EC2   Elastic Kubernetes Service   Elastic Container Registry   S3   VPC

EC2 > Instances > Launch an instance

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

| TF-EC2 | | 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

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

Recents   **Quick Start**

Amazon Linux	macOS	Ubuntu	Windows	Red Hat	SUSE Linux	Debian
aws	Mac	ubuntu	■■ Microsoft	RedHat	🦎	debian

🔍 **Browse more AMIs**

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type   **Free tier eligible**
ami-0f918f7e67a3323f0 (64-bit (x86)) / ami-02f607855bfce66b6 (64-bit (Arm))

▼ Summary

Number of instances | Info

| 1 |

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0f918f7e67a3323f0

Virtual ... r type (instance type)
t2.m...

Firewa... ...ecurity group)
launch-...zard-9

Storage (volumes)
1 volume(s) - 16 GiB

ⓘ **Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to ✕

Cancel   **Launch instance**

🖳 Preview code

EC2   Elastic Kubernetes Service   Elastic Container Registry   S3   VPC

☰   EC2 > Instances > Launch an instance

Amazon Machine Image (AMI)

Ubuntu Server 24.04 LTS (HVM), SSD Volume Type   **Free tier eligible**
ami-0f918f7e67a3323f0 (64-bit (x86)) / ami-02f607855bfce66b6 (64-bit (Arm))
Virtualization: hvm   ENA enabled: true   Root device type: ebs

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-0f918f7e67a3323f0	2025-06-10	ubuntu	Verified provider

▼ Instance type   Info | Get advice

Instance type

t2.micro   **Free tier eligible**
Family: t2   1 vCPU   1 GiB Memory   Current generation: true   On-Demand Windows base pricing: 0.017 USD per Hour
On-Demand RHEL base pricing: 0.0268 USD per Hour   On-Demand Linux base pricing: 0.0124 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0142 USD per Hour   On-Demand SUSE base pricing: 0.0124 USD per Hour

⚪ 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*

▼ Summary

Number of instances | Info

1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0f918f7e67a3323f0

Virtual server type (instance type)
t2.micro

Firewall (security group)
launch-wizard-9

Storage (volumes)
1 volume(s) - 16 GiB

ⓘ **Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to ✕

Cancel   **Launch instance**

🖥 Preview code
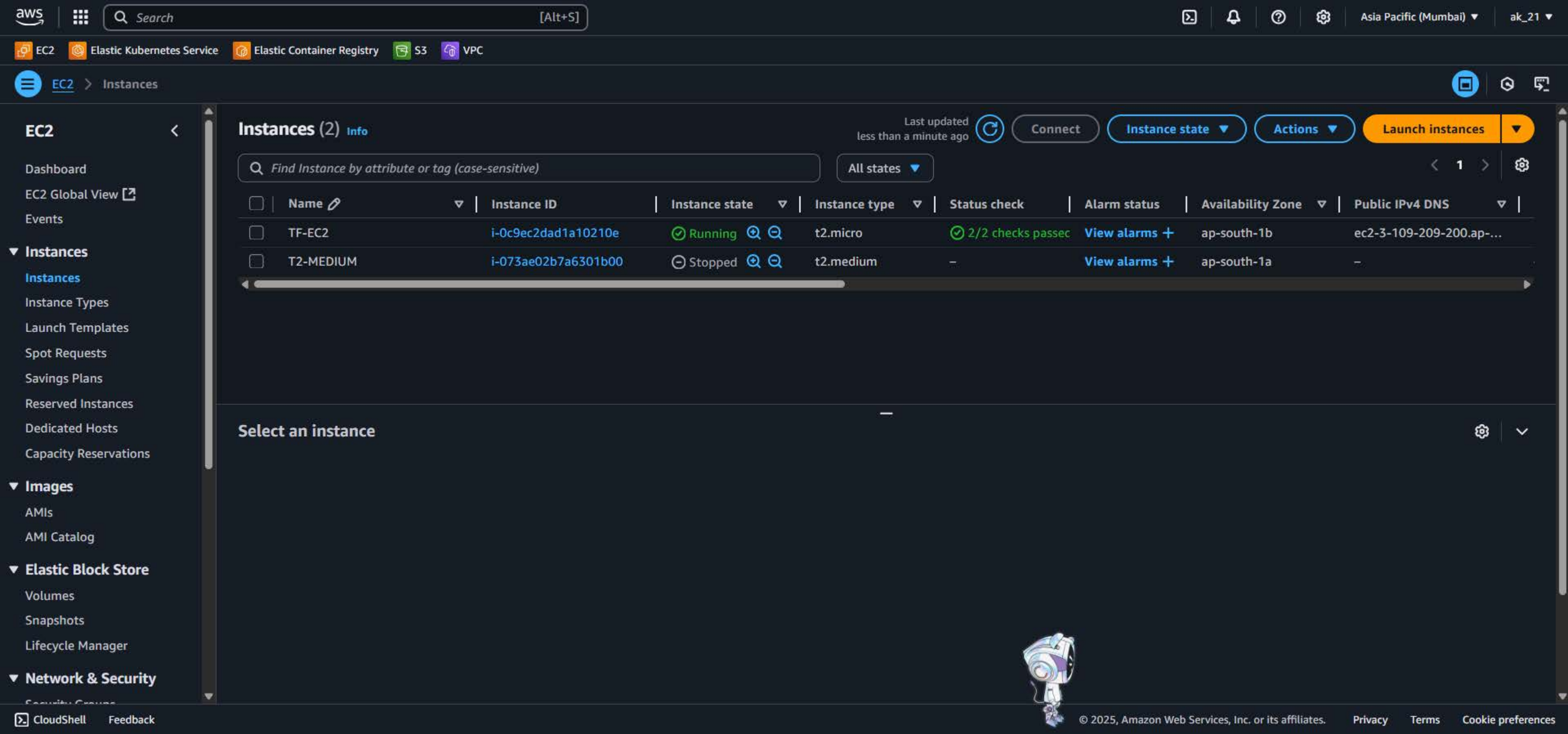
▼ Key pair (login) Info

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*

vmamzkey01   ▼   ↻ **Create new key pair**

▼ Network settings Info

Edit

Network | Info

vpc-025be5d44520af079

Subnet | Info

No preference (Default subnet in any availability zone)

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) | Info

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

Common security groups | Info

Select security groups   ▼

↻ **Compare security group rules**

launch-wizard-9   sg-0c94c647e8263e496 ✕
VPC: vpc-025be5d44520af079

▼ Summary

Number of instances | Info

1

Software Image (AMI)

Canonical, Ubuntu, 24.04, amd6...read more
ami-0f918f7e67a3323f0

Virtual server type (instance type)

t2.micro

Firewall (security group)

launch-wizard-9

Storage (volumes)

1 volume(s) - 16 GiB

ⓘ **Free tier:** In your first year of opening an AWS account,   ✕
you get 750 hours per month of t2.micro instance
usage (or t3.micro where t2.micro isn't available) when
used with free tier AMIs, 750 hours per month of public
IPv4 address usage, 30 GiB of EBS storage, 2 million
I/Os, 1 GB of snapshots, and 100 GB of bandwidth to

Cancel   **Launch instance**

💻 **Preview code**

Select security groups ▼

launch-wizard-9 sg-0c94c647e8263e496 ✕
VPC: vpc-025be5d44520af079

Security groups that you add or remove here will be added to or removed from all your network interfaces.

↻ Compare security group rules

▼ **Summary**

Number of instances | Info

1

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd6...read more
ami-0f918f7e67a3323f0

Virtual server type (instance type)
t2.micro

Firewall (security group)
launch-wizard-9

Storage (volumes)
1 volume(s) - 16 GiB

▼ Configure storage Info   Advanced

1x 16 GiB   gp3 ▼   Root volume, 3000 IOPS, Not encrypted

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

Add new volume

The selected AMI contains instance store volumes, however the instance does not allow any instance store volumes. None of the instance store volumes from the AMI will be accessible from the instance.

⏱ Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.   ↻

0 x File systems   Edit

ⓘ **Free tier:** In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to   ✕

Cancel   **Launch instance**

🖵 Preview code

▶ **Advanced details** Info

EC2 | Elastic Kubernetes Service | Elastic Container Registry | S3 | VPC

EC2

Dashboard
EC2 Global View ↗
Events

▼ **Instances**
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations

▼ **Images**
AMIs
AMI Catalog

▼ **Elastic Block Store**
Volumes
Snapshots
Lifecycle Manager

▼ **Network & Security**
Security Groups

Instances (2) Info

Last updated less than a minute ago

Connect | Instance state ▼ | Actions ▼ | **Launch instances** ▼

🔍 Find Instance by attribute or tag (case-sensitive)

All states ▼

< 1 >

	Name ✎	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☐	TF-EC2	i-0c9ec2dad1a10210e	⊘ Running ⊕ ⊖	t2.micro	⊘ 2/2 checks passec	View alarms +	ap-south-1b	ec2-3-109-209-200.ap-...
☐	T2-MEDIUM	i-073ae02b7a6301b00	⊝ Stopped ⊕ ⊖	t2.medium	–	View alarms +	ap-south-1a	–

Select an instance

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

System information as of Wed Jul 16 10:20:19 UTC 2025

  System load:  0.18              Processes:              106
  Usage of /:   11.8% of 14.46GB  Users logged in:        0
  Memory usage: 21%               IPv4 address for enX0:  172.31.2.125
  Swap usage:   0%

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-172-31-2-125:~$ sudo apt update && sudo apt upgrade -y
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

apply   Create or update infrastructure
destroy   Destroy previously-created infrastructure

All other commands:
  console   Try Terraform expressions at an interactive command prompt
  fmt   Reformat your configuration in the standard style
  force-unlock   Release a stuck lock on the current workspace
  get   Install or upgrade remote Terraform modules
  graph   Generate a Graphviz graph of the steps in an operation
  import   Associate existing infrastructure with a Terraform resource
  login   Obtain and save credentials for a remote host
  logout   Remove locally-stored credentials for a remote host
  metadata   Metadata related commands
  modules   Show all declared modules in a working directory
  output   Show output values from your root module
  providers   Show the providers required for this configuration
  refresh   Update the state to match remote systems
  show   Show the current state or a saved plan
  state   Advanced state management
  taint   Mark a resource instance as not fully functional
  test   Execute integration tests for Terraform modules
  untaint   Remove the 'tainted' state from a resource instance
  version   Show the current Terraform version
  workspace   Workspace management

Global options (use these before the subcommand, if any):
  -chdir=DIR   Switch to a different working directory before executing the
                 given subcommand.
  -help   Show this help output, or the help for a specified subcommand.
  -version   An alias for the "version" subcommand.
ubuntu@ip-172-31-2-125:~$ terraform --version
Terraform v1.12.2
on linux_amd64
ubuntu@ip-172-31-2-125:~$

```
  GNU nano 7.2                                                    provider.tf *

terraform {
  required_providers {
    aws = {
      source  = "hashicorp/aws"
      version = "~> 5.0"
    }
  }
}

provider "aws" {
  region = "ap-south-1"
  alias = "region1"
}

provider "aws" {
  region = "us-east-1"
  alias = "region2"
}
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
variable "ami_id1" {
    type      = string
    default = "ami-053b0d53c279acc90"
}

variable "ami_id2" {
    type      = string
    default = "ami-020cba7c55df1f615"
}

variable "key_name1" {
    type      = string
    default = "vmamzkey01"
}

variable "key_name2" {
    type      = string
    default = "ec2east"
}

variable "instance_type" {
    type = string
    default = "t2.micro"
}
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
resource "aws_security_group" "sg_region1" {
  provider = aws.region1
  name     = "allow-http-ssh"

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}

resource "aws_security_group" "sg_region2" {
  provider = aws.region2
  name     = "allow-http-ssh"
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

CloudShell   Feedback   Privacy   © 2025, Amazon Web Services, Inc. or its affiliates.   Terms   Cookie preferences

```
resource "aws_security_group" "sg_region2" {
provider = aws.region2
name     = "allow-http-ssh"

  ingress {
    from_port   = 80
    to_port     = 80
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  ingress {
    from_port   = 22
    to_port     = 22
    protocol    = "tcp"
    cidr_blocks = ["0.0.0.0/0"]
  }

  egress {
    from_port   = 0
    to_port     = 0
    protocol    = "-1"
    cidr_blocks = ["0.0.0.0/0"]
  }
}

resource "aws_instance" "south_ec2" {
    provider            = aws.region1
    ami                 = var.ami_id1
    instance_type       = var.instance_type
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
        cidr_blocks = ["0.0.0.0/0"]
    }
}

resource "aws_instance" "south_ec2" {
    provider              = aws.region1
    ami                   = var.ami_id1
    instance_type         = var.instance_type
    key_name              = var.key_name1
    vpc_security_group_ids = [aws_security_group.sg_region1.id]
    user_data             = file("userdata.tpl")
    tags = {
      Name = "South-EC2"
    }
}

resource "aws_instance" "east_ec2" {
    provider              = aws.region2
    ami                   = var.ami_id2
    instance_type         = var.instance_type
    key_name              = var.key_name2
    vpc_security_group_ids = [aws_security_group.sg_region2.id]
    user_data             = file("userdata.tpl")
    tags = {
      Name = "East-EC2"
    }
}
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

Search   [Alt+S]

Asia Pacific (Mumbai) ▼   ak_21 ▼

EC2   Elastic Kubernetes Service   Elastic Container Registry   S3   VPC

```
  GNU nano 7.2                                            userdata.tpl *
#!/bin/bash
sudo apt update
sudo apt upgrade -y
sudo apt install -y nginx
sudo systemctl start nginx
```

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location	M-U Undo	M-A Set Mark	M-] To Bracket	M-Q Previous	^B Back
^X Exit	^R Read File	^\ Replace	^U Paste	^J Justify	^/ Go To Line	M-E Redo	M-6 Copy	^Q Where Was	M-W Next	^F Forward

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano provider.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano variable.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano main.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano userdata.tpl
ubuntu@ip-172-31-2-125:~/terraform-task2$ ls
main.tf  provider.tf  userdata.tpl  variable.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ 
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano provider.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano variable.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano main.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ nano userdata.tpl
ubuntu@ip-172-31-2-125:~/terraform-task2$ ls
main.tf  provider.tf  userdata.tpl  variable.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$ terraform init
Initializing the backend...
Initializing provider plugins...
- Finding hashicorp/aws versions matching "~> 5.0"...
- Installing hashicorp/aws v5.100.0...
- Installed hashicorp/aws v5.100.0 (signed by HashiCorp)
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
ubuntu@ip-172-31-2-125:~/terraform-task2$ terraform fmt
main.tf
variable.tf
ubuntu@ip-172-31-2-125:~/terraform-task2$
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
+ protocol          = tcp
+ security_groups   = []
+ self              = false
+ to_port           = 22
  # (1 unchanged attribute hidden)
},
+ {
+ cidr_blocks       = [
    + "0.0.0.0/0",
  ]
+ from_port         = 80
+ ipv6_cidr_blocks  = []
+ prefix_list_ids   = []
+ protocol          = "tcp"
+ security_groups   = []
+ self              = false
+ to_port           = 80
  # (1 unchanged attribute hidden)
},
]
+ name                  = "allow-http-ssh"
+ name_prefix           = (known after apply)
+ owner_id              = (known after apply)
+ revoke_rules_on_delete = false
+ tags_all              = (known after apply)
+ vpc_id                = (known after apply)
}

Plan: 4 to add, 0 to change, 0 to destroy.


Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-2-125:~/terraform-task2$ terraform apply
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
      + enclave_options (known after apply)

      + ephemeral_block_device (known after apply)

      + instance_market_options (known after apply)

      + maintenance_options (known after apply)

      + metadata_options (known after apply)

      + network_interface (known after apply)

      + private_dns_name_options (known after apply)

      + root_block_device (known after apply)

    }

Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.south_ec2: Creating...
aws_instance.south_ec2: Still creating... [00m10s elapsed]
aws_instance.south_ec2: Still creating... [00m20s elapsed]
aws_instance.south_ec2: Still creating... [00m30s elapsed]
aws_instance.south_ec2: Creation complete after 31s [id=i-078eb7314fb82f71c]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-2-125:~/terraform-task2$
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

EC2   Elastic Kubernetes Service   Elastic Container Registry   S3   VPC

☰ EC2 > Instances

EC2 ‹

Dashboard
EC2 Global View ⬚
Events

▼ **Instances**
    Instances
    Instance Types
    Launch Templates
    Spot Requests
    Savings Plans
    Reserved Instances
    Dedicated Hosts
    Capacity Reservations

▼ **Images**
    AMIs
    AMI Catalog

▼ **Elastic Block Store**
    Volumes
    Snapshots
    Lifecycle Manager

▼ **Network & Security**
    Security Groups

Instances (1/2) Info   Last updated 1 minute ago ⟳   **Connect**   **Instance state ▼**   **Actions ▼**   **Launch instances** ▼

🔍 Find Instance by attribute or tag (case-sensitive)   **Running ▼**   ‹ 1 ›   ⚙

☐	Name ✎ ▽	Instance ID	Instance state ▽	Instance type ▽	Status check	Alarm status	Availability Zone ▽	Public IPv4 DNS ▽
☐	TF-EC2	i-0c9ec2dad1a10210e	⊘ Running 🔍 🔍	t2.micro	⊘ 2/2 checks passed	View alarms ✚	ap-south-1b	ec2-3-109-209-200.ap-...
☑	South-EC2	i-024da0f6a9e19a5b2	⊘ Running 🔍 🔍	t2.micro	⊘ 2/2 checks passed	View alarms ✚	ap-south-1b	ec2-13-203-79-198.ap-...

i-024da0f6a9e19a5b2 (South-EC2)   ⚙ ⌄

| **Details** | Status and alarms | Monitoring | Security | Networking | Storage | Tags |

▼ **Instance summary** Info

Instance ID
⧉ i-024da0f6a9e19a5b2

IPv6 address
–

Hostname type
IP name: ip-172-31-15-25.ap-south-1.compute.internal

Public IPv4 address
⧉ 13.203.79.198 | open address ⬚

Instance state
⊘ Running

Private IP DNS name (IPv4 only)
⧉ ip-172-31-15-25.ap-south-1.compute.internal

Private IPv4 addresses
⧉ 172.31.15.25

Public DNS
⧉ ec2-13-203-79-198.ap-south-1.compute.amazonaws.com |
open address ⬚

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

EC2

- Dashboard
- EC2 Global View ⧉
- Events

Instances
- Instances
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts
- Capacity Reservations

Images
- AMIs
- AMI Catalog

Elastic Block Store
- Volumes
- Snapshots
- Lifecycle Manager

Network & Security
- Security Groups

Instances (1/1) Info

Last updated less than a minute ago

Connect   Instance state ▼   Actions ▼   **Launch instances** ▼

Find Instance by attribute or tag (case-sensitive)   Running ▼   ‹ 1 ›

☑	Name ✎	▽	Instance ID	Instance state	▽	Instance type	▽	Status check	Alarm status	Availability Zone	▽	Public IPv4 DNS	▽	Public IPv4 ..
☑	East-EC2		i-071a8c78334e3c2b0	⊘ Running		t2.micro		⊘ 2/2 checks passed	View alarms ＋	us-east-1c		ec2-13-222-236-70.co...		13.222.236.70

i-071a8c78334e3c2b0 (East-EC2)

Details	Status and alarms	Monitoring	Security	Networking	Storage	Tags

▼ **Instance summary** Info

Instance ID
⧉ i-071a8c78334e3c2b0

IPv6 address
–

Hostname type
IP name: ip-172-31-86-203.ec2.internal

Public IPv4 address
⧉ 13.222.236.70 | open address ⧉

Instance state
⊘ Running

Private IP DNS name (IPv4 only)
⧉ ip-172-31-86-203.ec2.internal

Private IPv4 addresses
⧉ 172.31.86.203

Public DNS
⧉ ec2-13-222-236-70.compute-1.amazonaws.com | open address ⧉

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

```
)

Plan: 2 to add, 1 to change, 2 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

aws_instance.south_ec2: Destroying... [id=i-078eb7314fb82f71c]
aws_instance.east_ec2: Destroying... [id=i-0e91a8cfe2064df2b]
aws_instance.south_ec2: Still destroying... [id=i-078eb7314fb82f71c,  elapsed]
aws_instance.east_ec2: Still destroying... [id=i-0e91a8cfe2064df2b, 0     elapsed]
aws_instance.south_ec2: Still destroying... [id=i-078eb7314fb82f71c, 00  0s elapsed]
aws_instance.east_ec2: Still destroying... [id=i-0e91a8cfe2064df2b, 00m  s elapsed]
aws_instance.south_ec2: Destruction complete after 30s
aws_instance.south_ec2: Creating...
aws_instance.east_ec2: Still destroying... [id=i-0e91a8cfe2064df2b, 00m30s elapsed]
aws_instance.east_ec2: Destruction complete after 32s
aws_security_group.sg_region2: Modifying... [id=sg-03402524e85381130]
aws_security_group.sg_region2: Modifications complete after 1s [id=sg-03402524e85381130]
aws_instance.east_ec2: Creating...
aws_instance.south_ec2: Still creating... [00m10s elapsed]
aws_instance.east_ec2: Still creating... [00m10s elapsed]
aws_instance.south_ec2: Still creating... [00m20s elapsed]
aws_instance.east_ec2: Still creating... [00m20s elapsed]
aws_instance.south_ec2: Still creating... [00m30s elapsed]
aws_instance.south_ec2: Creation complete after 31s [id=i-024da0f6a9e19a5b2]
aws_instance.east_ec2: Still creating... [00m30s elapsed]
aws_instance.east_ec2: Creation complete after 35s [id=i-071a8c78334e3c2b0]

Apply complete! Resources: 2 added, 1 changed, 2 destroyed.
ubuntu@ip-172-31-2-125:~/terraform-task2$ terraform destroy -auto-approve
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125

```
              protocol           = tcp
            - security_groups  = []
            - self             = false
            - to_port          = 80
              # (1 unchanged attribute hidden)
          },
        ] -> null
      - name                   = "allow-http-ssh" -> null
      - owner_id               = "098588167308" -> null
      - revoke_rules_on_delete = false -> null
      - tags                   = {} -> null
      - tags_all               = {} -> null
      - vpc_id                 = "vpc-08d67a5f9f5f206bb" -> null
        # (1 unchanged attribute hidden)
    }

Plan: 0 to add, 0 to change, 4 to destroy.
aws_instance.south_ec2: Destroying... [id=i-024da0f6a9e19a5b2]
aws_instance.east_ec2: Destroying... [id=i-071a8c78334e3c2b0]
aws_instance.south_ec2: Still destroying... [id=i-024da0f6a9e19a5b2, 00m10s elapsed]
aws_instance.east_ec2: Still destroying... [id=i-071a8c78334e3c2b0, 00m10s elapsed]
aws_instance.south_ec2: Still destroying... [id=i-024da0f6a9e19a5b2, 00m20s elapsed]
aws_instance.east_ec2: Still destroying... [id=i-071a8c78334e3c2b0, 00m20s elapsed]
aws_instance.south_ec2: Still destroying... [id=i-024da0f6a9e19a5b2, 00m30s elapsed]
aws_instance.east_ec2: Still destroying... [id=i-071a8c78334e3c2b0, 00m30s elapsed]
aws_instance.east_ec2: Destruction complete after 32s
aws_security_group.sg_region2: Destroying... [id=sg-03402524e85381130]
aws_security_group.sg_region2: Destruction complete after 1s
aws_instance.south_ec2: Destruction complete after 40s
aws_security_group.sg_region1: Destroying... [id=sg-0bd65a8f5f7a782fd]
aws_security_group.sg_region1: Destruction complete after 0s

Destroy complete! Resources: 4 destroyed.
ubuntu@ip-172-31-2-125:~/terraform-task2$
```

i-0c9ec2dad1a10210e (TF-EC2)

PublicIPs: 3.109.209.200   PrivateIPs: 172.31.2.125