

Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1011-aws x86_64)

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

System information as of Sun Sep 21 20:49:57 UTC 2025

System load:	0.22	Temperature:	-273.1 C
Usage of /:	18.8% of 29.95GB	Processes:	120
Memory usage:	11%	Users logged in:	0
Swap usage:	0%	IPv4 address for ens5:	172.31.9.179

Expanded Security Maintenance **for** Applications is not enabled.

29 updates can be applied immediately.

18 of these updates are standard security updates.

To see these additional updates run: `apt list --upgradable`

6 additional security updates can be applied with ESM Apps.

Learn more about enabling ESM Apps service at <https://ubuntu.com/esm>

Last login: Sun Sep 21 20:46:41 2025 from 3.16.146.3

ubuntu@ip-172-31-9-179:~\$ neofetch

```
      .-/ +oossssoo+/- .
      `:+ssssssssssssssss+:`
      -+ssssssssssssssssyyssss+-
      .ossssssssssssssssdMMMMyssso.
      /ssssssssshdmmNNmmyNMMMMhsssss/
      +sssssssshmydMMMMMMNdddyssssss+
      /ssssssshNMMMyhhyyyhmNMMMNhssssss/
      .sssssssdMMMNhssssssssshNMMMdssssss.
      +sssshhhyNMMNysssssssssssyNMMMyssssss+
      ossyNMMMNyMMhssssssssssshmmhssssssso
      ossyNMMMNyMMhssssssssssshmmhssssssso
      2.499GHz
      +sssshhhyNMMNysssssssssssyNMMMyssssss+
      .sssssssdMMMNhssssssssshNMMMdssssss.
      /ssssssshNMMMyhhyyyhdNMMMNhssssss/
      +sssssssdmydMMMMMMNdddyssssss+
      /ssssssssshdmNNNNmyNMMMMhsssss/
      .ossssssssssssssssdMMMMyssso.
      -+ssssssssssssssssyyssss+-
      `:+ssssssssssssssss+:`
      .-/ +oossssoo+/- .
```

ubuntu@ip-172-31-9-179

OS: Ubuntu 24.04.3 LTS x86_64

Host: t3.large

Kernel: 6.14.0-1011-aws

Uptime: 2 hours, 44 mins

Packages: 1328 (dpkg), 4 (snap)

Shell: bash 5.2.21

Resolution: 800x600

Terminal: /dev/pts/0

CPU: Intel Xeon Platinum 8259CL (2) @

GPU: 00:03.0 Amazon.com, Inc. Device 1111

Memory: 722MiB / 7820MiB

```
ubuntu@ip-172-31-9-179:~$ whoami
ubuntu
ubuntu@ip-172-31-9-179:~$ date
Sun Sep 21 20:50:14 UTC 2025
ubuntu@ip-172-31-9-179:~$ git clone https://github.com/matx104/ibex-devops-task
Cloning into 'ibex-devops-task'...
remote: Enumerating objects: 58, done.
remote: Counting objects: 100% (58/58), done.
remote: Compressing objects: 100% (55/55), done.
remote: Total 58 (delta 4), reused 53 (delta 2), pack-reused 0 (from 0)
Receiving objects: 100% (58/58), 94.99 KiB | 5.00 MiB/s, done.
Resolving deltas: 100% (4/4), done.

ubuntu@ip-172-31-9-179:~$ cd ibex-devops-task/ && ls -lah
total 124K
drwxrwxr-x 8 ubuntu ubuntu 4.0K Sep 21 20:50 .
drwxr-x--- 9 ubuntu ubuntu 4.0K Sep 21 20:50 ..
drwxrwxr-x 8 ubuntu ubuntu 4.0K Sep 21 20:50 .git
-rw-rw-r-- 1 ubuntu ubuntu 356 Sep 21 20:50 .gitignore
-rw-rw-r-- 1 ubuntu ubuntu 2.5K Sep 21 20:50 EVALUATOR_README.md
-rw-rw-r-- 1 ubuntu ubuntu 9.3K Sep 21 20:50 Jenkinsfile
-rw-rw-r-- 1 ubuntu ubuntu 1.1K Sep 21 20:50 LICENSE
-rw-rw-r-- 1 ubuntu ubuntu 5.4K Sep 21 20:50 QUICK_START.md
-rw-rw-r-- 1 ubuntu ubuntu 9.3K Sep 21 20:50 README.md
drwxrwxr-x 2 ubuntu ubuntu 4.0K Sep 21 20:50 app
drwxrwxr-x 2 ubuntu ubuntu 4.0K Sep 21 20:50 ci-cd
-rwxrwxr-x 1 ubuntu ubuntu 1.3K Sep 21 20:50 deploy.sh
-rwxrwxr-x 1 ubuntu ubuntu 275 Sep 21 20:50 destroy.sh
drwxrwxr-x 2 ubuntu ubuntu 4.0K Sep 21 20:50 docs
-rwxrwxr-x 1 ubuntu ubuntu 11K Sep 21 20:50 github-init.sh
-rwxrwxr-x 1 ubuntu ubuntu 6.6K Sep 21 20:50 init.sh
drwxrwxr-x 6 ubuntu ubuntu 4.0K Sep 21 20:50 reports
-rwxrwxr-x 1 ubuntu ubuntu 15K Sep 21 20:50 scan.sh
-rwxrwxr-x 1 ubuntu ubuntu 424 Sep 21 20:50 setup.sh
drwxrwxr-x 3 ubuntu ubuntu 4.0K Sep 21 20:50 terraform
```

```
ubuntu@ip-172-31-9-179:~/ibex-devops-task$ sudo nano terraform/terraform.tfvars
ubuntu@ip-172-31-9-179:~/ibex-devops-task$ ./deploy.sh
```

```
==> terraform init
```

```
Initializing the backend...
```

```
Initializing provider plugins...
```

- Finding hashicorp/aws versions matching "~> 5.0"...
- Finding hashicorp/tls versions matching "~> 4.0"...
- Finding latest version of hashicorp/local...
- Installing hashicorp/aws v5.100.0...
- Installed hashicorp/aws v5.100.0 (signed by HashiCorp)
- Installing hashicorp/tls v4.1.0...
- Installed hashicorp/tls v4.1.0 (signed by HashiCorp)
- Installing hashicorp/local v2.5.3...
- Installed hashicorp/local v2.5.3 (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 now 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.

```
==> terraform plan
data.aws_ami.amazon_linux_2: Reading...
data.aws_vpc.default: Reading...
data.aws_ami.amazon_linux_2: Read complete after 1s [id=ami-01810692271ee44a4]
data.aws_vpc.default: Read complete after 1s [id=vpc-022ada8203ec3ef0a]
data.aws_subnets.default: Reading...
data.aws_subnets.default: Read complete after 0s [id=us-east-2]
```

Terraform used the selected providers to generate the following execution plan.
Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_iam_access_key.cicd_user[0] will be created
+ resource "aws_iam_access_key" "cicd_user" {
  + create_date           = (known after apply)
  + encrypted_secret      = (known after apply)
  + encrypted_ses_smtp_password_v4 = (known after apply)
  + id                   = (known after apply)
  + key_fingerprint       = (known after apply)
  + secret               = (sensitive value)
  + ses_smtp_password_v4 = (sensitive value)
  + status               = "Active"
  + user                 = "ibex-devops-project-sept-2025-104-cicd-user"
}

# aws_iam_instance_profile.ec2_profile will be created
+ resource "aws_iam_instance_profile" "ec2_profile" {
  + arn          = (known after apply)
  + create_date = (known after apply)
  + id          = (known after apply)
  + name        = "ibex-devops-project-sept-2025-104-ec2-profile"
  + name_prefix = (known after apply)
  + path        = "/"
  + role        = "ibex-devops-project-sept-2025-104-ec2-role"
  + tags_all    = (known after apply)
  + unique_id   = (known after apply)
}

# aws_iam_role.ec2_role will be created
```

```

+ resource "aws_iam_role" "ec2_role" {
  + arn                = (known after apply)
  + assume_role_policy = jsonencode(
    {
      + Statement = [
        + {
          + Action    = "sts:AssumeRole"
          + Effect     = "Allow"
          + Principal = {
            + Service = "ec2.amazonaws.com"
          }
        },
      ]
      + Version = "2012-10-17"
    }
  )
  + create_date          = (known after apply)
  + force_detach_policies = false
  + id                  = (known after apply)
  + managed_policy_arns  = (known after apply)
  + max_session_duration = 3600
  + name                 = "ibex-devops-project-sept-2025-104-ec2-role"
  + name_prefix          = (known after apply)
  + path                 = "/"
  + tags                 = {
    + "Environment" = "dev"
    + "ManagedBy"  = "Terraform"
    + "Name"        = "ibex-devops-project-sept-2025-104-ec2-role"
  }
  + tags_all             = {
    + "Environment" = "dev"
    + "ManagedBy"  = "Terraform"
    + "Name"        = "ibex-devops-project-sept-2025-104-ec2-role"
  }
  + unique_id            = (known after apply)

  + inline_policy (known after apply)
}

# aws_iam_role_policy.ec2_s3_policy will be created
+ resource "aws_iam_role_policy" "ec2_s3_policy" {
  + id          = (known after apply)
  + name        = "ibex-devops-project-sept-2025-104-ec2-s3-policy"

```

```

+ name_prefix = (known after apply)
+ policy      = (known after apply)
+ role        = (known after apply)
}

# aws_iam_user.cicd_user[0] will be created
+ resource "aws_iam_user" "cicd_user" {
  + arn          = (known after apply)
  + force_destroy = false
  + id           = (known after apply)
  + name         = "ibex-devops-project-sept-2025-104-cicd-user"
  + path         = "/"
  + tags         = {
    + "Environment" = "dev"
    + "ManagedBy"  = "Terraform"
    + "Name"        = "ibex-devops-project-sept-2025-104-cicd-user"
  }
  + tags_all     = {
    + "Environment" = "dev"
    + "ManagedBy"  = "Terraform"
    + "Name"        = "ibex-devops-project-sept-2025-104-cicd-user"
  }
  + unique_id    = (known after apply)
}

# aws_iam_user_policy.cicd_policy[0] will be created
+ resource "aws_iam_user_policy" "cicd_policy" {
  + id          = (known after apply)
  + name        = "ibex-devops-project-sept-2025-104-cicd-policy"
  + name_prefix = (known after apply)
  + policy      = jsonencode(
    {
      + Statement = [
        + {
          + Action = [
            + "ec2:StartInstances",
            + "ec2:StopInstances",
            + "ec2:RunInstances",
            + "ec2:TerminateInstances",
            + "ec2:DescribeInstances",
            + "ec2:DescribeInstanceStatus",
          ]
          + Condition = {

```

```

        + StringEquals = {
            + "ec2:ResourceTag/ManagedBy" = "Terraform"
        }
    }
    + Effect      = "Allow"
    + Resource    = "*"
    + Sid        = "EC2Management"
},
+ {
    + Action      = [
        + "s3:CreateBucket",
        + "s3:DeleteBucket",
        + "s3:ListBucket",
        + "s3:GetObject",
        + "s3:PutObject",
        + "s3:DeleteObject",
    ]
    + Effect      = "Allow"
    + Resource    = [
        + "arn:aws:s3:::ibex-devops-project-sept-2025-104",
        + "arn:aws:s3:::ibex-devops-project-sept-2025-104/*",
    ]
    + Sid         = "S3Management"
},
+ {
    + Action      = [
        + "ec2:CreateSecurityGroup",
        + "ec2:DeleteSecurityGroup",
        + "ec2:AuthorizeSecurityGroupIngress",
        + "ec2:AuthorizeSecurityGroupEgress",
        + "ec2:RevokeSecurityGroupIngress",
        + "ec2:RevokeSecurityGroupEgress",
        + "ec2:DescribeSecurityGroups",
    ]
    + Effect      = "Allow"
    + Resource    = "*"
    + Sid         = "SecurityGroupManagement"
},
    ]
+ Version      = "2012-10-17"
}
)
+ user         = "ibex-devops-project-sept-2025-104-cicd-user"

```

```

}

# aws_instance.docker_host will be created
+ resource "aws_instance" "docker_host" {
  + ami                    = "ami-01810692271ee44a4"
  + arn                   = (known after apply)
  + associate_public_ip_address = (known after apply)
  + availability_zone       = (known after apply)
  + cpu_core_count          = (known after apply)
  + cpu_threads_per_core    = (known after apply)
  + disable_api_stop        = (known after apply)
  + disable_api_termination = (known after apply)
  + ebs_optimized           = (known after apply)
  + enable_primary_ipv6     = (known after apply)
  + get_password_data       = false
  + host_id                 = (known after apply)
  + host_resource_group_arn = (known after apply)
  + iam_instance_profile    =
"ibex-devops-project-sept-2025-104-ec2-profile"
  + id                     = (known after apply)
  + instance_initiated_shutdown_behavior = (known after apply)
  + instance_lifecycle     = (known after apply)
  + instance_state         = (known after apply)
  + instance_type          = "t2.micro"
  + ipv6_address_count      = (known after apply)
  + ipv6_addresses         = (known after apply)
  + key_name                = "ibex-devops-project-sept-2025-104-key"
  + monitoring              = (known after apply)
  + outpost_arn             = (known after apply)
  + password_data          = (known after apply)
  + placement_group        = (known after apply)
  + placement_partition_number = (known after apply)
  + primary_network_interface_id = (known after apply)
  + private_dns             = (known after apply)
  + private_ip              = (known after apply)
  + public_dns              = (known after apply)
  + public_ip               = (known after apply)
  + secondary_private_ips   = (known after apply)
  + security_groups         = (known after apply)
  + source_dest_check       = true
  + spot_instance_request_id = (known after apply)
  + subnet_id               = (known after apply)
  + tags                    = {

```



```

    + "Environment" = "dev"
    + "ManagedBy"   = "Terraform"
    + "Name"         = "ibex-devops-project-sept-2025-104-docker-host"
  }
+ tags_all = {
    + "Environment" = "dev"
    + "ManagedBy"   = "Terraform"
    + "Name"         = "ibex-devops-project-sept-2025-104-docker-host"
  }
+ tenancy = (known after apply)
+ user_data = (known after apply)
+ user_data_base64 = (known after apply)
+ user_data_replace_on_change = false
+ vpc_security_group_ids = (known after apply)

+ capacity_reservation_specification (known after apply)

+ cpu_options (known after apply)

+ ebs_block_device (known after apply)

+ 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 {
    + delete_on_termination = true
    + device_name           = (known after apply)
    + encrypted             = true
    + iops                  = (known after apply)
    + kms_key_id            = (known after apply)
    + tags_all              = (known after apply)
    + throughput            = (known after apply)
  }

```

```

        + volume_id          = (known after apply)
        + volume_size        = 20
        + volume_type        = "gp3"
    }
}

# aws_key_pair.ec2_key_pair will be created
+ resource "aws_key_pair" "ec2_key_pair" {
    + arn                = (known after apply)
    + fingerprint       = (known after apply)
    + id                = (known after apply)
    + key_name           = "ibex-devops-project-sept-2025-104-key"
    + key_name_prefix    = (known after apply)
    + key_pair_id        = (known after apply)
    + key_type           = (known after apply)
    + public_key         = (known after apply)
    + tags               = {
        + "Environment" = "dev"
        + "ManagedBy"  = "Terraform"
        + "Name"        = "ibex-devops-project-sept-2025-104-keypair"
    }
    + tags_all           = {
        + "Environment" = "dev"
        + "ManagedBy"  = "Terraform"
        + "Name"        = "ibex-devops-project-sept-2025-104-keypair"
    }
}

# aws_s3_bucket.artifacts will be created
+ resource "aws_s3_bucket" "artifacts" {
    + acceleration_status = (known after apply)
    + acl                 = (known after apply)
    + arn                 = (known after apply)
    + bucket              = "ibex-devops-project-sept-2025-104"
    + bucket_domain_name  = (known after apply)
    + bucket_prefix       = (known after apply)
    + bucket_regional_domain_name = (known after apply)
    + force_destroy       = false
    + hosted_zone_id      = (known after apply)
    + id                  = (known after apply)
    + object_lock_enabled = (known after apply)
    + policy               = (known after apply)
    + region              = (known after apply)
}

```

```

+ request_payer          = (known after apply)
+ tags                   = {
  + "Environment" = "dev"
  + "ManagedBy"  = "Terraform"
  + "Name"        = "ibex-devops-project-sept-2025-104-artifacts"
}
+ tags_all               = {
  + "Environment" = "dev"
  + "ManagedBy"  = "Terraform"
  + "Name"        = "ibex-devops-project-sept-2025-104-artifacts"
}
+ website_domain         = (known after apply)
+ website_endpoint       = (known after apply)

+ cors_rule (known after apply)

+ grant (known after apply)

+ lifecycle_rule (known after apply)

+ logging (known after apply)

+ object_lock_configuration (known after apply)

+ replication_configuration (known after apply)

+ server_side_encryption_configuration (known after apply)

+ versioning (known after apply)

+ website (known after apply)
}

# aws_s3_bucket_public_access_block.artifacts will be created
+ resource "aws_s3_bucket_public_access_block" "artifacts" {
  + block_public_acls      = true
  + block_public_policy    = true
  + bucket                 = (known after apply)
  + id                     = (known after apply)
  + ignore_public_acls     = true
  + restrict_public_buckets = true
}

```

```

# aws_s3_bucket_versioning.artifacts will be created
+ resource "aws_s3_bucket_versioning" "artifacts" {
  + bucket = (known after apply)
  + id     = (known after apply)

  + versioning_configuration {
    + mfa_delete = (known after apply)
    + status     = "Enabled"
  }
}

# aws_security_group.ec2_sg will be created
+ resource "aws_security_group" "ec2_sg" {
  + arn                = (known after apply)
  + description        = "Security group for EC2 instance"
  + egress              = [
    + {
      + cidr_blocks      = [
        + "0.0.0.0/0",
      ]
      + description      = "Allow all outbound traffic"
      + from_port        = 0
      + ipv6_cidr_blocks = []
      + prefix_list_ids  = []
      + protocol         = "-1"
      + security_groups  = []
      + self              = false
      + to_port          = 0
    },
  ]
  + id                 = (known after apply)
  + ingress            = [
    + {
      + cidr_blocks      = [
        + "0.0.0.0/0",
      ]
      + description      = "HTTP from anywhere"
      + from_port        = 80
      + ipv6_cidr_blocks = []
      + prefix_list_ids  = []
      + protocol         = "tcp"
      + security_groups  = []
      + self              = false
    }
  ]
}

```

```

        + to_port          = 80
      },
      + {
        + cidr_blocks      = [
          + "0.0.0.0/0",
        ]
        + description      = "SSH from allowed IPs"
        + from_port        = 22
        + ipv6_cidr_blocks = []
        + prefix_list_ids  = []
        + protocol          = "tcp"
        + security_groups   = []
        + self              = false
        + to_port           = 22
      },
    ]
    + name                = "ibex-devops-project-sept-2025-104-ec2-sg"
    + name_prefix          = (known after apply)
    + owner_id             = (known after apply)
    + revoke_rules_on_delete = false
    + tags                 = {
      + "Environment" = "dev"
      + "ManagedBy"  = "Terraform"
      + "Name"        = "ibex-devops-project-sept-2025-104-ec2-sg"
    }
    + tags_all            = {
      + "Environment" = "dev"
      + "ManagedBy"  = "Terraform"
      + "Name"        = "ibex-devops-project-sept-2025-104-ec2-sg"
    }
    + vpc_id              = "vpc-022ada8203ec3ef0a"
  }
}

```

local_file.private_key will be created

```

+ resource "local_file" "private_key" {
  + content                = (sensitive value)
  + content_base64sha256  = (known after apply)
  + content_base64sha512 = (known after apply)
  + content_md5           = (known after apply)
  + content_sha1          = (known after apply)
  + content_sha256        = (known after apply)
  + content_sha512        = (known after apply)
  + directory_permission = "0777"
}

```

```

+ file_permission      = "0400"
+ filename             = "./keys/ibex-devops-project-sept-2025-104-key.pem"
+ id                   = (known after apply)
}

# tls_private_key.ec2_ssh_key will be created
+ resource "tls_private_key" "ec2_ssh_key" {
  + algorithm           = "RSA"
  + ecdsa_curve         = "P224"
  + id                  = (known after apply)
  + private_key_openssh = (sensitive value)
  + private_key_pem     = (sensitive value)
  + private_key_pem_pkcs8 = (sensitive value)
  + public_key_fingerprint_md5 = (known after apply)
  + public_key_fingerprint_sha256 = (known after apply)
  + public_key_openssh   = (known after apply)
  + public_key_pem      = (known after apply)
  + rsa_bits             = 4096
}

```

Plan: 14 to add, 0 to change, 0 to destroy.

Changes to Outputs:

```

+ application_url      = (known after apply)
+ ec2_public_dns       = (known after apply)
+ ec2_public_ip       = (known after apply)
+ iam_role_arn         = (known after apply)
+ iam_user_access_key  = (sensitive value)
+ iam_user_secret_key  = (sensitive value)
+ s3_bucket_arn       = (known after apply)
+ s3_bucket_name       = (known after apply)
+ security_group_id    = (known after apply)
+ ssh_connection_command = (known after apply)
+ ssh_private_key_path = "./keys/ibex-devops-project-sept-2025-104-key.pem"

```

Saved the plan to: tfplan

To perform exactly these actions, run the following **command** to apply:

```
terraform apply "tfplan"
Continue with deployment? (yes/no) yes
```

```
==> terraform apply
tls_private_key.ec2_ssh_key: Creating...
aws_iam_user.cicd_user[0]: Creating...
aws_iam_role.ec2_role: Creating...
aws_s3_bucket.artifacts: Creating...
aws_security_group.ec2_sg: Creating...
aws_iam_user.cicd_user[0]: Creation complete after 0s
[id=ibex-devops-project-sept-2025-104-cicd-user]
aws_iam_user_policy.cicd_policy[0]: Creating...
aws_iam_access_key.cicd_user[0]: Creating...
aws_iam_role.ec2_role: Creation complete after 1s
[id=ibex-devops-project-sept-2025-104-ec2-role]
aws_iam_access_key.cicd_user[0]: Creation complete after 1s [id=AKIA2FGJ6YNGKSRLT7OZ]
aws_iam_instance_profile.ec2_profile: Creating...
aws_iam_user_policy.cicd_policy[0]: Creation complete after 1s
[id=ibex-devops-project-sept-2025-104-cicd-user:ibex-devops-project-sept-2025-104-cicd-policy]
aws_s3_bucket.artifacts: Creation complete after 1s
[id=ibex-devops-project-sept-2025-104]
aws_s3_bucket_public_access_block.artifacts: Creating...
aws_s3_bucket_versioning.artifacts: Creating...
aws_iam_role_policy.ec2_s3_policy: Creating...
aws_s3_bucket_public_access_block.artifacts: Creation complete after 0s
[id=ibex-devops-project-sept-2025-104]
aws_iam_role_policy.ec2_s3_policy: Creation complete after 0s
[id=ibex-devops-project-sept-2025-104-ec2-role:ibex-devops-project-sept-2025-104-ec2-s3-policy]
tls_private_key.ec2_ssh_key: Creation complete after 2s
[id=63ac513c0055f942f2274804f2328b0191c13265]
```

```
aws_key_pair.ec2_key_pair: Creating...
local_file.private_key: Creating...
local_file.private_key: Creation complete after 0s
[id=d0891342908c0c130f0f3e8230ef42a2cf2206ab]
aws_key_pair.ec2_key_pair: Creation complete after 0s
[id=ibex-devops-project-sept-2025-104-key]
aws_security_group.ec2_sg: Creation complete after 3s [id=sg-0d1b8a0c032c0e255]
aws_s3_bucket_versioning.artifacts: Creation complete after 2s
[id=ibex-devops-project-sept-2025-104]
aws_iam_instance_profile.ec2_profile: Creation complete after 5s
[id=ibex-devops-project-sept-2025-104-ec2-profile]
aws_instance.docker_host: Creating...
aws_instance.docker_host: Still creating... [00m10s elapsed]
aws_instance.docker_host: Still creating... [00m20s elapsed]
aws_instance.docker_host: Still creating... [00m30s elapsed]
aws_instance.docker_host: Provisioning with 'local-exec'...
aws_instance.docker_host (local-exec): Executing: ["/bin/sh" "-c" "echo 52.14.7.182 >
instance_ip.txt"]
aws_instance.docker_host: Creation complete after 34s [id=i-06cbb64d2d97f1ac7]
```

Apply complete! Resources: 14 added, 0 changed, 0 destroyed.

Outputs:

```
application_url = "http://52.14.7.182"
ec2_public_dns = "ec2-52-14-7-182.us-east-2.compute.amazonaws.com"
ec2_public_ip = "52.14.7.182"
iam_role_arn =
"arn:aws:iam::698355729228:role/ibex-devops-project-sept-2025-104-ec2-role"
iam_user_access_key = <sensitive>
iam_user_secret_key = <sensitive>
s3_bucket_arn = "arn:aws:s3:::ibex-devops-project-sept-2025-104"
s3_bucket_name = "ibex-devops-project-sept-2025-104"
security_group_id = "sg-0d1b8a0c032c0e255"
ssh_connection_command = "ssh -i ./keys/ibex-devops-project-sept-2025-104-key.pem
ec2-user@52.14.7.182"
ssh_private_key_path = "./keys/ibex-devops-project-sept-2025-104-key.pem"
✔ Deployment complete!
Application URL: http://52.14.7.182
```



```
ubuntu@ip-172-31-9-179:~/ibex-devops-task$ curl http://52.14.7.182
```

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <title>Ibex DevOps Demo Application</title>
```

```
  <style>
```

```
    * {
```

```
      margin: 0;
```

```
      padding: 0;
```

```
      box-sizing: border-box;
```

```
    }
```

```
    body {
```

```
      font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto, Oxygen,
```

```
Ubuntu, Cantarell, sans-serif;
    background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
    min-height: 100vh;
    display: flex;
    align-items: center;
    justify-content: center;
    color: white;
}

.container {
    text-align: center;
    padding: 2rem;
    background: rgba(255, 255, 255, 0.1);
    border-radius: 20px;
    backdrop-filter: blur(10px);
    box-shadow: 0 8px 32px 0 rgba(31, 38, 135, 0.37);
    max-width: 600px;
    margin: 20px;
}

h1 {
    font-size: 2.5rem;
    margin-bottom: 1rem;
    text-shadow: 2px 2px 4px rgba(0, 0, 0, 0.2);
}

.logo {
    width: 100px;
    height: 100px;
    margin: 0 auto 2rem;
    background: white;
    border-radius: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
    font-size: 2rem;
    box-shadow: 0 4px 15px rgba(0, 0, 0, 0.2);
}

.info {
    background: rgba(255, 255, 255, 0.2);
    border-radius: 10px;
    padding: 1.5rem;
```

```
        margin: 2rem 0;
    }

    .info h2 {
        margin-bottom: 1rem;
        font-size: 1.5rem;
    }

    .info-grid {
        display: grid;
        grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
        gap: 1rem;
        margin-top: 1rem;
    }

    .info-item {
        background: rgba(255, 255, 255, 0.1);
        padding: 1rem;
        border-radius: 8px;
        transition: transform 0.3s ease;
    }

    .info-item:hover {
        transform: translateY(-5px);
        background: rgba(255, 255, 255, 0.2);
    }

    .label {
        font-weight: bold;
        display: block;
        margin-bottom: 0.5rem;
        opacity: 0.9;
    }

    .value {
        font-size: 1.1rem;
    }

    .status {
        display: inline-block;
        padding: 0.5rem 1rem;
        background: #10b981;
        border-radius: 20px;
    }
```

```
    margin: 1rem 0;
    font-weight: bold;
    box-shadow: 0 2px 10px rgba(16, 185, 129, 0.3);
  }

.tech-stack {
  display: flex;
  justify-content: center;
  gap: 1rem;
  flex-wrap: wrap;
  margin-top: 2rem;
}

.tech-badge {
  background: rgba(255, 255, 255, 0.2);
  padding: 0.5rem 1rem;
  border-radius: 15px;
  font-size: 0.9rem;
  transition: all 0.3s ease;
}

.tech-badge:hover {
  background: rgba(255, 255, 255, 0.3);
  transform: scale(1.05);
}

.footer {
  margin-top: 2rem;
  opacity: 0.8;
  font-size: 0.9rem;
}

@keyframes pulse {
  0% {
    box-shadow: 0 0 0 0 rgba(16, 185, 129, 0.7);
  }
  70% {
    box-shadow: 0 0 0 10px rgba(16, 185, 129, 0);
  }
  100% {
    box-shadow: 0 0 0 0 rgba(16, 185, 129, 0);
  }
}
```

```

        .status {
            animation: pulse 2s infinite;
        }
    </style>
</head>
<body>
    <div class="container">
        <div class="logo"> Application Running</div>

        <div class="info">
            <h2>Deployment Information</h2>
            <div class="info-grid">
                <div class="info-item">
                    <span class="label">Environment</span>
                    <span class="value" id="environment">Production</span>
                </div>
                <div class="info-item">
                    <span class="label">Version</span>
                    <span class="value" id="version">v1.0.0</span>
                </div>
                <div class="info-item">
                    <span class="label">Deployment Time</span>
                    <span class="value" id="deployment-time">-</span>
                </div>
                <div class="info-item">
                    <span class="label">Container</span>
                    <span class="value">Docker/Nginx</span>
                </div>
            </div>
        </div>

        <div class="tech-stack">
            <span class="tech-badge">AWS EC2</span>
            <span class="tech-badge">Terraform</span>
            <span class="tech-badge">Docker</span>
            <span class="tech-badge">Jenkins/GitLab CI</span>
            <span class="tech-badge">S3</span>
        </div>
    </div>
</body>
</html>

```

```
        <span class="tech-badge">Nginx</span>
    </div>

    <div class="footer">
        <p>Successfully deployed via CI/CD Pipeline</p>
        <p style="margin-top: 0.5rem; opacity: 0.7;">(c) 2024 Ibex DevOps Task
Submission</p>
    </div>
</div>

<script>
    // Set deployment time
    document.getElementById('deployment-time').textContent = new
Date().toLocaleString();

    // Get environment from URL parameters or default
    const urlParams = new URLSearchParams(window.location.search);
    const env = urlParams.get('env') || 'Production';
    document.getElementById('environment').textContent = env;

    // Simulate version from build
    const version = `v1.0.${Math.floor(Math.random() * 100)}`;
    document.getElementById('version').textContent = version;
</script>
</body>
</html>
```

```
ubuntu@ip-172-31-9-179:~/ibex-devops-task$ echo "End of Simulation"
End of Simulation
ubuntu@ip-172-31-9-179:~/ibex-devops-task$
```