch instance](#) ▾ [Migrate a server](#)

Service health

[AWS Health Dashboard](#) [C](#)

Region

Activate Windows
Go to Settings to activate Windows

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie policy](#)

28°C Light rain 10:26 9/26

EC2 Dashboard X

EC2 Global View

Events

Console-to-Code [Preview](#)

▼ Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity

Reservations [New](#)

▼ Images

CloudShell Feedback

Type here to search

Last updated less than a minute ago

C Connect Instance state ▾ A

Find Instance by attribute or tag (case-sensitive)

Name	Instance ID	Instance state
linux	i-043be5196791fe009	Running
publicec2	i-01b322ed3f1e6360d	Running
privateec2	i-05b5a40a8590ae9ac	Running
linuxec2	i-079b4c416f10b078d	Running

Select an instance

US East (N. Virginia) us-east-1

US East (Ohio) us-east-2

US West (N. California) us-west-1

US West (Oregon) us-west-2

Asia Pacific (Mumbai) ap-south-1

Asia Pacific (Osaka) ap-northeast-3

Asia Pacific (Seoul) ap-northeast-2

Asia Pacific (Singapore) ap-southeast-1

Asia Pacific (Sydney) ap-southeast-2

Asia Pacific (Tokyo) ap-northeast-1

Canada (Central) ca-central-1

Activate Windows Go to Settings to activate Windows.

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

28°C Light rain 10:27 AM 9/26/2024

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

linux1

Add additional tags

▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.



Type here to search



▼ Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

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

Recents

Quick Start



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 AMI

ami-0ebfd941bbafe70c6 (64-bit (x86), uefi-preferred) / ami-00e73ddc3a6fc7dfe (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

 CloudShell  Feedback

© 2024, Amazon Web Services, Inc. or its affiliates

 Type here to search     

28

Gmail YouTube Maps Convert Next

aws Services Search [Alt+S] N. Virginia

Instance type info | Get advice

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

▼ Key pair (login) Info

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*

Select Create new key pair

Activate V Go to Settings

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy

Type here to search

▼ Key pair (login) [Info](#)

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*

Select



[Create new key pair](#)

▼ Network settings [Info](#)

[Edit](#)

Network [Info](#)

vpc-0de7b231c5f08a4eb

Subnet [Info](#)

No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)

Activate W
Go to Settings

All [Feedback](#)

© 2024, Amazon Web Services, Inc. or its affiliates.

Privacy

Type here to search



28°C Light rain

Create key pair

X

Key pair name

Key pairs allow you to connect to your instance securely.

linux1

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

RSA

RSA encrypted private and public key pair

ED25519

ED25519 encrypted private and public key pair

Private key file format

.pem

For use with OpenSSH

.ppk

Cancel

Create key pair



 **Info** **IP access** With source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

▼ **Configure storage** Info

[Advanced](#)

1x GiB ▾ Root volume (Not encrypted)



1x GiB ▾ EBS volume (Not encrypted)

[Remove](#)

 Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage



[Add new volume](#)

Activate VPC
CloudFront

X

view

Instances (1/5) [Info](#)

Last updated 8 minutes ago

C Connect Instance state Actions ▾ La

Find Instance by attribute or tag (case-sensitive) All states ▾

Name	Instance ID	Instance state	Instance type	Status
<input checked="" type="checkbox"/> linux	i-043be5196791fe009	<input checked="" type="radio"/> Running	t2.micro	<input checked="" type="radio"/> 2/2 ch
<input type="checkbox"/> publicec2	i-01b322ed3f1e6360d	<input checked="" type="radio"/> Running	t2.micro	<input checked="" type="radio"/> 2/2 ch
<input type="checkbox"/> privateec2	i-05b5a40a8590ae9ac	<input checked="" type="radio"/> Running	t2.micro	<input checked="" type="radio"/> 2/2 ch
<input type="checkbox"/> linux1	i-0873c0076207c017c	<input checked="" type="radio"/> Running	t2.micro	<input type="radio"/> Initializ
<input type="checkbox"/> linuxec2	i-079b4c416f10b078d	<input checked="" type="radio"/> Running	t2.micro	<input checked="" type="radio"/> 2/2 che

i-043be5196791fe009 (linux)

Details Status and alarms Monitoring Security Networking Storage Tags

▼ Instance summary [Info](#)

Activate
Go to Summary

EC2 > Instances > i-0873c0076207c017c

Instance summary for i-0873c0076207c017c (linux1) [Info](#)



[Connect](#)

Instance state ▾

Actions ▾

Updated less than a minute ago

Instance ID

[i-0873c0076207c017c \(linux1\)](#)

Public IPv4 address

[3.86.113.147 | open address ↗](#)

Private IPv4 addresses

[172.31.90.121](#)

IPv6 address

-

Instance state

[Running](#)

Public IPv4 DNS



ec2-3-86-113-147.compute-1.amazonaws.com

[| open address ↗](#)

Hostname type

IP name: ip-172-31-90-121.ec2.internal

Private IP DNS name (IPv4 only)

[ip-172-31-90-121.ec2.internal](#)

Answer private resource DNS name

IPv4 (A)

Instance type

t2.micro

Elastic IP addresses

[View details](#)

[View details](#)

[Activate Windows](#)

Instance ID

 [i-0873c0076207c017c \(linux1\)](#)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is `linux1.pem`.
3. Run this command, if necessary, to ensure your key is not publicly viewable.
 `chmod 400 "linux1.pem"`

4. Connect to your instance using its Public DNS:

 [ec2-3-86-113-147.compute-1.amazonaws.com](#)

 Command copied

 `ssh -i "linux1.pem" ec2-user@ec2-3-86-113-147.compute-1.amazonaws.com`

 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.

Canc

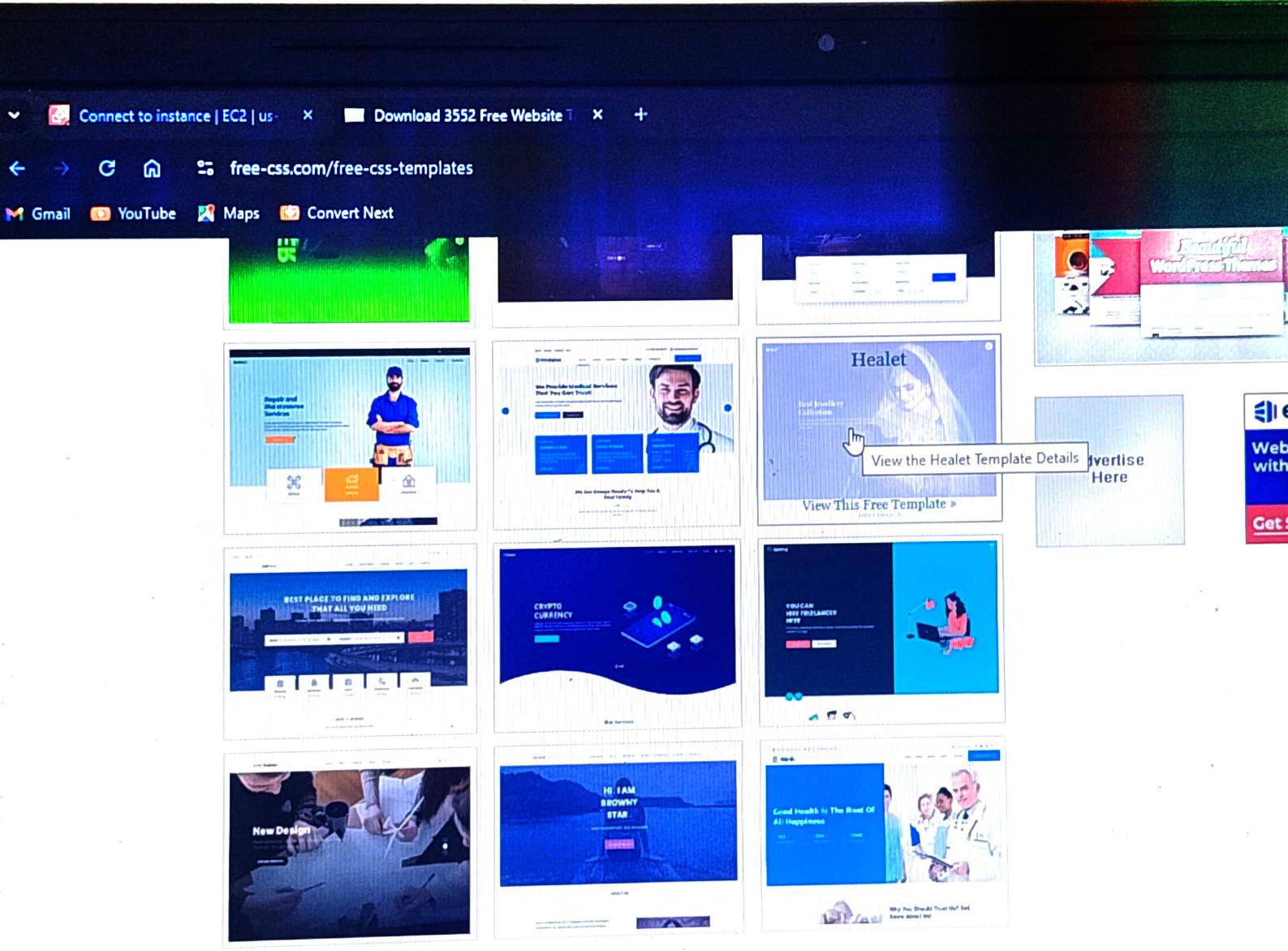
```
Connect to instance | EC2 | us- +  
ec2-user@ip-172-31-90-121:~  
Microsoft Windows [Version 10.0.19045.4894]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\admin>cd Downloads  
C:\Users\admin\Downloads>ssh -i "linux1.pem" ec2-user@ec2-3-86-113-147.compute-1.amazonaws.com  
The authenticity of host 'ec2-3-86-113-147.compute-1.amazonaws.com (3.86.113.147)' can't be established.  
ECDSA key fingerprint is SHA256:CYE1KAGuCA7oK96hLl54gYen+zllUE6d+anPAqgtLZ0.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added 'ec2-3-86-113-147.compute-1.amazonaws.com,3.86.113.147' (ECDSA) to the list of known hosts.  
, #  
~~ \_ #####_ Amazon Linux 2023  
~~ \_ #####\|  
~~ \###|  
~~ \#/ \|--> https://aws.amazon.com/linux/amazon-linux-2023  
~~ V~' .- /  
~~ .- / /  
~~ / / /  
~/m/ /  
ec2-user@ip-172-31-90-121 ~]$
```

```
root@ip-172-31-90-121:/var/www/html
Verifying : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
Verifying : apr-util-1.6.3-1.amzn2023.0.1.x86_64 2/12
Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 3/12
Verifying : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch 4/12
Verifying : httpd-2.4.62-1.amzn2023.x86_64 5/12
Verifying : httpd-core-2.4.62-1.amzn2023.x86_64 6/12
Verifying : httpd-filesystem-2.4.62-1.amzn2023.noarch 7/12
Verifying : httpd-tools-2.4.62-1.amzn2023.x86_64 8/12
Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 9/12
Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch 10/12
Verifying : mod_http2-2.0.27-1.amzn2023.0.3.x86_64 11/12
Verifying : mod_lua-2.4.62-1.amzn2023.x86_64 12/12

Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
httpd-2.4.62-1.amzn2023.x86_64
httpd-filesystem-2.4.62-1.amzn2023.noarch
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mod_http2-2.0.27-1.amzn2023.0.3.x86_64

april-util-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-core-2.4.62-1.amzn2023.x86_64
httpd-tools-2.4.62-1.amzn2023.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_lua-2.4.62-1.amzn2023.x86_64

complete!
root@ip-172-31-90-121 ~]# cd /var/www/html
root@ip-172-31-90-121 html]# echo "Welcome to IEC" > index.html
root@ip-172-31-90-121 html]# systemctl start httpd
root@ip-172-31-90-121 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
root@ip-172-31-90-121 html]# curl localhost
Welcome to IEC
root@ip-172-31-90-121 html]#
```



EMAIL OPT-IN
FOR WORDPRESS

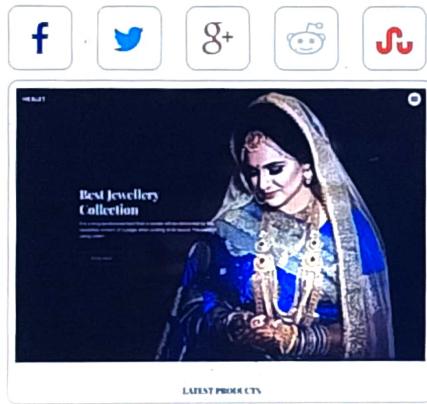
SOCIAL SHARING
FOR WORDPRESS

WORDPRESS THEME

THEMES

HEALET FREE CSS TEMPLATE

OUR SPONSORS



HTML Design

HTML 5

Responsive, 4 Columns

Dark on Light



Creative Commons

26 September 2023

Bootstrap, eCommerce, Jewellery, jQuery, Responsive

DOWNLOAD

LIVE DEMO

Download the Healet website template



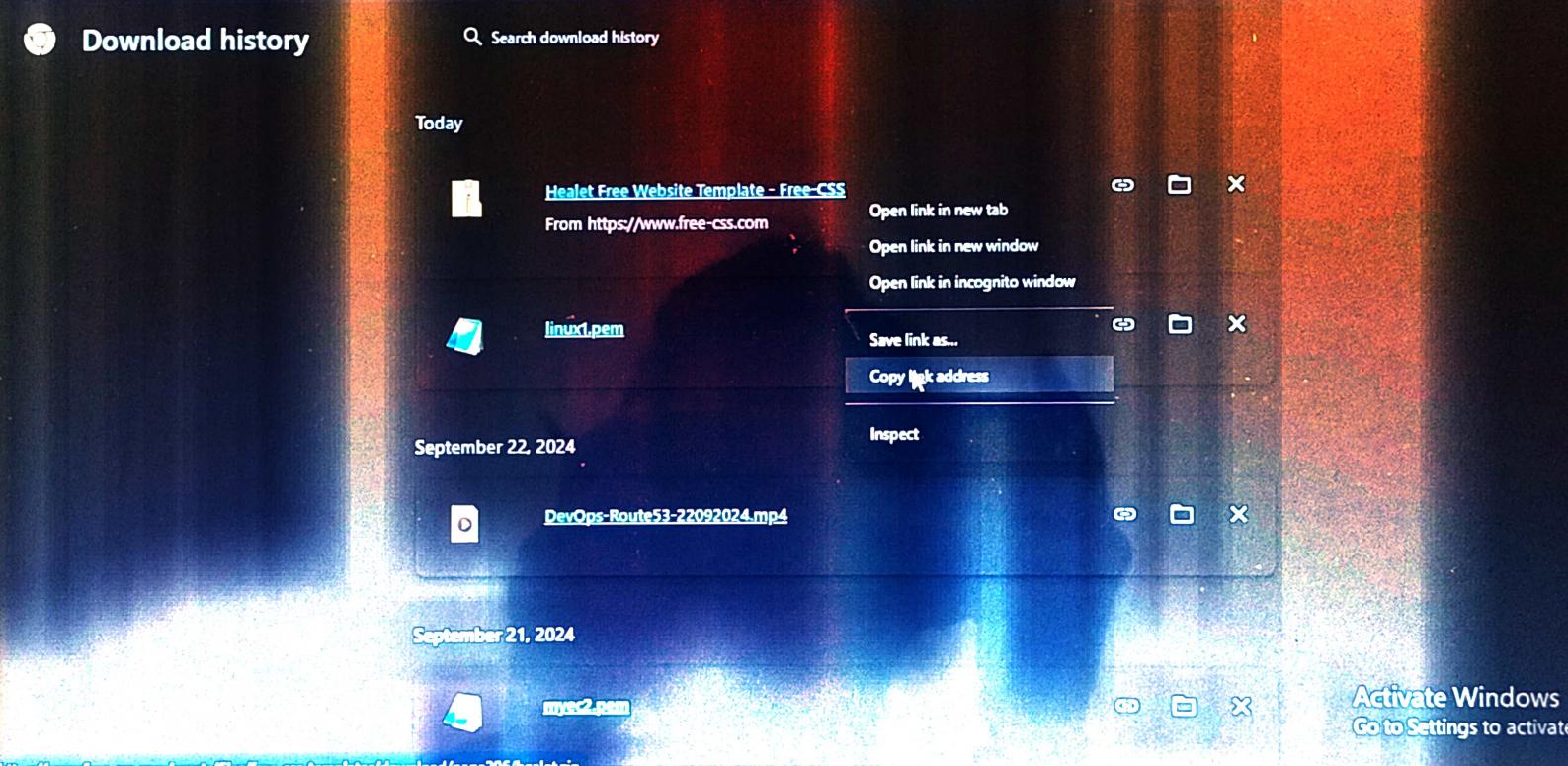
Type here to search

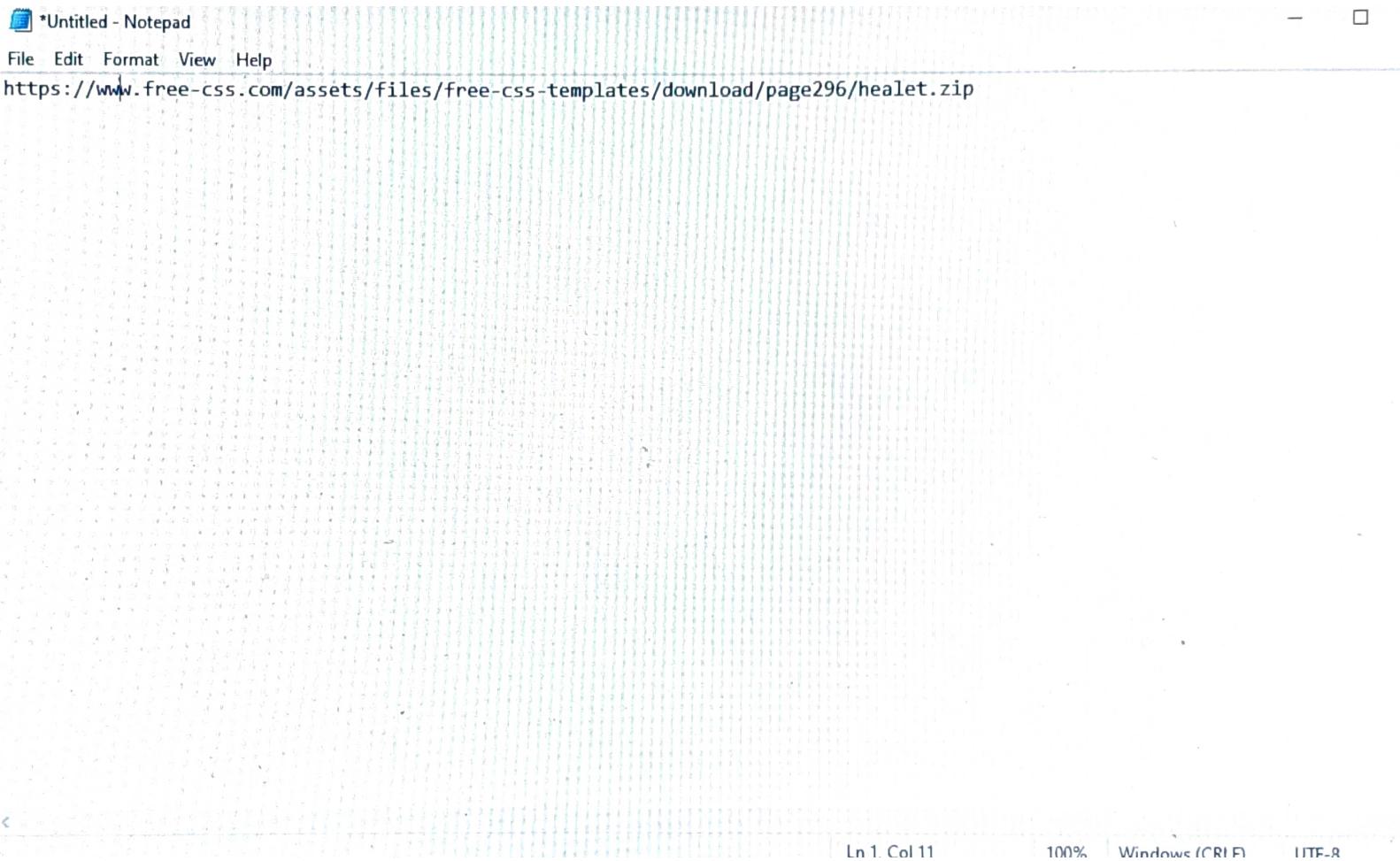


Activate Win
Go to Settings to

<https://www.free-css.com/assets/files/free-css-template/heulet/>

28°C Light rain





```
Select root@ip-172-31-90-121:/var/www/html
Try `wget --help` for more options.
[root@ip-172-31-90-121 html]# wget https://www.free-css.com/assets/files/free-css-templates/download/page296/healelet.zip
--2024-09-26 05:45:24--  https://www.free-css.com/assets/files/free-css-templates/download/page296/healelet.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::28f
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2040237 (1.9M) [application/zip]
Saving to: 'healelet.zip'

healelet.zip          100%[=====] 1.95M  2.35MB/s  in 0.8s
2024-09-26 05:45:25 (2.35 MB/s) - 'healelet.zip' saved [2040237/2040237]

[root@ip-172-31-90-121 html]# ls
healelet.zip  index.html
[root@ip-172-31-90-121 html]# unzip healelet.zip
Archive: healelet.zip
  creating: healelet-html/
  inflating: healelet-html/about.html
  inflating: healelet-html/blog.html
  creating: healelet-html/css/
  inflating: healelet-html/css/bootstrap.css
  inflating: healelet-html/css/font-awesome.min.css
  inflating: healelet-html/css/responsive.css
  inflating: healelet-html/css/style.css
  inflating: healelet-html/css/style.css.map
  inflating: healelet-html/css/style.scss
  creating: healelet-html/fonts/
```

```
Select root@ip-172-31-90-121:/var/www/html
←   inflating: healet-html/index.html
    creating: healet-html/js/
      inflating: healet-html/js/bootstrap.js
      inflating: healet-html/js/custom.js
      inflating: healet-html/js/jquery-3.4.1.min.js
      inflating: healet-html/shop.html
[root@ip-172-31-90-121 html]# ls
healet.html  healet.zip  index.html
[root@ip-172-31-90-121 html]# cd healet.html
-bash: cd: healet.html: No such file or directory
[root@ip-172-31-90-121 html]# cd healet.html/
-bash: cd: healet.html/: No such file or directory
[root@ip-172-31-90-121 html]# cd healet-html/
[root@ip-172-31-90-121 healet-html]# ls
about.html  blog.html  css  fonts  images  index.html  js  shop.html
[root@ip-172-31-90-121 healet-html]# cp -a * ../
cp: overwrite '../index.html'?
[root@ip-172-31-90-121 healet-html]# ls
about.html  blog.html  css  fonts  images  index.html  js  shop.html
[root@ip-172-31-90-121 healet-html]# cd ..
[root@ip-172-31-90-121 html]# rm -rf healet-html
[root@ip-172-31-90-121 html]# rm -rf healet.zip
[root@ip-172-31-90-121 html]# ls
about.html  blog.html  css  fonts  images  index.html  js  shop.html
[root@ip-172-31-90-121 html]# systemctl start httpd
[root@ip-172-31-90-121 html]# systemctl enable httpd
[root@ip-172-31-90-121 html]# curl localhost
Welcome to IEC
[root@ip-172-31-90-121 html]#
```

These rules control the incoming traffic that's allowed to reach the instance.

Inbound rules Info

Security group rule ID	Type	Info	Protocol	Port range	Source	Info	Description - optional	Info	
sgr-0198d3e249805d853	SSH	▼	TCP	22	Cu... ▼	Q			Delete
-	HTTP	▼	TCP	80	An... ▼	Q 0.0.0.0/0 X			Delete

[Add rule](#)

Cancel

[Preview changes](#)

[Save rules](#)

Go to Settings to activate Windows.

```
[root@ip-172-31-90-121 ~]# fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 560B80C4-7164-45FA-B599-60A610A3D271

Device      Start    End  Sectors Size Type
/dev/xvda1   24576 16777182 16752607   8G Linux filesystem
/dev/xvda127 22528     24575      2048   1M BIOS boot
/dev/xvda128  2048    22527    20480  10M EFI System

Partition table entries are not in disk order.

Disk /dev/xvdb: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
[root@ip-172-31-90-121 ~]# fdisk /dev/xvdb

Welcome to fdisk (util-linux 2.37.4).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table.
Created a new DOS disklabel with disk identifier 0x04cd45cb.

Command (m for help): m
```

```
root@ip-172-31-90-121:/media/part1
← Created a new DOS disklabel with disk identifier 0x04cd45cb.

M Command (m for help): m

a Help:
DOS (MBR)
a toggle a bootable flag
b edit nested BSD disklabel
c toggle the dos compatibility flag

Generic
d delete a partition
F list free unpartitioned space
l list known partition types
n add a new partition
p print the partition table
t change a partition type
v verify the partition table
i print information about a partition

Misc
m print this menu
u change display/entry units
x extra functionality (experts only)

Script
I load disk layout from sfdisk script file
O dump disk layout to sfdisk script file
```

```
root@ip-172-31-90-121:/media/part1
← G  create a new empty SGI (IRIX) partition table
o  create a new empty DOS partition table
s  create a new empty Sun partition table
M

a|Command (m for help): n
Partition type
 p  primary (0 primary, 0 extended, 4 free)
 e  extended (container for logical partitions)
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-41943039, default 2048): 2048
Last sector, +/-sectors or +/-size{K,M,G,T,P} (2048-41943039, default 41943039): +10G
Created a new partition 1 of type 'Linux' and of size 10 GiB.

Command (m for help): m

Help:

DOS (MBR)
 a  toggle a bootable flag
 b  edit nested BSD disklabel
 c  toggle the dos compatibility flag

Generic
 d  delete a partition
 F  list free unpartitioned space
 l  list known partition types
 n  add a new partition
```

```
[root@ip-172-31-90-121:/media/part1]
```

```
o  create a new empty DOS partition table  
s  create a new empty Sun partition table
```

```
a Command (m for help): p
```

```
Disk /dev/xvdb: 20 GiB, 21474836480 bytes, 41943040 sectors  
Units: sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disklabel type: dos  
Disk identifier: 0x04cd45cb
```

```
Device      Boot Start      End Sectors Size Id Type  
/dev/xvdb1        2048 20973567 20971520 10G 83 Linux
```

```
Command (m for help): w
```

```
The partition table has been altered.  
Calling ioctl() to re-read partition table.  
Syncing disks.
```

```
[root@ip-172-31-90-121 ~]# fdisk /dev/xvdb
```

```
Welcome to fdisk (util-linux 2.37.4).  
Changes will remain in memory only, until you decide to write them.  
Be careful before using the write command.
```

```
Command (m for help): m
```

```
Help:
```

```
✓  Connect to instance EC2 Linux  X  Mediobit's free website  □
root@ip-172-31-90-121:/media/part1
← Command (m for help): n
Partition type
M p primary (1 primary, 0 extended, 3 free)
e extended (container for logical partitions)
Select (default p): p
Partition number (2-4, default 2): 2
First sector (20973568-41943039, default 20973568): 2048
Value out of range.
First sector (20973568-41943039, default 20973568): 20973568
Last sector, +/-sectors or +/-size{K,M,G,T,P} (20973568-41943039, default 41943039): +10G
Value out of range.
Last sector, +/-sectors or +/-size{K,M,G,T,P} (20973568-41943039, default 41943039): 21000000
Created a new partition 2 of type 'Linux' and of size 10 GiB.

Command (m for help): w
The partition table has been altered.
Calling ioctl() to re-read partition table.
Syncing disks.

[root@ip-172-31-90-121 ~]# fdisk -l
Disk /dev/xvda: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 560B80C4-7164-45FA-B599-60A610A3D271

Device      Start    End  Sectors Size Type
/dev/xvda1  24576 16777182 16752607   8G Linux filesystem
```

```
root@ip-172-31-90-121:/media/part1
← Partition table entries are not in disk order.

Disk /dev/xvdb: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x04cd45cb

Device      Boot   Start     End   Sectors  Size Type
/dev/xvdb1        2048 20973567 20971520   10G 83 Linux
/dev/xvdb2    20973568 41943039 20969472   10G 83 Linux
[root@ip-172-31-90-121 ~]# mkfs -t xfs /dev/xvdb1
meta-data=/dev/xvdb1              isize=512    agcount=4, agsize=655360 blks
                                  sectsz=512   attr=2, projid32bit=1
                                  crc=1       finobt=1, sparse=1, rmapbt=0
                                  reflink=1   bigtime=1 inobtcount=1
data                    =         bsize=4096   blocks=2621440, imaxpct=25
                           sunit=0    swidth=0 blks
naming      =version 2          bsize=4096   ascii-ci=0, ftype=1
log         =internal log       bsize=4096   blocks=16384, version=2
                           sectsz=512   sunit=0 blks, lazy-count=1
realtime    =none               extsz=4096   blocks=0, rtextents=0
[root@ip-172-31-90-121 ~]# mkfs -t xfs /dev/xvdb2
meta-data=/dev/xvdb2              isize=512    agcount=4, agsize=655296 blks
                                  sectsz=512   attr=2, projid32bit=1
                                  crc=1       finobt=1, sparse=1, rmapbt=0
                                  reflink=1   bigtime=1 inobtcount=1
                           bsize=4096   blocks=2621184, imaxpct=25
```

```
root@ip-172-31-90-121:/media/part1
← ino=13108)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
tmpfs on /tmp type tmpfs (rw,nosuid,nodev,seclabel,size=486140k,nr_inodes=1048576)
ramfs on /run/credentials/systemd-sysctl.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
ramfs on /run/credentials/systemd-tmpfiles-setup-dev.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
systemd-1 on /boot/efi type autofs (rw,relatime,fd=38,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=14186)
ramfs on /run/credentials/systemd-tmpfiles-setup.service type ramfs (ro,nosuid,nodev,noexec,relatime,seclabel,mode=700)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
/dev/xvda128 on /boot/efi type vfat (rw,noatime,fmask=0077,dmask=0077,codepage=437,iocharset=ascii,shortname=winnt,error_s=remount-ro,x-systemd.automount)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=97224k,nr_inodes=24306,mode=700,uid=1000,gid=1000)
/dev/xvdb1 on /media/part1 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
/dev/xvdb2 on /media/part2 type xfs (rw,relatime,seclabel,attr2,inode64,logbufs=8,logbsize=32k,noquota)
[root@ip-172-31-90-121 ~]# cd /media/part1
[root@ip-172-31-90-121 part1]# ls
[root@ip-172-31-90-121 part1]# touch t1
[root@ip-172-31-90-121 part1]# ls
t1
[root@ip-172-31-90-121 part1]#
```

<ul style="list-style-type: none"> Reserved Instances Dedicated Hosts Capacity Reservations <small>New</small> ▼ Images AMIs AMI Catalog ▼ Elastic Block Store Volumes Snapshots Lifecycle Manager ▼ Network & Security Security Groups Elastic IPs 	<p>Amazon Linux (Inferred)</p> <p>Platform details</p> <p>Linux/UNIX</p> <p>Stop protection</p> <p>Disabled</p> <p>Instance auto-recovery</p> <p>Default</p> <p>AMI Launch index</p> <p>0</p> <p>Credit specification</p> <p>standard</p> <p>Usage operation</p> <p>RunInstances</p>	<p>ami-0ebfd941bbafe70c6</p> <p>AMI name</p> <p>al2023-ami-2023.5.20240916.0-kernel-6.1-x86_64</p> <p>Launch time</p> <p>Thu Sep 26 2024 10:39:52 GMT+0530 (India Standard Time) (about 1 hour)</p> <p>Lifecycle</p> <p>normal</p> <p>Key pair assigned at launch</p> <p>linux1</p> <p>Kernel ID</p> <p>-</p> <p>RAM disk ID</p> <p>-</p>	<p>disabled</p> <p>Termination protection</p> <p>Disabled</p> <p>AMI location</p> <p>amazon/al2023-ami-2023.5.20240916.0-kernel-6.1-x86_64</p> <p>Stop-hibernate behavior</p> <p>Disabled</p> <p>State transition reason</p> <p>-</p> <p>State transition message</p> <p>-</p> <p>Owner</p> <p>Activate Windows</p> <p>664457182020</p> <p>Go to Settings to activate</p>
--	--	---	---

https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#Volumes:

© 2024, Amazon Web Services, Inc. or its affiliates.

[Privacy](#) [Terms](#)

