

AWS PROJECT

STEP1: TO CREATE An ACM (AWS CERTIFICATE MANAGER)

TO HOST THE APPLICATION SECURELY (WE NEED A CERTIFICATE MANAGER) – LICENCE.

We get the certificate from ACM.

The screenshot shows the AWS Certificate Manager (ACM) interface. At the top, there is a navigation bar with tabs like 'Certificates' and 'Request certificate'. Below this, a main panel displays 'Certificates (1)' with a single item. On the right side of the main panel, there are buttons for 'Import' and 'Request'. A yellow oval highlights the 'Request' button. The URL in the browser is 'us-east-1.console.aws.amazon.com/acm/home?region=us-east-1#/certificates/request'. The browser's address bar also shows 'Certificate Manager | us-east-1'. The bottom of the screen shows the Windows taskbar with various pinned icons and the date/time '15-10-2023 12:35'.

Click on request

The screenshot shows the AWS Certificate Manager (ACM) interface. A success message at the top states: "Successfully deleted certificate with ID b05bc8db-7f3a-4810-b9de-74f06690d87c. You have successfully deleted the certificate with ID b05bc8db-7f3a-4810-b9de-74f06690d87c from your account in AWS Certificate Manager." Below this, the "Request public certificate" section is visible, showing a "Domain names" input field containing "*.jaleela.ir". The "Validation method" section has "DNS validation - recommended" selected. The bottom status bar shows the date and time as 15-10-2023.

Create a domain for a application.

The screenshot shows the "Request certificate" step in the AWS Certificate Manager (ACM) console. It displays fields for "Domain names" (*.jaleela.ir), "Validation method" (selected as "DNS validation - recommended"), and "Key algorithm" (selected as "RSA 2048"). The bottom status bar shows the date and time as 15-10-2023.

Choose the dns validation and key algorithm.

Certificate Manager | us-east-1 | Microsoft Teams Classic

Customer Authent... Google Gmail Load balancers YouTube Maps News Translate Sign in [Jenkins] Network interfaces...

Elastic Kubernetes Service IAM EC2 VPC Lambda S3 CloudFormation Route 53

AWS Certificate Manager (ACM)

List certificates Request certificate Import certificate AWS Private CA

Key algorithm RSA 2048
RSA is the most widely used key type.

ECDSA P 256 Equivalent in cryptographic strength to RSA 3072.

ECDSA P 384 Equivalent in cryptographic strength to RSA 7680.

Tags Info To help you manage your certificates, you can optionally assign your own metadata to each resource in the form of tags.

No tags associated with this resource.

Add tag You can add 50 more tag(s).

Cancel Previous Request

Click on request for a certificate.

Certificate | ec2fullacc | Load balan... | Instances | VPC Consol... | Route 53 | Dashboard | WordPress | WordPress | production | +

Customer Authent... Google Gmail Load balancers YouTube Maps News Translate Sign in [Jenkins] Network interfaces...

Elastic Kubernetes Service IAM EC2 VPC Lambda S3 CloudFormation Route 53

AWS Certificate Manager (ACM)

List certificates Request certificate Import certificate AWS Private CA

Successfully requested certificate with ID 3686f444-f545-472d-ae4c-6a23e6ec5a71
A certificate request with a status of pending validation has been created. Further action is needed to complete the validation and approval of the certificate.

Successfully deleted certificate with ID b05bc8db-7f3a-4810-b9de-74f06690d87c
You have successfully deleted the certificate with ID b05bc8db-7f3a-4810-b9de-74f06690d87c from your account in AWS Certificate Manager.

AWS Certificate Manager > Certificates > 3686f444-f545-472d-ae4c-6a23e6ec5a71

3686f444-f545-472d-ae4c-6a23e6ec5a71

Delete

Certificate status

Identifier	Status
3686f444-f545-472d-ae4c-6a23e6ec5a71	Issued

ARN arnawsacm:us-east-1:540686108349:certificate/3686f444-f545-472d-ae4c-6a23e6ec5a71

Type Amazon Issued

CloudShell Feedback

30°C Mostly clear

Certificate is created successfully.

The screenshot shows the AWS Certificate Manager (ACM) interface. A successfully issued certificate for the domain *.jaleela.in is displayed. The certificate details include:

- Identifier: 3686f444-f545-472d-ae4c-6a23e6ec5a71
- Status: Issued
- ARN: arn:aws:acm:us-east-1:540686108349:certificate/3686f444-f545-472d-ae4c-6a23e6ec5a71
- Type: Amazon Issued

Below this, a table lists the domain *.jaleela.in with its status as Success, renewal status as -, and CNAME record details:

Domain	Status	Renewal status	Type	CNAME name	CNAME value
*.jaleela.in	Success	-	CNAME	_170d1f3847e1e3b6abae6d549e98a7e4.jaleela.in.	_d4da1cb0d0e90cb31ff26ab773cadf7b.yxmqqtjrff.acm-validations.aws.

Details for the certificate show it is In use, No, with a serial number of 03:5a:04:2c:5:dd:15:8:a:9:d:3:5:h:8:-1:h:f:0:-7:, requested at October 15, 2023, 18:27:15, and has Renewal eligibility Ineligible.

After certificate manager is created successfully,then I create a record in a Route53.

The screenshot shows the AWS Route 53 DNS records creation page. A success message indicates the certificate was deleted successfully:

Successfully deleted certificate with ID b05bc8db-7f3a-4810-b9de-74f06690d87c from your account in AWS Certificate Manager.

The Create DNS records in Amazon Route 53 (0/1) section shows the following table:

Domain	Validation status	Type	CNAME name	CNAME value	Is domain in Route 53?
*.jaleela.in	Success	CNAME	_170d1f3847e1e3b6abae6d549e98a7e4.jaleela.in.	_d4da1cb0d0e90cb31ff26ab773cadf7b.yxmqqtjrff.acm-validations.aws.	Yes

At the bottom right, there are 'Cancel' and 'Create records' buttons.

Already this domain in route53.

STEP2:create a route53

The screenshot shows the AWS Route 53 dashboard. On the left, there's a navigation sidebar with options like Hosted zones, Health checks, IP-based routing, Traffic flow, Domains, Resolver, and VPCs. The main area has four main sections: 'DNS management' (1 Hosted zone), 'Traffic management' (Create policy button), 'Availability monitoring' (4 Health checks), and 'Domain registration' (Register domain button). Below these is a 'Register domain' section with a search bar and a note about domain names. At the bottom, there's a status bar showing 'ENG - AFG in 2 hours' and system icons.

Click on a hosted zone.

This screenshot shows the 'Hosted zones' page. It displays one hosted zone named 'jaleela.in'. The table includes columns for Hosted zone name, Type, Created by, Record count, Description, and Hosted zo...'. The 'Create hosted zone' button is highlighted in orange at the top right of the table.

Hosted zone name	Type	Created by	Record count	Description	Hosted zo...
jaleela.in	Public	Route 53	10	-	Z003925525...

Hosted zone is created.

This screenshot shows the 'Records' page for the 'jaleela.in' hosted zone. It lists two records: an NS record and an SOA record. To the right, a dropdown menu titled 'Value/Route traffic to' is open, showing several nameserver entries: ns-1795.awsdns-32.co.uk., ns-555.awsdns-05.net., ns-1105.awsdns-10.org., ns-180.awsdns-22.com., and ns-1795.awsdns-32.co.uk. a...'. A yellow circle highlights this dropdown menu.

To attach the nameserver from a godaddy domain. Domain is available in a ROUTE53.So we will create record inside the domain name based on the application.

STEP3:CREATE AN RDS INSTANCE

Create a subnet for a db instance(for high availability)

Subnet group details

Name
You won't be able to modify the name after your subnet group has been created.

Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

Description

VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#create-db-subnet-group:

Add subnets

Amazon RDS

Availability Zones
Choose the Availability Zones that include the subnets you want to add.

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.

For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades
You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases. [RDS User Guide](#) [Aurora User Guide](#)

Databases (1)

Group resources Actions

Create a database.

The screenshot shows the AWS RDS 'Launch a DB instance' wizard. The 'Standard create' option is selected. On the right, the MySQL engine details are shown, including its popularity and various features like support for up to 64 TiB and automated backups. The MySQL icon is highlighted with a blue border.

Choose the dbengine-MySQL.

A yellow circle highlights the 'MySQL 8.0.33' dropdown under 'Engine Version'. The 'Show versions that support the Amazon RDS Optimized Writes' checkbox is also highlighted with a yellow circle. Below, the 'Templates' section shows 'Production', 'Dev/Test', and 'Free tier' options, with 'Free tier' selected. The 'Availability and durability' section shows 'Multi-AZ DB Cluster - new' as the deployment option.

Choose the mysql latest version.

The screenshot shows the AWS RDS MySQL instance configuration page. On the left, there's a sidebar with 'Master password' and 'Confirm master password'. Below it, the 'Instance configuration' section lists 'Amazon RDS Optimized Writes - new' and 'DB instance class' (selected as db.t3.micro). On the right, a panel titled 'MySQL' provides information about the database engine, including its popularity and various features like automated backup and cross-region replication.

Create a security group for RDS database.

The screenshot shows the 'Create Security Group' page in the AWS VPC console. It includes fields for 'Description' (set to 'securitygroupforRDS') and 'VPC Info' (set to 'vpc-0514d39da7e557921'). The 'Inbound rules' section contains three entries: one for HTTP (port 80), one for HTTPS (port 443), and one for MySQL/Aurora (port 3306).

Inbound rule for a securitygroup.

The screenshot shows the AWS VPC console with a success message: "Security group (sg-0ad370d9c432b5483 | RDSSecuritygroup) was created successfully". The details pane shows the security group name is "RDSSecuritygroup", owner is "540686108349", and it has 3 inbound rules and 1 outbound rule. The "Inbound rules" tab is selected.

SecuritygroupforRDS is created successfully.

The screenshot shows the "Create DB instance" wizard. The "DB subnet group" section is highlighted with a yellow circle, showing "prodsubnet" selected. The "Public access" section is also highlighted with a yellow circle, showing "No" selected. The "VPC security group (firewall)" section is highlighted with a large yellow circle, showing "Choose existing" selected with "RDSsecuritygroup" chosen from the dropdown. The "Existing VPC security groups" dropdown also contains "RDSsecuritygroup".

Assign all the configurations for a db instance.

Do the same configuration for a disaster db instance.



Prod db is created successfully.(along with a required subnet,security group,db name and VPC).



Disaster db is created successfully.

A detailed screenshot of the AWS RDS instance configuration page for the 'disasterdb' instance. The page is divided into sections: 'Summary', 'Connectivity & security', 'Monitoring', 'Logs & events', 'Configuration', 'Maintenance & backups', and 'Tags'. The 'Connectivity & security' tab is active. In this section, there is a yellow oval highlighting the 'Endpoint & port' row, which contains the endpoint 'proddb.cdtwj6gf01oa.us-east-1.rds.amazonaws.com' and port '3306'. Other sections show networking details like 'Availability Zone: us-east-1a' and security settings involving 'VPC security groups: RDSsecuritygroup (sg-0ad370d9c432b5483)' and 'Active' status.

Inside the db endpoints are available(for connecting ec2 to RDS(db instance)).

STEP4:CREATE AN IAM ROLE.

The screenshot shows the AWS IAM Roles page. The left sidebar has 'Identity and Access Management (IAM)' selected under 'Access management'. The main content area displays a table titled 'Roles (58) info' with columns for 'Role name', 'Trusted entities', and 'Last activity'. The table lists various AWS service roles and account-level roles. A search bar and pagination controls are at the top of the table. The bottom of the page includes standard browser navigation and status bars.

Role name	Trusted entities	Last activity
apilambdafunction-role-3xmqz7jq	AWS Service: lambda	-
aws-ec2-spot-fleet-tagging-role	AWS Service: spotfleet	-
aws-elasticbeanstalk-service-role	AWS Service: elasticbeanstalk	-
AWS-QuickSetup-StackSet-Local-AdministrationRole	AWS Service: cloudformation	-
AWS-QuickSetup-StackSet-Local-ExecutionRole	Account: 540686108349	-
AWSControlTowerAdmin	AWS Service: controlltower	-
AWSControlTowerCloudTrailRole	AWS Service: cloudtrail	-
AWSControlTowerConfigAggregatorRoleForOrganizations	AWS Service: config	-
AWSControlTowerStackSetRole	AWS Service: cloudformation	-

This screenshot is identical to the one above, but the 'Last activity' column for the 'AWSControlTowerAdmin' role shows '24 hours ago' instead of '-'. This indicates that a new entry was made to the table after the first screenshot was taken.

Click on create role.

The screenshot shows the AWS IAM console interface. A modal window titled "Create role | IAM | Global" is open, showing Step 3: Name, review, and create. In the "Trusted entity type" section, the "AWS service" option is selected. Below it, the "Use case" section shows "EC2" selected. The bottom of the screen shows a dark navigation bar with various icons and links.

To choose the EC2 service.

The screenshot shows the "Permissions policies" section of the IAM console. The "AdministratorAccess" policy is selected and highlighted. The table shows columns for Policy name, Type, and Description.

Policy name	Type	Description
AdministratorAccess	AWS managed - job function	Provides full access to AWS services an...

To give a full admin access to a ec2 instance.

The screenshot shows the "Role details" section of the IAM console. The role name is set to "ec2fullaccess". The "Step 1: Select trusted entities" section displays a JSON-based trust policy:

```
1 = []
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
6       ],
5       "Principal": [
4         "arn:aws:iam::123456789012:root"
5       ]
2     }
1   ]
```

The screenshot shows the AWS IAM console with a role named 'ec2fullaccess' created. The 'Permissions' tab is selected, displaying the ARN of the role. The role allows EC2 instances to call AWS services on behalf of the user.

Role is created for full access to a ec2 service.

STEP5:EC2 INSTANCE CREATION

Two launch the two instances for production and disaster recovery.

The screenshot shows the AWS Management Console with the EC2 service selected. The left sidebar has sections for Instances, Images, and Elastic Block Store. The main content area is titled 'Instances' and shows a table with one row: 'No instances'. A 'Launch instances' button is at the bottom. Below it is a modal window titled 'Select an instance'.

Launch the instance.

To create a security group for ec2-instance.

The screenshot shows the AWS Management Console with the VPC service selected. The left sidebar has sections for Egress-only internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, Peering connections, Security (with Network ACLs and Security groups selected), DNS firewall, and Network Firewall. The main content area is titled 'Security Groups (23)' and shows a table with 23 rows of security group details. A 'Create security group' button is at the top right. Below the table is a modal window titled 'Filter security groups'.

Click on security group.

The screenshot shows the AWS VPC Console with the 'Inbound rules' tab selected. There are four rules listed:

- SSH (TCP port 22) from Anywhere (0.0.0.0/0)
- HTTP (TCP port 80) from Anywhere (0.0.0.0/0)
- HTTPS (TCP port 443) from Anywhere (0.0.0.0/0)
- MySQL/Aurora (TCP port 3306) from Anywhere (0.0.0.0/0)

An 'Add rule' button is at the bottom left. A warning message at the bottom states: "Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only."

Inbound rules for ec2 security group.

The screenshot shows the AWS VPC console with a success message: "Security group (sg-0a9565070347e1732 | ec2securitygroupforwordpress) was created successfully".

The security group details are as follows:

Security group name: ec2securitygroupforwordpress	Security group ID: sg-0a9565070347e1732	Description: wordpressecsecuritygroup	VPC ID: vpc-0514d39da7e557921
Owner: 540686108349	Inbound rules count: 4 Permission entries	Outbound rules count: 1 Permission entry	

The 'Inbound rules' tab is selected, showing 4 entries. The interface includes standard AWS navigation and status bars at the top and bottom.

Security group is created successfully. Now I launch the instance.

Linux

aws

Mac

ubuntu®

Microsoft

Red Hat

SUSE

Browse more AMIs
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type
ami-0bb4c991fa89d4b9b (64-bit (x86)) / ami-a0445ce583184891 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs

Description
Amazon Linux 2 Kernel 5.10 AMI 2.0.20230926.0 x86_64 HVM gp2

Architecture AMI ID
64-bit (x86) ami-0bb4c991fa89d4b9b Verified provider

Instance type

Info

Instance type
t2.micro Free tier eligible
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

All generations

Compare instance types

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-0bb4c991fa89d4b9b

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) ~ 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or

Cancel Launch instance Review commands

Choose the linux os,ami,instance type.

Create

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](#) ▾

ec2securitygroupforwordpress sg-0a9565070347e1732 X
VPC: vpc-0514d39da7e557921

Compare security group rules

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

▼ Configure storage [Info](#) Advanced

1x GiB ▾ Root volume (Not encrypted)

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

Add new volume

0 x File systems

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-0bb4c991fa89d4b9b

Virtual server type (instance type)
t2.micro

Firewall (security group)
ec2securitygroupforwordpress

Storage (volumes)
1 volume(s) ~ 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or

Cancel **Launch instance** Review commands

Attach the security group,that we had created.

Inbound rules control the incoming traffic that's allowed to reach the instance.						
Security group rule ID	Type info	Protocol info	Port range info	Source info	Description - optional info	
sgr-048f3175c034ced94	SSH	TCP	22	Custom ▾	<input type="text" value="Q"/> <input type="text" value="0.0.0.0"/> <input type="button" value="Delete"/>	
sgr-00c776d913d0bcfdc	HTTP	TCP	80	Custom ▾	<input type="text" value="Q"/> <input type="text" value="0.0.0.0"/> <input type="button" value="Delete"/>	
sgr-0247fc9ac5864ceec	HTTPS	TCP	443	Custom ▾	<input type="text" value="Q"/> <input type="text" value="0.0.0.0"/> <input type="button" value="Delete"/>	

[Add rule](#)

⚠ Rules with source of 0.0.0.0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. [X](#)

Inbound rule of the security group of the instances.(ssh,http,https).

Advanced details Info

Purchasing option Info
 Request Spot Instances

Domain join directory Info
 Create new directory

IAM instance profile Info
 arn:awsiam::540686108349:instance-profile/ec2fullaccess Create new IAM profile

Hostname type Info

DNS Hostname Info
 Enable IP name IPv4 (A record) DNS requests
 Enable resource-based IPv4 (A record) DNS requests
 Enable resource-based IPv6 (AAAA record) DNS requests

Instance auto-recovery Info

Summary

Number of instances Info

Software Image (AMI)
 Amazon Linux 2 Kernel 5.10 AMI... [read more](#)
 ami-0bb4c991fa89d4b9b

Virtual server type (instance type)
 t2.micro

Firewall (security group)
 ec2Securitygroupforwordpress

Storage (volumes)
 1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or

[Cancel](#) Launch instance [Review commands](#)

To attach the ec2 full access IAM role to a ec2-server.

```
#!/bin/bash

yum install httpd php-mysql -y

amazon-linux-extras install -y php7.3

cd /var/www/html

echo "healthy" > healthy.html

wget https://wordpress.org/latest.tar.gz

tar -xzf latest.tar.gz

cp -r wordpress/* /var/www/html/

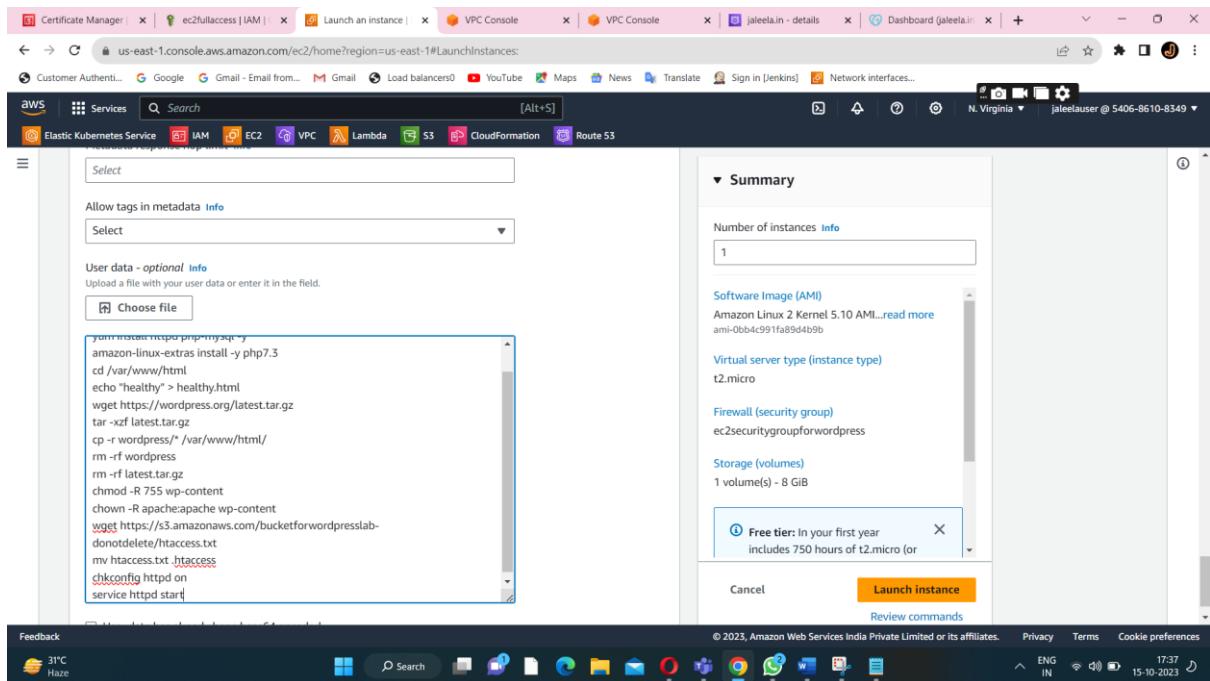
rm -rf wordpress
```

```

rm -rf latest.tar.gz
chmod -R 755 wp-content
chown -R apache:apache wp-content
wget https://s3.amazonaws.com/bucketforwordpresslab-donotdelete/.htaccess.txt
mv htaccess.txt .htaccess
chkconfig httpd on
service httpd start

```

attach the above Userdata.



Then launch 2 instances.

Number of instances [Info](#)

2

When launching more than 1 instance, consider [EC2 Auto Scaling](#).

Software Image (AMI)

Amazon Linux 2 Kernel 5.10 AMI...[read more](#)
ami-0bb4c991fa89d4b9b

Virtual server type (instance type)

t2.micro

Firewall (security group)

ec2securitygroupforwordpress

Storage (volumes)

1 volume(s) - 8 GiB

Cancel
Launch instance

[Review commands](#)

Instances (2/2) Info								
<input type="text"/> Find instance by attribute or tag (case-sensitive)		Instance state	Actions	Launch instances				
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	
<input checked="" type="checkbox"/> prodinstance	i-09ed9a2c1609ce591	Running	t2.micro	Initializing	No alarms	us-east-1c	ec2-18-212-203-	
<input checked="" type="checkbox"/> disasterinstance	i-09c24407fe3769e4c	Running	t2.micro	Initializing	No alarms	us-east-1c	ec2-18-209-51-2	

Two instances is launched successfully.

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with links like EC2 Dashboard, EC2 Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes, Feedback). The main area displays two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
prodinstance	i-09ed9a2c1609ce591	Running	t2.micro	Initializing	No alarms	us-east-1c	ec2-18-212-203-
disasterinstance	i-09c24407fe3769e4c	Running	t2.micro	Initializing	No alarms	us-east-1c	ec2-18-209-51-2

Below the table, there's a detailed view for the first instance:

Instance: i-09ed9a2c1609ce591 (prodinstance)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary: Instance ID - i-09ed9a2c1609ce591 (prodinstance); Public IPv4 address - 18.212.203.146 [open address]; Private IPv4 addresses - 172.31.24.153; IPv6 address - -; Instance state - Running.

At the bottom right, there are links for 2023, Amazon Web Services India Private Limited or its affiliates, Privacy, Terms, and Cookie preferences. The system status bar shows ENG IN, 17:41, and 15-10-2023.

Copy the public ip-address of a ec2-instnace.

The screenshot shows a WordPress setup page at 18.212.203.146/wp-admin/setup-config.php. The page has a large WordPress logo at the top. Below it, there's a box with the following text:

Welcome to WordPress. Before getting started, you will need to know the following items.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

This information is being used to create a `wp-config.php` file. If for any reason this automatic file creation does not work, do not worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`. Need more help? [Read the support article on wp-config.php](#).

In all likelihood, these items were supplied to you by your web host. If you do not have this information, then you will need to contact them before you can continue. If you are ready...

[Let's go!](#)

At the bottom right, there are links for 2023, Amazon Web Services India Private Limited or its affiliates, Privacy, Terms, and Cookie preferences. The system status bar shows ENG IN, 17:40, and 15-10-2023.

After I copy the public ip address and hit in a chrome.we will get a wordpress page.(for both the instances).

STEP6:CREATE A LOAD BALANCER

Choose the application load balancer

To handle the traffic (to route the traffic to an instance).

After creating a loadbalancer we have a dns name.

To attach the loadbalancer to a instances.

The screenshot shows the AWS Management Console interface for the EC2 service. The top navigation bar includes links for Certificate Manager, ec2fullaccess, Load balancer, VPC Console, VPC Console, Route 53 | Go!, Dashboard, WordPress, WordPress, and Network interfaces. The main menu on the left lists services like AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, Lifecycle Manager, Network & Security, Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces, Load Balancing, Load Balancers, Target Groups, Auto Scaling, and Auto Scaling Groups. The central content area is titled 'Load balancers' and contains a table with columns for Name, DNS name, State, VPC ID, Availability Zones, Type, and Date. A message at the bottom states 'No load balancers' and 'You don't have any load balancers in us-east-1'. At the bottom of the page, there are links for Feedback, Weather (31°C Haze), Search, and various system icons. The footer includes copyright information for 2023, privacy terms, cookie preferences, and a timestamp of 17:45 on 15-10-2023.

Create a load balancer to handle the traffic.

The screenshot shows the AWS Management Console with multiple tabs open at the top. The main window displays the 'Step 1 Create' wizard for creating a load balancer. The 'Classic Load Balancer - previous generation' option is selected. A detailed diagram illustrates the architecture: a 'CLB' (Classic Load Balancer) connects to four EC2 instances via 'HTTP', 'HTTPS', 'TCP', and 'SSL' ports. The 'Create' button is visible in the top right corner of the wizard window.

Click on a classic load balancer.

Security group for a load balancer

Create security group Info

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name Info

Name cannot be edited after creation.

Description Info

VPC Info

 X

Change the path healthy.html

Attach the necessary inbound-rule for a loadbalancer security group.

Security group is created successfully.

will be deprecated soon.

[EC2](#) > [Load balancers](#) > Create Classic Load Balancer

Create Classic Load Balancer [Info](#)

The Classic Load Balancer distributes incoming application traffic across multiple EC2 instance targets in multiple Availability Zones. This increases the fault tolerance of your applications. Elastic Load Balancing detects unhealthy instances and routes traffic only to healthy instances.

► How Classic Load Balancers work

Basic configuration

Load balancer name

Name must be unique within your AWS account and can't be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)

Scheme can't be changed after the load balancer is created.

 Internet-facing

--

Create a load balancer.

us-east-1a (use1-az1)

Subnet

IPv4 address

Assigned by AWS

us-east-1b (use1-az2)

Subnet

IPv4 address

Assigned by AWS

us-east-1c (use1-az4)

Subnet

Attached the subnet to a load balancer.

Listeners and routing Info

A listener is a process that checks for connection requests using the protocol and port you configure. The settings you define for a listener determine how the load balancer routes requests to its registered targets.

▼ Listener HTTP:80

Instance HTTP:80

[Remove](#)

Listener protocol Listener port

HTTP ▼ : 80
1-65535

Instance protocol Instance port

HTTP ▼ : 80
1-65535

[Add listener](#)

▼ Listener HTTP:80

Instance HTTP:80

[Remove](#)

Listener protocol Listener port

HTTP ▼ : 80
1-65535

Instance protocol Instance port

HTTP ▼ : 80
1-65535

[Add listener](#)

Health checks Info

Your load balancer automatically performs health checks to test the availability of all registered instances. Traffic is only routed to healthy instances, which is determined on their response to the health check.

Ping target

The health check ping is sent using the protocol and port you specify. If using HTTP/HTTPS protocol, you must also provide the destination path.

Ping protocol Ping port

HTTP ▼ : 80
1-65535

Ping path

/healthy.html

[► Advanced health check settings](#)

Change the path for healthy.html

Listener is used to route the traffic to a necessary instance.

Add instances

Select EC2 Instances to register to your load balancer. Requests will be routed to registered instances that meet the health check requirements. For maximum fault tolerance, we recommend maintaining approximately equivalent numbers of instances in each Availability Zone enabled for the load balancer. If demand on your instances changes, you can register or deregister instances without disrupting the flow of requests to your application. [Learn more](#)

VPC
vpc-0514d39da7e557921

Available instances (1/2)

Instance ID	Name	State	Security groups	Zone	Public IP
i-09ed9a2c1609ce591	prodinstance	Running	ec2securitygroupforwordpress	us-east-1c	18.212.20
i-09c24407fe3769e4c	disasterinstance	Running	ec2securitygroupforwordpress	us-east-1c	18.209.51

Cancel Confirm

Feedback

SH113 / NSK Salai Closed road

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences ENG IN 18:12 15-10-2023

Attach the instance to a loadbalancer.

Load balancer: prodloadbalancer

Instance ID	Name	Health status	Health status description
i-09ed9a2c1609ce591	prodinstance	In-service	Not applicable

Instance is should be in-service(instance is available)

Basic configuration Edit productionloadbalancer • Internet-facing	Network mapping Edit VPC vpc-0514d39da7e557921 default vpc • us-east-1a subnet-03179c60e1ce1f9e0 • us-east-1b subnet-023bed4412cc5d6c9 • us-east-1c subnet-0d3b9cacef8f3794a • us-east-1d subnet-05629e45600e357a9 • us-east-1e subnet-0bdf2bcf19116b5e0 • us-east-1f subnet-09df9e93639926d52	Security groups Edit • loadbalancersecuritygroup sg-055d568796fc74373	Listeners and routing Edit • HTTP:80
Health checks Edit HTTP:80/index.html • Timeout: 2 seconds • Interval: 5 seconds • Unhealthy threshold: 2 • Unhealthy threshold: 10	Instances Edit 1 instance added • 1 instance in us-east-1c	Attributes Edit • Cross-zone load balancing: On • Connection draining: On • Connection draining timeout: 300 seconds	Tags Edit None

Click on create load balancer.

The screenshot shows the AWS CloudFormation console with a successful stack creation. The stack name is 'Step 1 Create', and it has 1 resource in the 'CREATE_COMPLETE' state. The resources are 'VPC' and 'Route 53'. The VPC section shows a VPC ID of 'vpc-0514d39da7e557921' and 6 Availability Zones. The Route 53 section shows a hosted zone with an ARN of 'arn:aws:route53:::hostedzone/Z2QW3PQHJLXK3'. The status bar at the bottom indicates 'ENG IN' and the date '15-10-2023'.

Load balancer is created successfully.

Keep all the configuration has it is and create a load balancer.

Attach the necessary subnet for high availability.

Then choose the endpoint of DNS (Domain name of a load balancer).

The screenshot shows the WordPress 'Setup Configuration' page. It displays the following information:

- Welcome message: "Welcome to WordPress. Before getting started, you will need to know the following items."
- Items required:
 - Database name
 - Database username
 - Database password
 - Database host
 - Table prefix (if you want to run more than one WordPress in a single database)
- A note about wp-config.php: "This information is being used to create a wp-config.php file. If for any reason this automatic file creation does not work, do not worry. All this does is fill in the database information to a configuration file. You may also simply open wp-config-sample.php in a text editor, fill in your information, and save it as wp-config.php. Need more help? Read the support article on wp-config.php."
- A note about web host information: "In all likelihood, these items were supplied to you by your web host. If you do not have this information, then you will need to contact them before you can continue. If you are ready..."
- A blue 'Let's go!' button.

The status bar at the bottom indicates 'ENG IN' and the date '15-10-2023'.

Ping on the incognito-window.Got a same wordpress output in a ec2-instance.

DISASTER LOADBALANCER

The screenshot shows the AWS Classic Load Balancer console. The URL is <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateCLBWizard>. The page title is "Create Classic Load Balancer". A banner at the top says "Introducing the new Classic Load Balancer console experience" and "We've redesigned the Classic Load Balancer console to make it easier to use. By default, we'll always bring you to the new Classic Load Balancer experience. You can access the old console experience; however, it will be deprecated soon." Below the banner, the navigation path is "EC2 > Load balancers > Create Classic Load Balancer". The main content area has a section titled "How Classic Load Balancers work" and "Basic configuration". In the "Basic configuration" section, the "Load balancer name" field is set to "disasterloadbalancer". The "Scheme" dropdown is set to "Internet-facing". At the bottom of the configuration section, there are "Feedback" and "Cookie preferences" buttons, along with a weather widget showing "30°C Mostly sunny". The status bar at the bottom right shows "ENG IN 18:01 15-10-2023".

Add the same load balancer security group to a disaster instance

The screenshot shows the AWS Classic Load Balancer console. The URL is <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateCLBWizard>. The page title is "Create Classic Load Balancer". The navigation path is "EC2 > Load balancers > Create Classic Load Balancer". The current step is "Advanced health check settings". The main content area has a section titled "Instances (0)". It shows a table with columns: Instance ID, Name, State, and Security groups. A note says "No instances added". Below the table, there is an "Attributes" section with a note about specifying additional features at launch. A checkbox for "Enable cross-zone load balancing" is checked, with a note explaining its functionality. At the bottom of the attributes section, there is a note about creating the load balancer using the console. The status bar at the bottom right shows "ENG IN 18:02 15-10-2023".

Click on add instance to add the required instance.

The screenshot shows the AWS CloudFormation console with a stack named 'disasterloadbalancer' in progress. The 'Basic configuration' tab is active, displaying a VPC configuration with a specific VPC ID and subnet details. The 'Listeners and routing' tab shows a single listener for port 80. The 'Health checks' tab provides configuration for health checks, including timeout and interval settings.

The screenshot shows the WordPress setup configuration page. It displays a welcome message and a list of items needed for database configuration. Below this, it explains the creation of a wp-config.php file and provides instructions for manual configuration if automatic creation fails. A prominent 'Let's go!' button is at the bottom of the form.

Now I ping the DNS of a disasterloadbalancer got the same wordpress as a output.

Both the production and disaster recovery load balancer is created successfully.

STEP7:CONNECT AN EC2 INSTANCE TO A DB INSTANCE

Connct the production instance.

To connect the RDS (db instance) install the MYSQL on a production instance.

Commands:sudo su,yum install mysql

Endpoint of db instance is used to connect EC2 instance.

```
mysql Ver 15.1 Distrib 5.5.68-MariaDB, for Linux (x86_64) using readline 5.1
[ec2-user@ip-172-31-24-153 ~]$ mysql -h proddb.cdtwj6gf01oa.us-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.33 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| prod |
| sys |
+-----+
5 rows in set (0.00 sec)

MySQL [(none)]> use prod;
Database changed
MySQL [prod]> []
```

mysql -h proddb.cdtwj6gf01oa.us-east-1.rds.amazonaws.com -P 3306 -u admin -p(for connecting RDS instance and EC2-INSTANCE).

this command is used to connect ec2 instance to db instance.

Same db creation for disaster DB creation.

Using the security group,assign the name to a disaster db.

Database options

Initial database name [Info](#)

disaster

If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group [Info](#)

default.mysql8.0

Option group [Info](#)

default:mysql-8-0

Initial database name:disaster is need to provide because of then only database is created in MYSQL db instance.

```

aws | Services | Search [Alt+S]
Elastic Kubernetes Service IAM EC2 VPC Lambda S3 CloudFormation Route 53

mysql Ver 15.1 Distrib 5.5.68-MariaDB, for Linux (x86_64) using readline 5.1
[ec2-user@ip-172-31-17-109 ~]$ mysql -h disasterdb.cdtwj6gf01oa.us-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 8.0.33 source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database      |
+-----+
| disaster      |
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.01 sec)

MySQL [(none)]> use disaster;
Database changed
MySQL [disaster]> []

```

Same configuration,endpoint and security group is applicable for disaster db(then we connect ec2 to RDS (db instance)).

SECURITY GROUP FOR RDS(MYSQLDB)

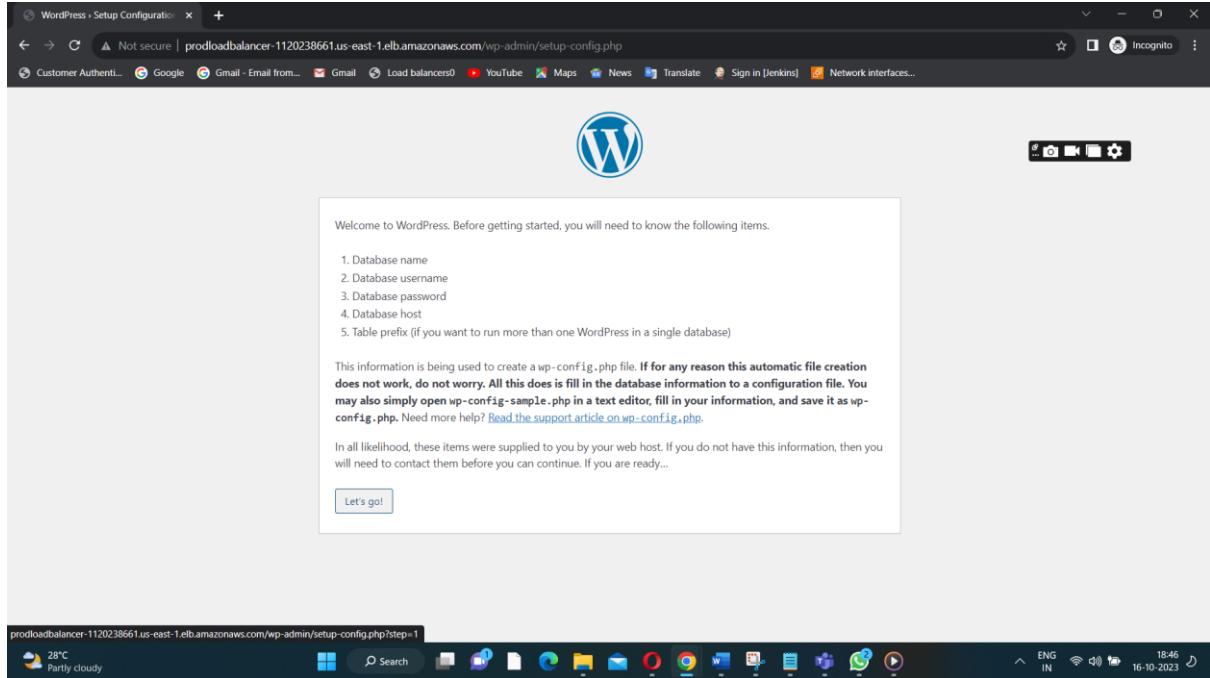
The screenshot shows the 'Edit inbound rules' section of the AWS VPC Security Groups interface. It displays a single rule for port 3306:

Security group rule ID	Type	Protocol	Port range	Source	Description
sgr-081bef994ee3d4ecd	MySQL/Aurora	TCP	3306	Custom (0.0.0.0/0)	- optional

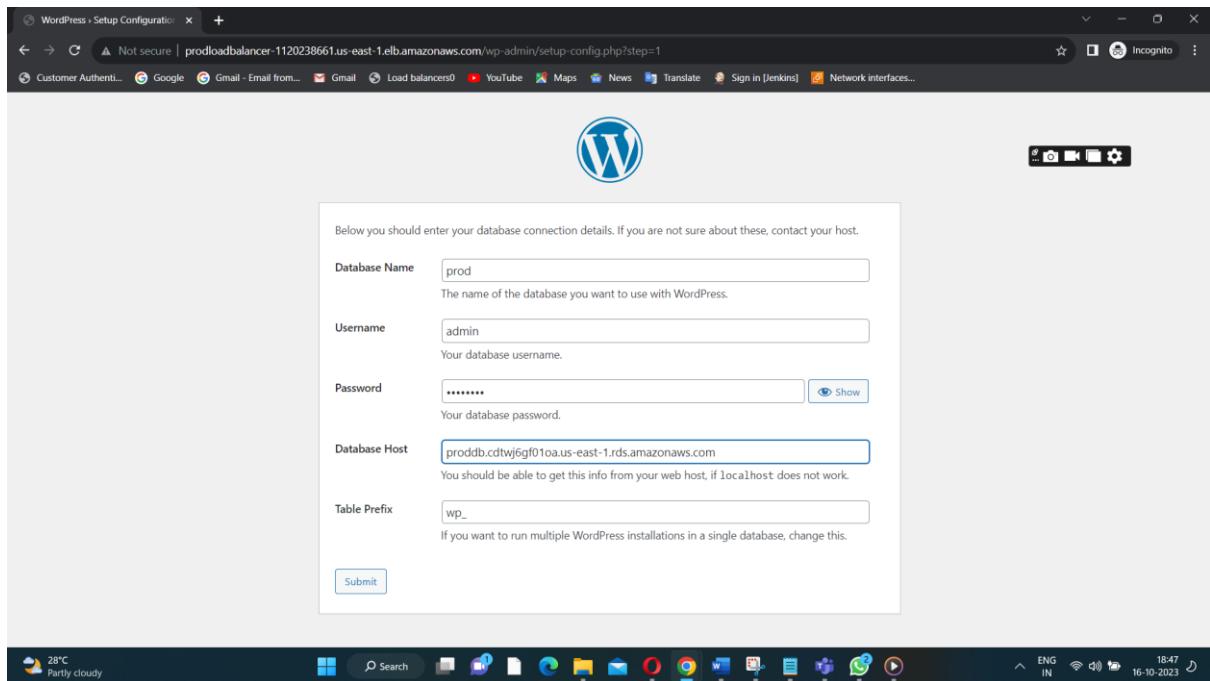
A warning message at the bottom states: "⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." There are 'Cancel', 'Preview changes', and 'Save rules' buttons at the bottom right.

**Security group for RDS INSTANCE should have a
MySQL/Aurora: port-number is 3306.**

STEP8:Connect the wordpress using db credentials.



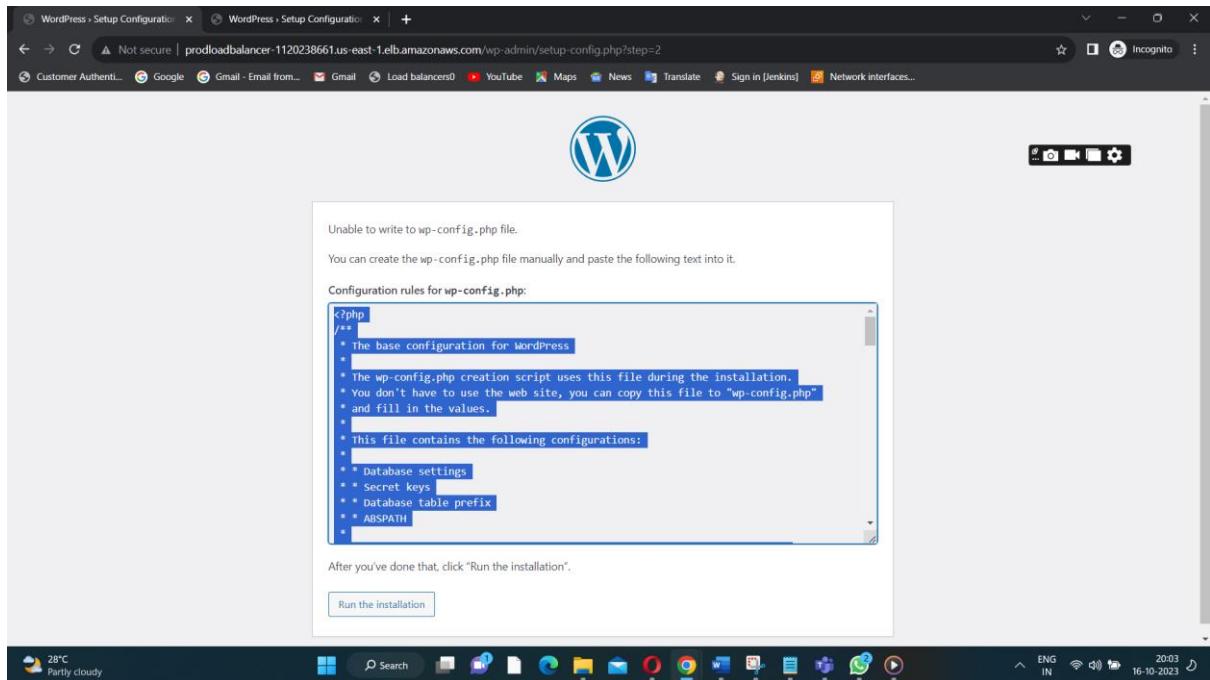
Click on let's go.



Provide all the proddb credentials.

Click on submit.

We got the configuration file.



Manually attach the config file in a wordpress.

```
mysql Ver 15.1 Distrib 5.5.68-MariaDB, for Linux (x86_64) using readline 5.1
[ec2-user@ip-172-31-24-153 ~]$ cd /var/www/html
[ec2-user@ip-172-31-24-153 html]$ ls
healthy.html license.txt wp-activate.php wp-blog-header.php wp-config-sample.php wp-cron.php wp-links-opml.php wp-login.php wp-settings.php wp-trackback.php
index.php readme.html wp-admin wp-comments-post.php wp-content wp-includes wp-load.php wp-mail.php wp-signup.php xmlrpc.php
[ec2-user@ip-172-31-24-153 html]$ 
```

Inside the cd /var/www/html path all the files are there.

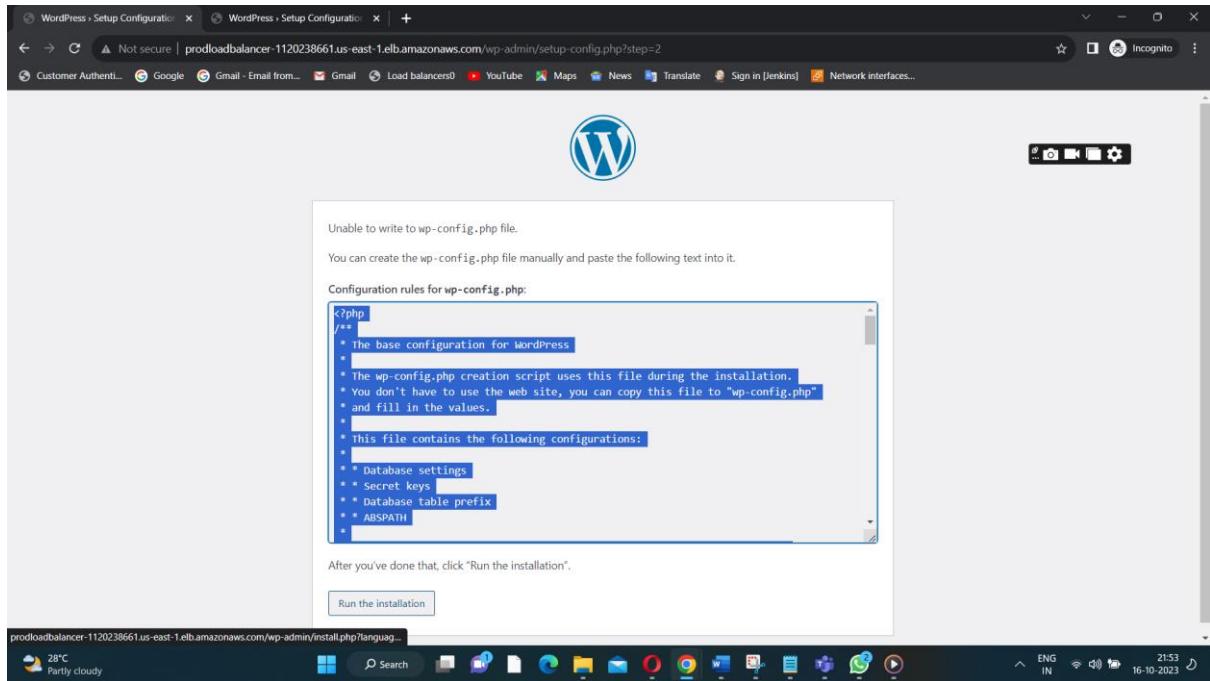
```
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the installation.
 * You don't have to use the web site, you can copy this file to "wp-config.php"
 * and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * Database settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/documentation/article/editing-wp-config-php/
 *
 * @package WordPress
 */

// ** Database settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'prod' );

/** Database username */
:wq![]
```

After entering the file inside the vi wp-config.php we can run the installation path.

Config the file for a software listener.



Click run the installation.

Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Do not worry, you can always change these settings later.

Site Title production

Username admin

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password prod1234  Hide

Weak

Important: You will need this password to log in. Please store it in a secure location.

Confirm Password Confirm use of weak password

Your Email prod43@gmail.com

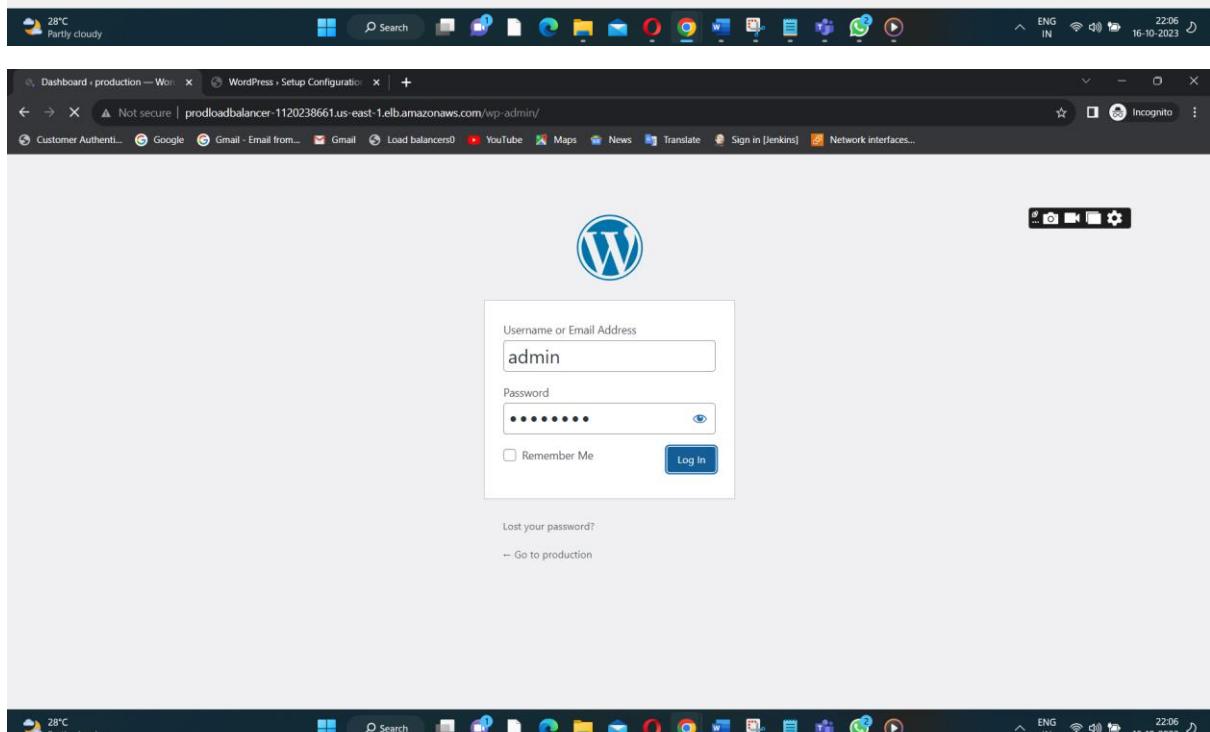
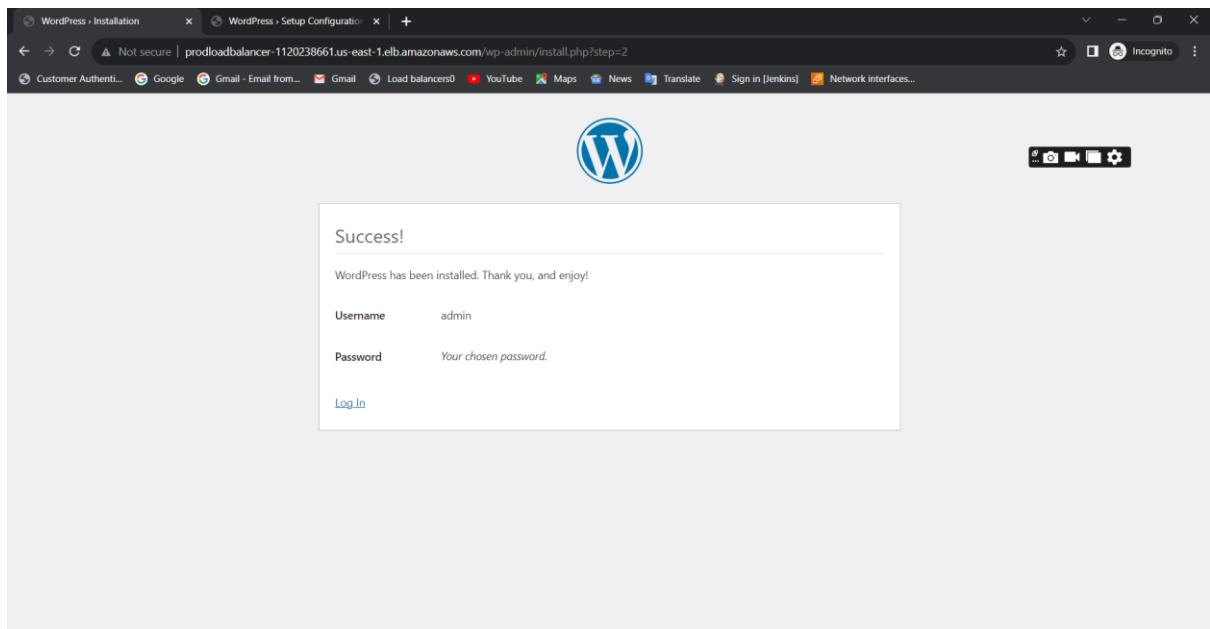
Double-check your email address before continuing.

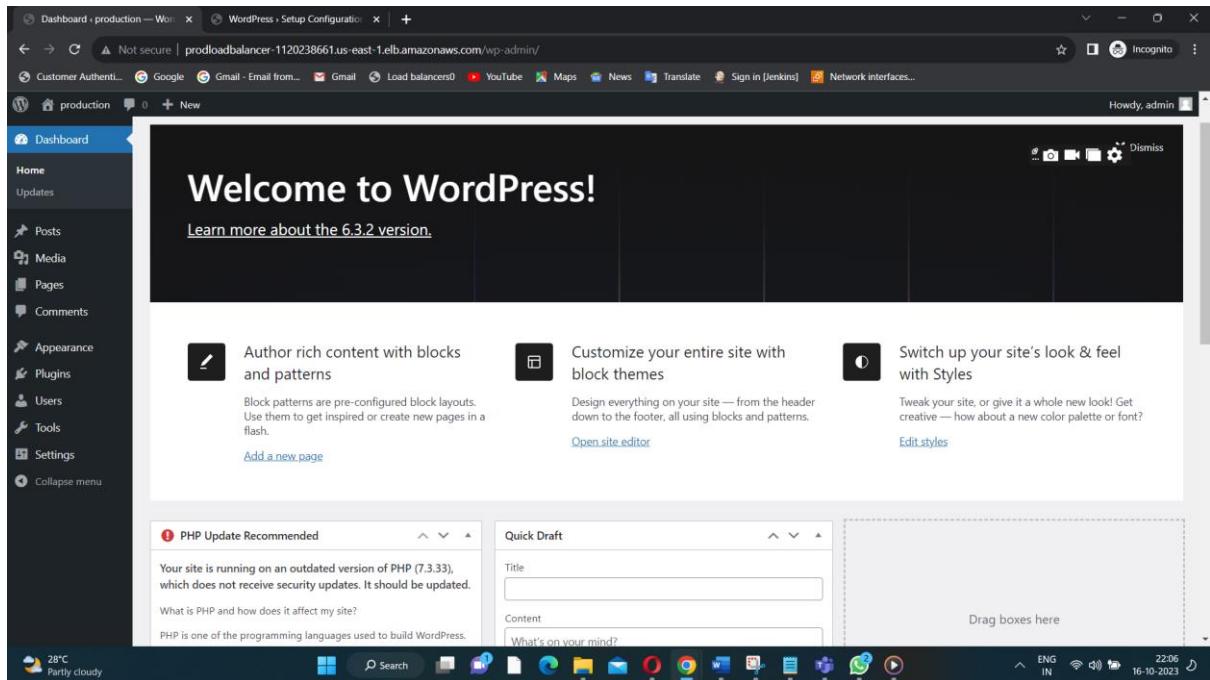
Search engine visibility Discourage search engines from indexing this site

It is up to search engines to honor this request.

[Install WordPress](#)

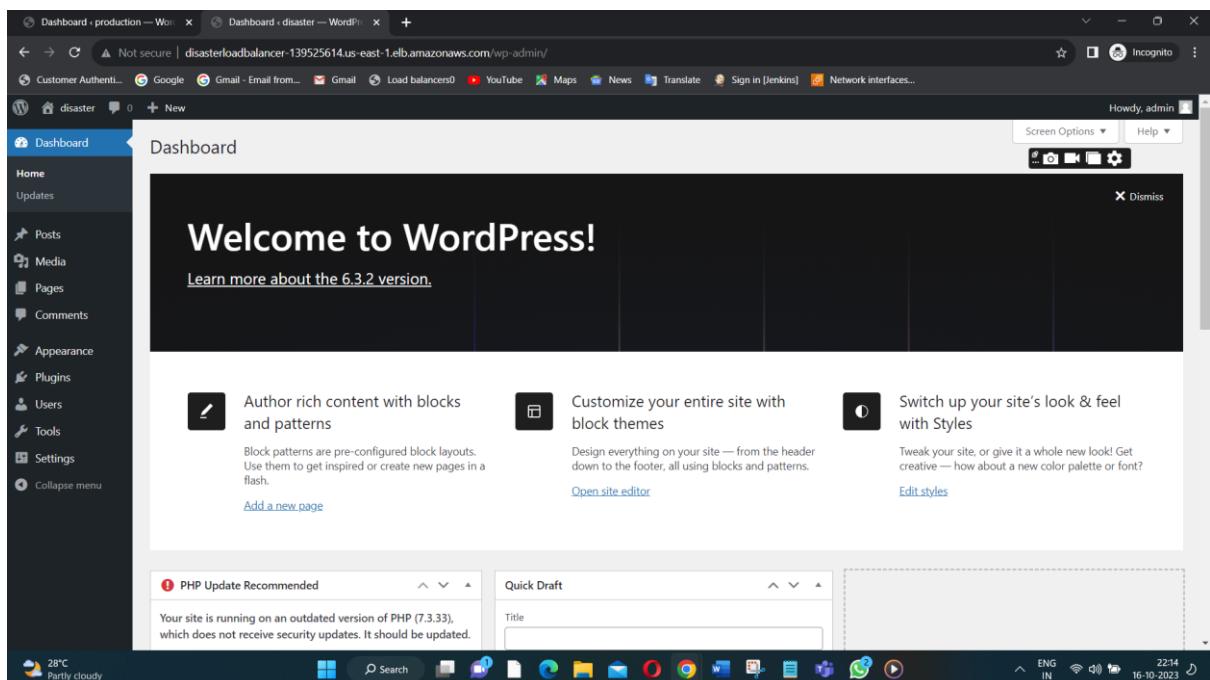
After providing our own credentials we should install wordpress.





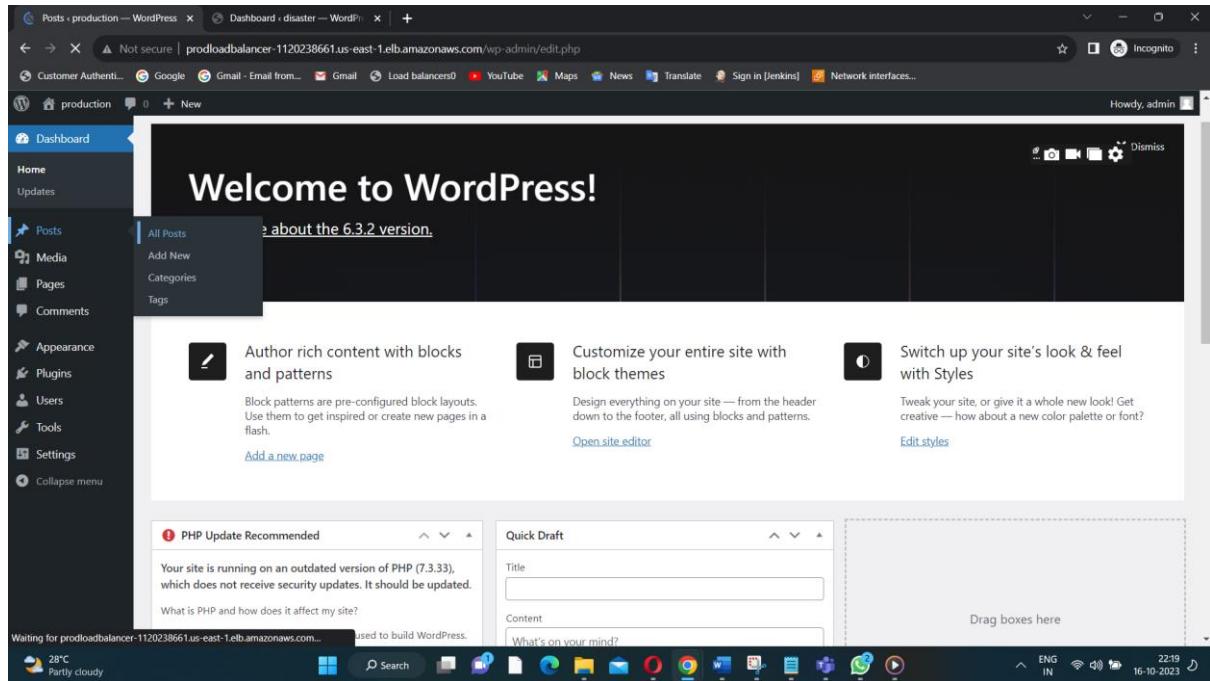
Wordpress page will get displayed.

Same process we need to done for another (disaster machine).

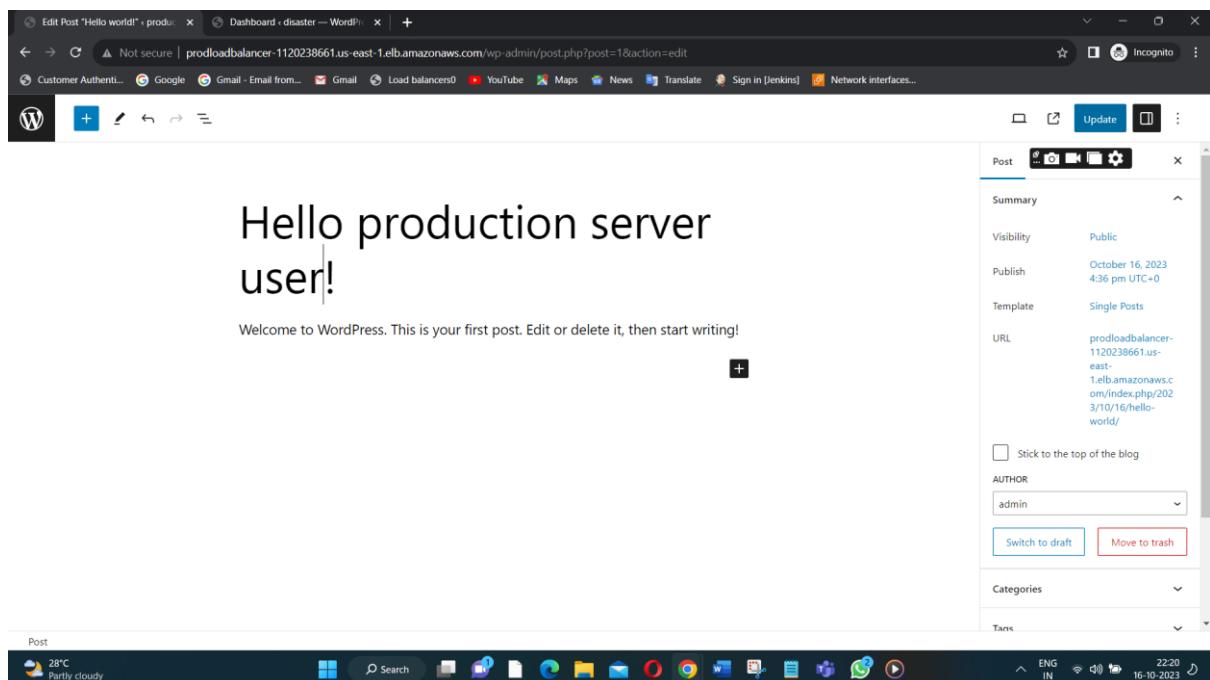


Wordpress page for disaster instance (need to wp-config.php in a cd /var/www/html file path.)

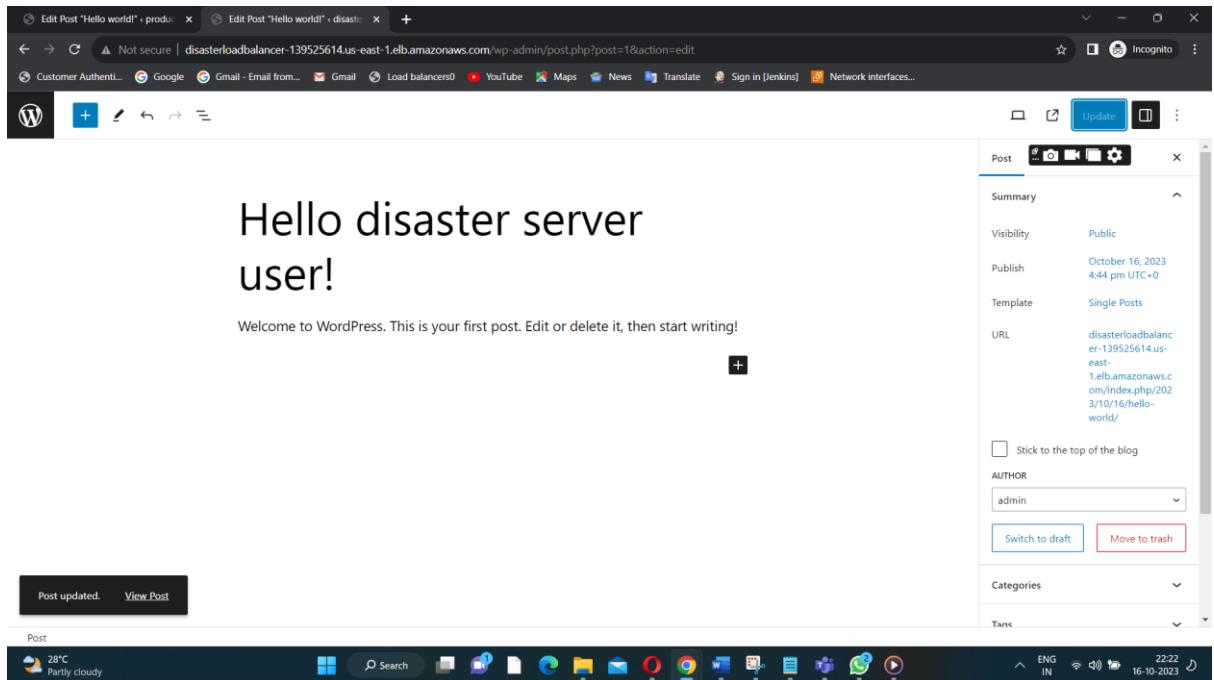
I want a different content in my production and disaster wordpress.



Click on post and all posts.

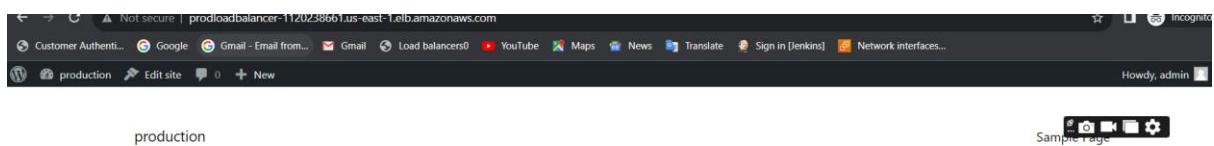


Give the message and click on updating.



Disaster server wordpress update(post updates).

Just editing the content on both the servers.



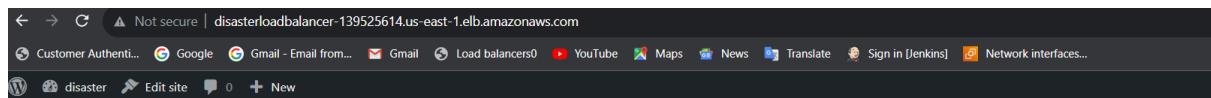
Mindblown: a blog about philosophy.

Hello production server user!

Welcome to WordPress. This is your first post.
Edit or delete it, then start writing!

October 16, 2023

Now I ping with productionloadbalancer(I got the output)like this.



disaster

Sar

Mindblown: a blog about philosophy.

Hello disaster server user!

Welcome to WordPress. This is your first post.
Edit or delete it, then start writing!

October 16, 2023

Now I pin with a disaster load balancer(I got the output of updated content).

STEP9:CONNECT A ROUTE53.

Create a record and route a traffic to a destinated load balancer.

Quick create record Switch to wizard

Record 1 Delete

Record name Info prod .jaleela.in Keep blank to create a record for the root domain. Record type Info A – Routes traffic to an IPv4 address and some AWS resources

Alias Route traffic to Info Alias to Application and Classic Load Balancer ▼

US East (N. Virginia) ▼

dualstack.prodloadbalancer-1120238661.us-east-1.elb.amazonaws.com X

Alias hosted zone ID: Z35SXDOTRQ7X7K

Evaluate target health Yes

Routing policy Info Simple routing ▼

Add another record

Cancel Create records

A yellow circle highlights the 'Alias' section, specifically the 'Route traffic to' dropdown and the 'Alias to Application and Classic Load Balancer' input field.

Alias to be chosen to route the traffic to a destinated load balancer.(for production server we choose production load balancer).

The screenshot shows the AWS Route 53 Record details page. A success message at the top states: "Record for jaleela.in was successfully created. Route 53 propagates your changes to all of the Route 53 authoritative DNS servers within 60 seconds. Use 'View status' button to check propagation status." Below this, a table lists various DNS records. One record, "prod.jaleela.in", is highlighted with a checked checkbox and has its details expanded on the right side. The expanded details show the record name as "prod.jaleela.in", record type as "A", value as "dualstack.prodloadbalancer-1120238661.us-east-1.elb.amazonaws.com", alias status as "Yes", and TTL as "-".

Record for production(prod.jaleela.in)-domain is created successfully.

The screenshot shows the "Create record" interface in the AWS Route 53 console. A "Quick create record" form is displayed. The "Record name" field contains "dr" and the "Record type" dropdown is set to "Info" (A). The "Alias" radio button is selected. In the "Route traffic to" section, "Alias to Application and Classic Load Balancer" is chosen, and the "US East (N. Virginia)" region is selected. The "Alias" dropdown shows "dualstack.disasterloadbalancer-139525614.us-east-1.elb.amazonaws.com". Under "Routing policy", "Simple routing" is selected, and "Evaluate target health" is set to "Yes". At the bottom right of the form is a "Switch to wizard" link.

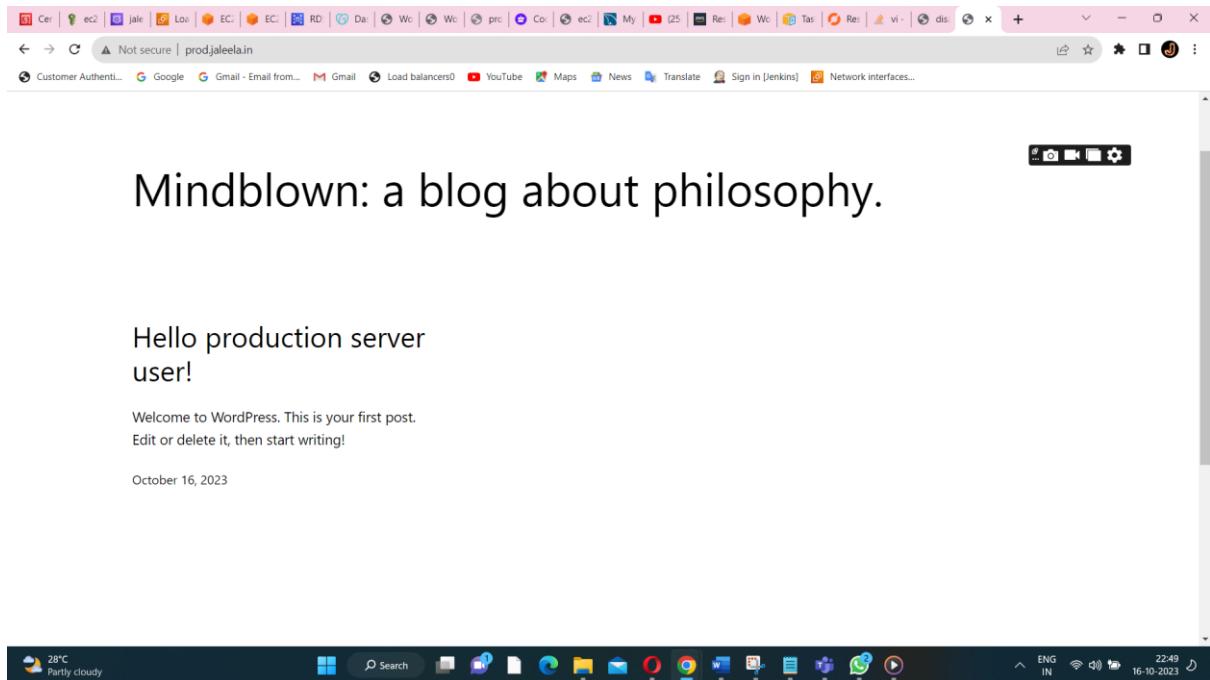
For the disaster too we can create a record.

The screenshot shows the AWS Route 53 service in the AWS Management Console. A modal window titled "Record for jaleela.in was successfully created." is displayed, stating "Route 53 propagates your changes to all of the Route 53 authoritative DNS servers within 60 seconds. Use 'View status' button to check propagation status." Below this, a table lists various DNS records. One record, "dr.jaleela.in" with type "A", has a checked checkbox and is highlighted. The "Record details" pane on the right shows the record name as "dr.jaleela.in", record type as "A", value as "dualstack.disasterloadbalancer-139525614.us-east-1.elb.amazonaws.com.", and routing policy as "Simple".

Disaster too record is created successfully.

The screenshot shows a WordPress blog post titled "Mindblown: a blog about philosophy." The post content includes the text "Hello disaster server user!" and "Welcome to WordPress. This is your first post. Edit or delete it, then start writing!". The date "October 16, 2023" is visible at the bottom of the post. The browser toolbar at the top shows the URL as "Not secure | dr.jaleela.in".

Now I ping the dr.jaleela.in,I got the output of a wordpress(for disaster server).

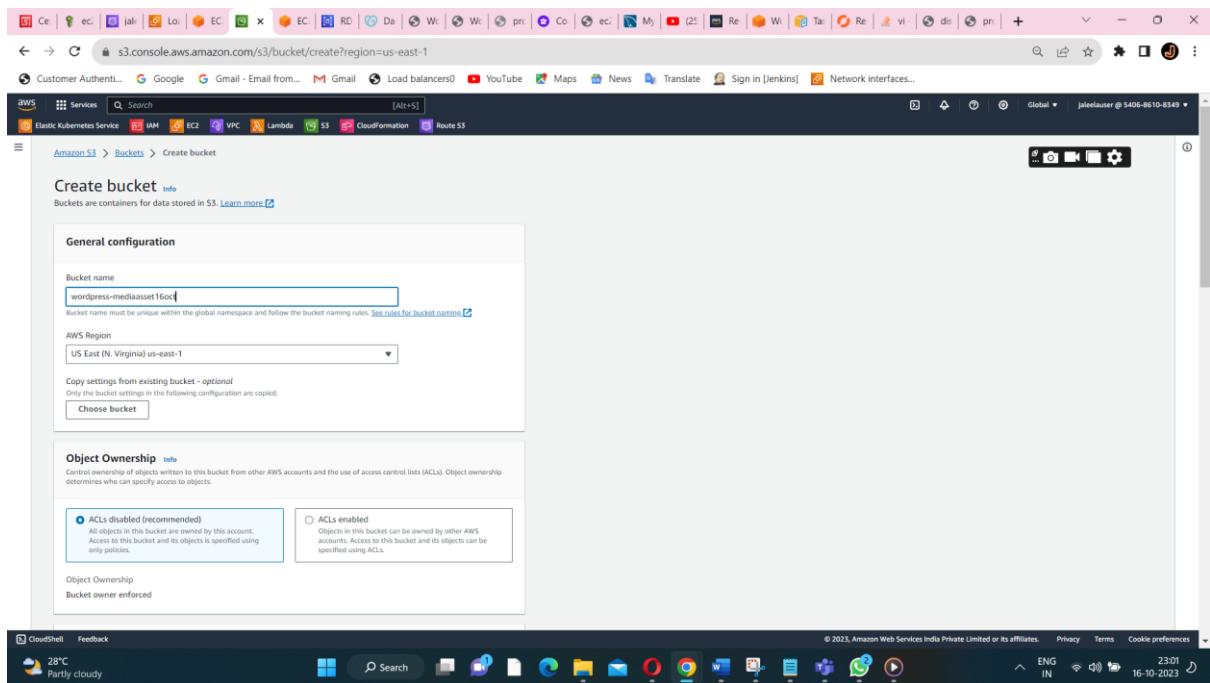


For prod.jaleela.in (we got the output of a wordpress of productions server).

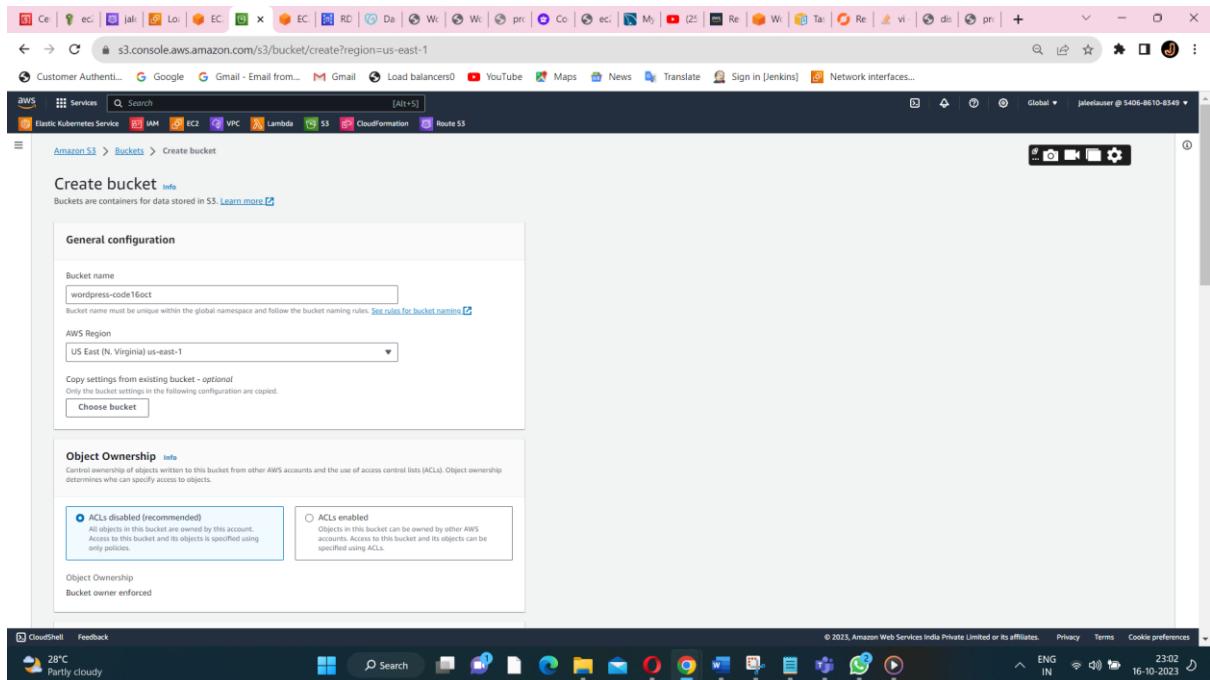
But both the domain name is not secured.

STEP10:DISASTER RECOVERY

Two create a two s3 bucket for storing code level content and a another bucket is for storing a media level content.



One of the bucket name is wordpress-mediaasset16oct(one bucket is used for storing media files).



Create a another bucket for storing a code.

<input type="radio"/>	wordpress-code16oct	US East (N. Virginia) us-east-1	Objects can be public	October 16, 2023, 23:02:16 (UTC+05:30)
<input type="radio"/>	wordpress-mediaasset16oct	US East (N. Virginia) us-east-1	Objects can be public	October 16, 2023, 23:01:33 (UTC+05:30)

Both the bucket is created successfully.

```

[ec2-user@ip-172-31-24-153 ~]$ sudo su
[root@ip-172-31-24-153 ec2-user]# crontab -e

```

Amazon Linux 2
AL2 End of Life is 2025-06-30.
A newer version of Amazon Linux is available!
Amazon Linux 2023, GA and supported until 2028-03-15.
<https://aws.amazon.com/linux/amazon-linux-2023/>

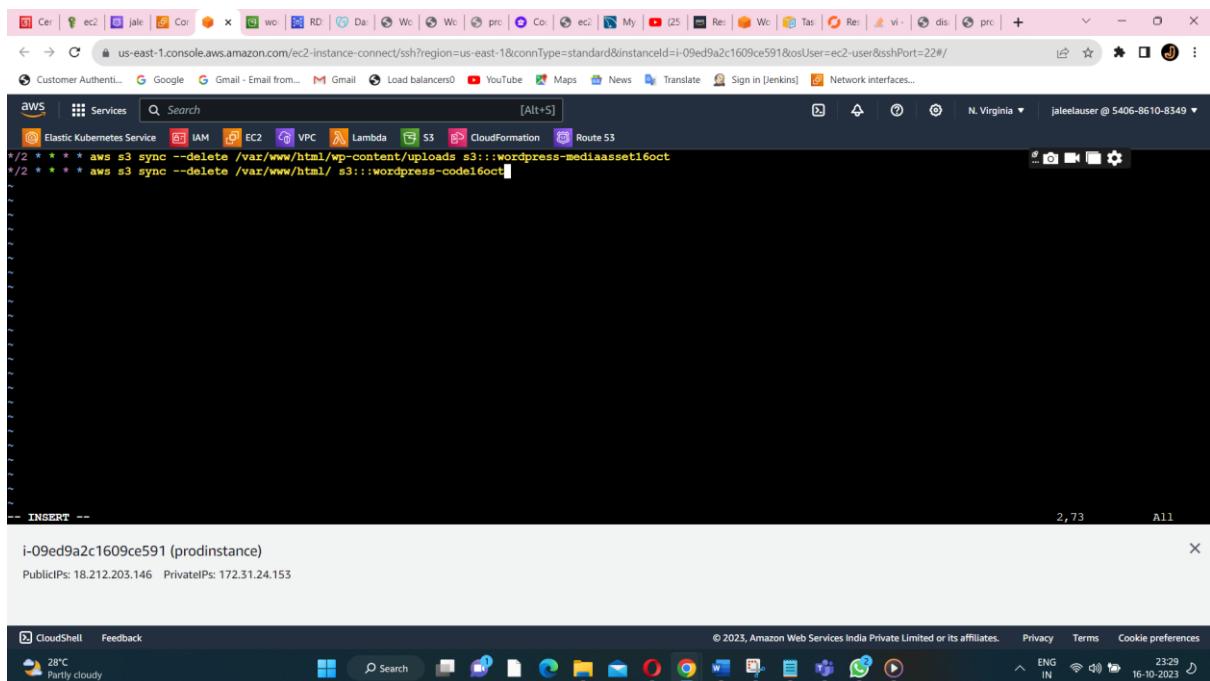
Crontab -e is used to write a scheduling job's occurred in a s3 bucket.

To create a crontab file for scheduling the jobs in a s3 bucket.

Cd /var/www/html

Cd wp-content/

Vi crontab -e(for writing the scheduling cron tab file).



Attach the cron-tab in a production server.(to get a wordpress upload content is stored in a s3 bucket).

```
*/2 * * * * aws s3 sync --delete /var/www/html/wp-content/uploads s3:::wordpress-mediaasset16oct
```

```
*/2 * * * * aws s3 sync --delete /var/www/html/ s3:::wordpress-code16oct
```

to sync with a s3 bucket.

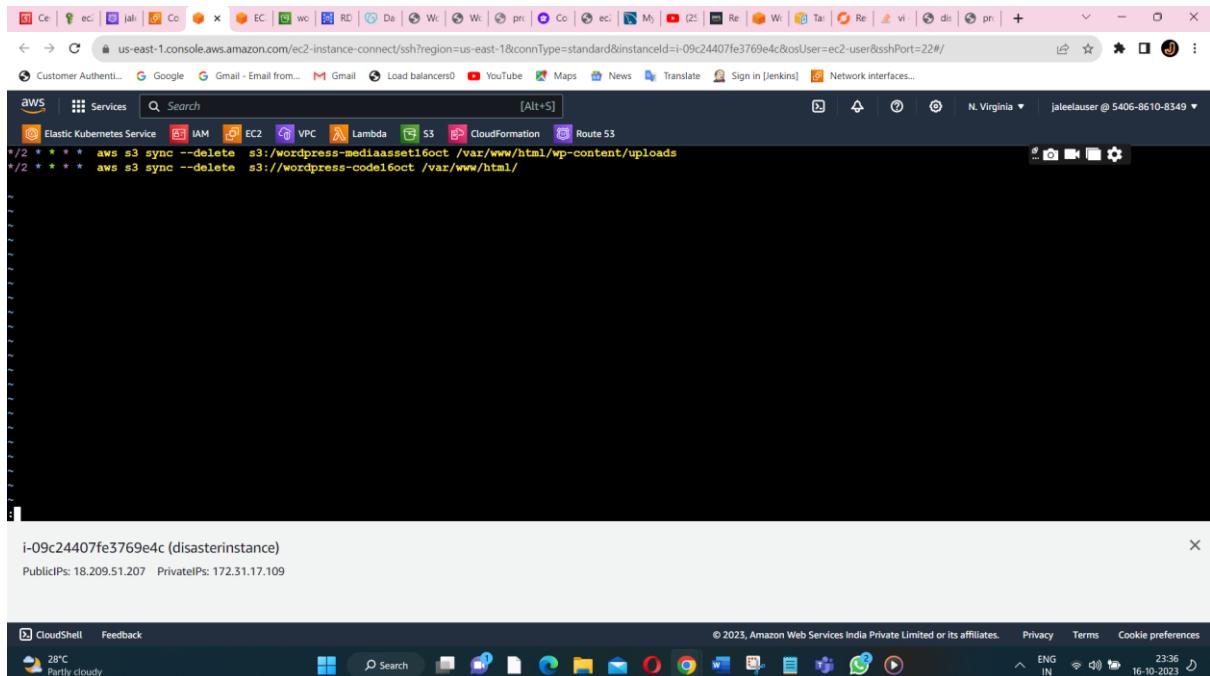
```

last login: Mon Oct 16 17:49:15 2023 from ec2-18-206-107-27.compute-1.amazonaws.com
'          #
~\_ #####      Amazon Linux 2
~~ \#####\ AL2 End of Life is 2025-06-30.
~~  \###| 
~~   \#/ __>
~~    V~' '--->
~~     / A newer version of Amazon Linux is available!
~~ .-' / Amazon Linux 2023, GA and supported until 2028-03-15.
~/m/ /' https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-24-153 ~]$ sudo su
[root@ip-172-31-24-153 ec2-user]# crontab -e
no crontab for root - using an empty one
crontab: installing new crontab
[root@ip-172-31-24-153 ec2-user]# crontab -l
*/2 * * * * aws s3 sync --delete /var/www/html/wp-content/uploads s3:::wordpress-mediaasset16oct
*/2 * * * * aws s3 sync --delete /var/www/html/ s3:::wordpress-code16oct
[root@ip-172-31-24-153 ec2-user]# []

```

Crontab -l is used to view the crontab jobs inside the crontab -e file.

Same crontab job is uploaded in a disaster server to get all the contents that is uploaded in a production server.



Add the crontab jobs.

```

[root@ip-172-31-17-109 ec2-user]# crontab -l
*/2 * * * * aws s3 sync --delete s3:::wordpress-mediaasset16oct /var/www/html/wp-content/uploads
*/2 * * * * aws s3 sync --delete s3:::wordpress-code16oct /var/www/html/

```

Is used to view the disaster path in a disaster server.

```
*/2 * * * * aws s3 sync --delete s3://wordpress-mediaasset16oct /var/www/html/wp-content/uploads
```

```
*/2 * * * * aws s3 sync --delete s3://wordpress-code16oct /var/www/html/
```

Command for disaster server(wordpress)

```
*/2 * * * * aws s3 sync --delete /var/www/html/wp-content/uploads s3://wordpress-mediaasset16oct
```

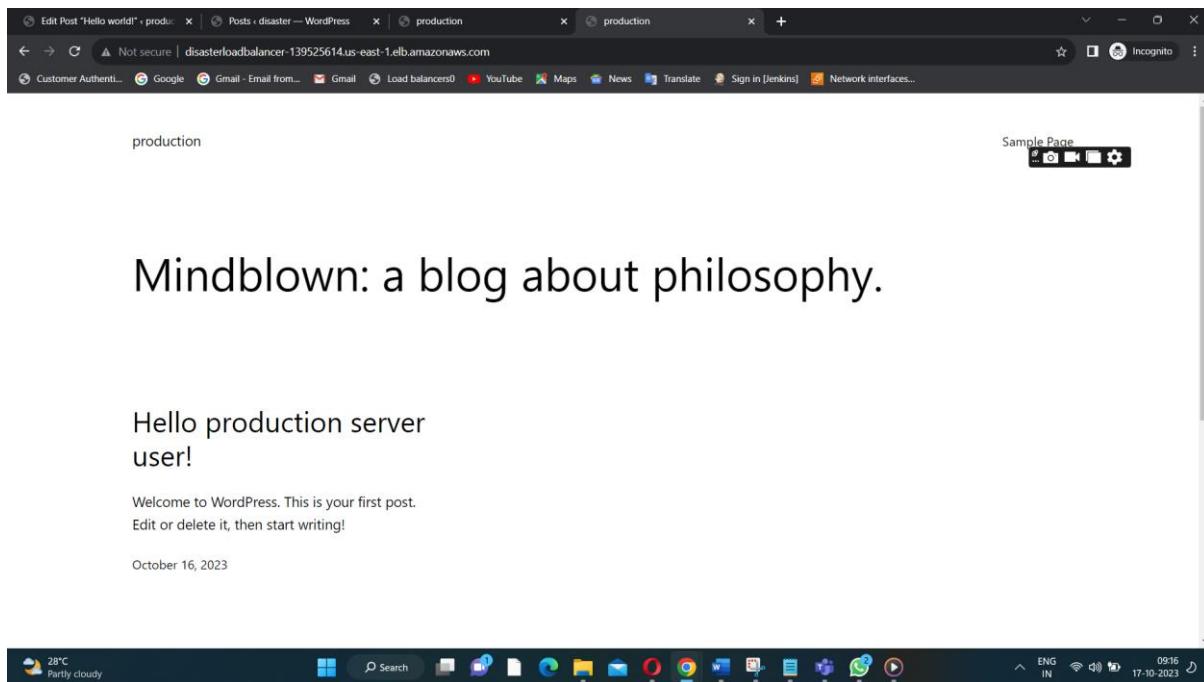
```
*/2 * * * * aws s3 sync --delete /var/www/html/ s3://wordpress-code16oct
```

Commands for production server(wordpress)

For to upload the content from a production server to a disaster server(to provide the replication).

Name	Type	Last modified	Size	Storage class
healthy.html	html	October 17, 2023, 09:06:03 (UTC-05:30)	8.0 B	Standard
index.php	php	October 17, 2023, 09:06:03 (UTC-05:30)	405.0 B	Standard
license.txt	txt	October 17, 2023, 09:06:03 (UTC-05:30)	19.4 KB	Standard
readme.html	html	October 17, 2023, 09:06:03 (UTC-05:30)	7.2 KB	Standard
wp-activate.php	php	October 17, 2023, 09:06:03 (UTC-05:30)	7.0 KB	Standard
wp-admin/	Folder	-	-	-
wp-blog-header.php	php	October 17, 2023, 09:06:07 (UTC-05:30)	551.0 B	Standard
wp-comments-post.php	php	October 17, 2023, 09:06:07 (UTC-05:30)	2.3 KB	Standard
wp-config-sample.php	php	October 17, 2023, 09:06:07 (UTC-05:30)	2.9 KB	Standard
wp-config.php	php	October 17, 2023, 09:06:07 (UTC-05:30)	3.2 KB	Standard
wp-content/	Folder	-	-	-
wp-cron.php	php	October 17, 2023, 09:06:10 (UTC-05:30)	5.5 KB	Standard
wp-includes/	Folder	-	-	-
wp-links-opml.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	2.4 KB	Standard
wp-load.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	3.8 KB	Standard
wp-login.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	48.5 KB	Standard
wp-mail.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	8.3 KB	Standard
wp-settings.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	25.0 KB	Standard
wp-signup.php	php	October 17, 2023, 09:06:24 (UTC-05:30)	33.6 KB	Standard

Here after in the s3 bucket all the content get added.



Mindblown: a blog about philosophy.

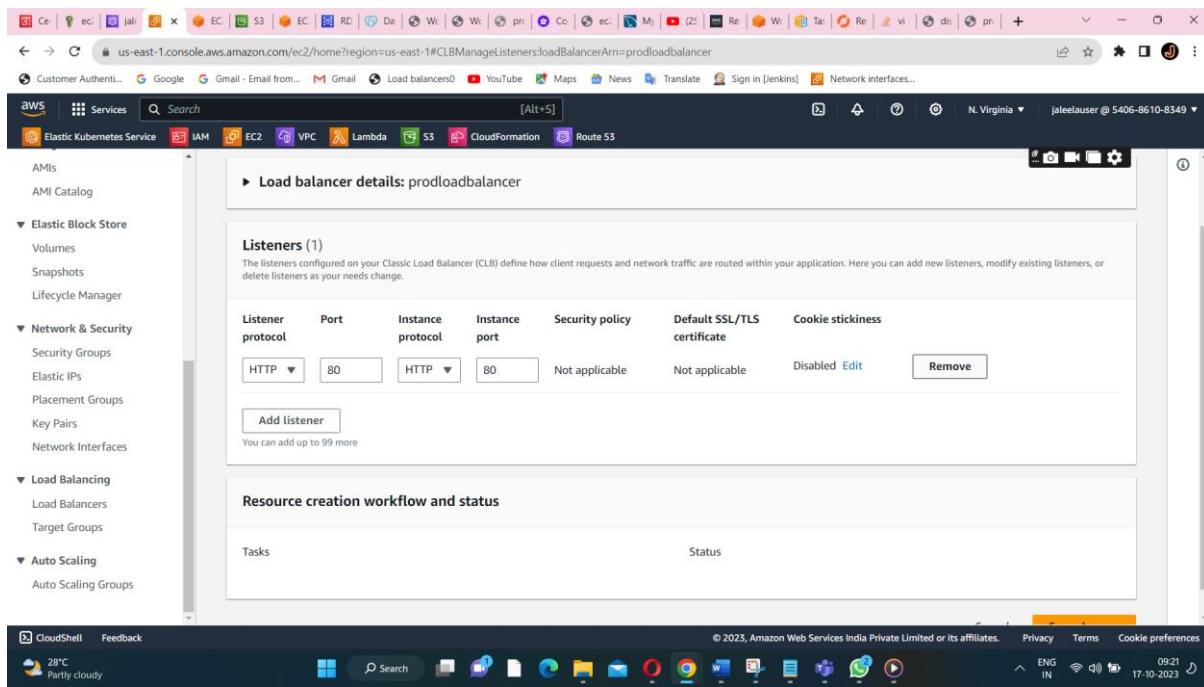
Hello production server user!

Welcome to WordPress. This is your first post.
Edit or delete it, then start writing!

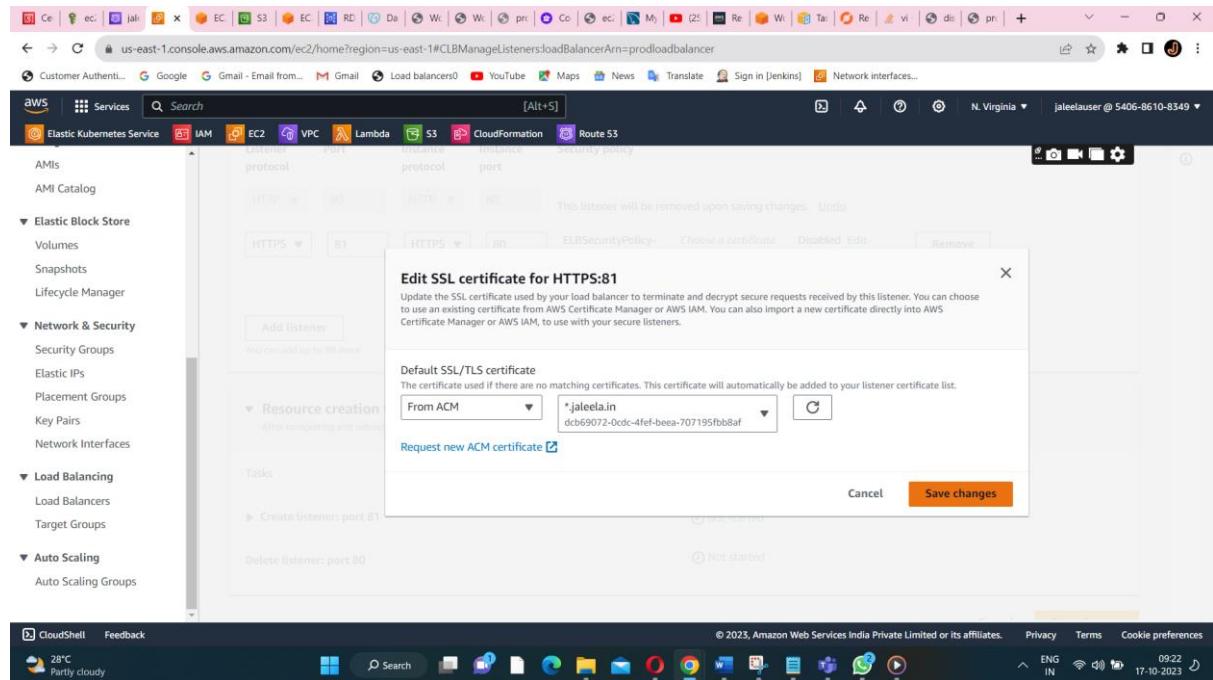
October 16, 2023

After I load the disaster server page(I got the production server page) output.

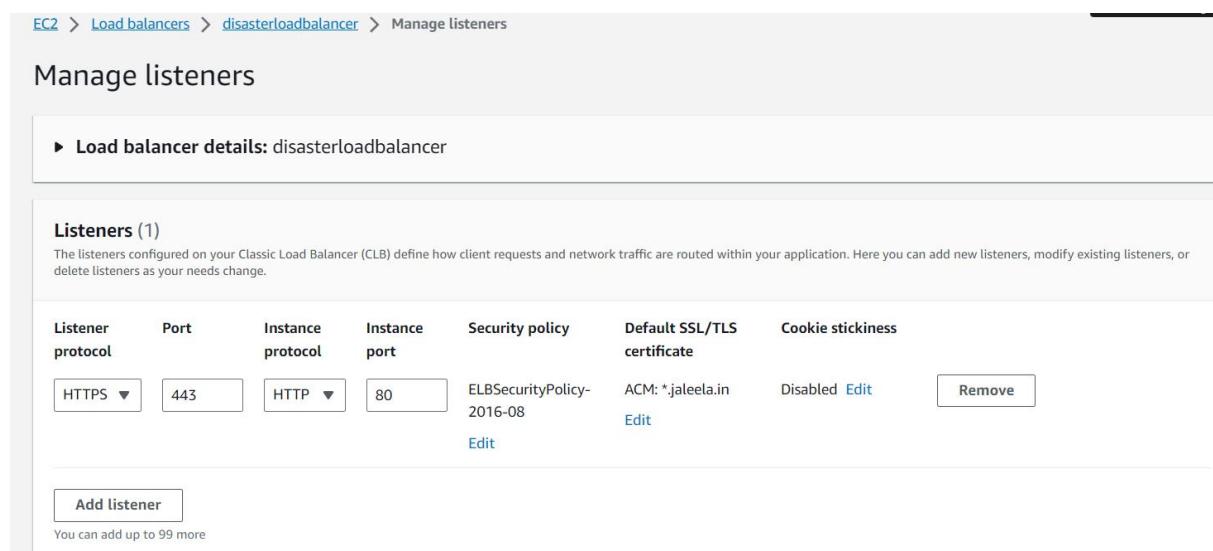
STEP11:To make the domain name secure to add the certificate to a loadbalancer domain name.



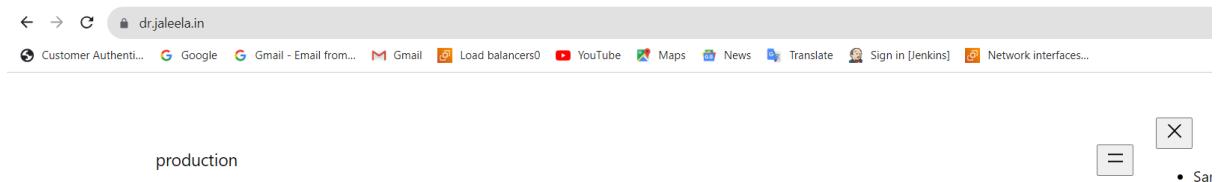
Inside the loadbalancer listener is there(to remove the http and add https along with a certificate).



To add the certificate from a ACM(aws certificate manager).



Listener protocol should be:443 and instance protocol should be:80.



Mindblown: a blog about philosophy.

Hello production server user!

Welcome to WordPress. This is your first post.
Edit or delete it, then start writing!

October 16, 2023

While ping <https://dr.jaleela.in> I got the output.

The screenshot shows the 'Manage listeners' section of the AWS CloudWatch Metrics interface. It displays a single listener configuration for a load balancer named 'prodloadbalancer'. The listener details are as follows:

Listener protocol	Port	Instance protocol	Instance port	Security policy	Default SSL/TLS certificate	Cookie stickiness
HTTPS ▾	443	HTTP ▾	80	ELBSecurityPolicy-2016-08	ACM: *.jaleela.in Edit	Disabled Edit Remove

Below the table, there is a button labeled 'Add listener' and a note stating 'You can add up to 99 more'.

At the bottom, there is a section titled 'Resource creation workflow and status' with a note: 'After completion and submitting the above steps, all server-side tasks and their statuses become available for monitoring.'

Listener for prodloadbalancer.



Mindblown: a blog about philosophy.

Hello production server user!

Welcome to WordPress. This is your first post.

Edit or delete it, then start writing!

October 16, 2023

Here,ping <https://prod.jaleela.in> got the output.

Here,we successfully executed the secured website launching.

