

Next Attempt

## Step 1 — Launch WordPress Instances via CloudFormation

- CloudFormation stack created with parameters:
  - VPC and Public Subnet selection
  - Key Pair for SSH access
  - Latest Amazon Linux 2023 AMI via SSM Parameter
- Security Group configured for HTTP (80) and SSH (22)
- UserData script installs Apache, PHP, MariaDB, and WordPress
- IAM role attached for CloudWatch monitoring

## Step 2 - Verify WordPress Deployment

- Prod and Dev instances reachable via:

Prod Instance:

<http://34.204.50.59/wordpress/> and <http://3.210.198.207/wordpress/>

Dev-Instance:

<http://54.84.44.103/wordpress>

CloudFormation > Stacks > Create stack

Step 3: Configure stack options

**Stack name:** wordpress-devprod

Stack name must contain only letters (a-z, A-Z), numbers (0-9), and hyphens (-) and start with a letter. Max 128 characters. Character count: 17/128.

**Parameters:**

- KeyPairName:** Select an existing KeyPair  
wpdp-key
- LatestAmiId:** Latest Amazon Linux 2023 AMI  
/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86\_64
- SubnetId:** Select a PUBLIC subnet (Auto-assign Public IP enabled)  
subnet-0dbd8e74010a2b38f
- VpcId:** Select your VPC (Default VPC recommended)  
vpc-0a73247a02e22776b

Cancel Previous Next

CloudFormation > Stacks > wordpress-devprod

Events (21)					
Operation ID	Timestamp	Logical ID	Status	Detailed status	
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:33:58 UTC-0800	wordpress-devprod	CREATE_COMPLETE		
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:33:55 UTC-0800	CloudWatchInstanceProfile	CREATE_COMPLETE		
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:33:00 UTC-0800	DevWordPressInstance	CREATE_COMPLETE		
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:33:00 UTC-0800	ProdWordPressInstance	CREATE_COMPLETE		
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:32:50 UTC-0800	wordpress-devprod	CREATE_IN_PROGRESS	CONFIGURATION_IN_PROGRESS	
196f765a-f44a-4e30-a75c-ff67981851a7	2026-02-18 15:32:50 UTC-0800	DevWordPressInstance	CREATE_IN_PROGRESS	CONFIGURATION_IN_PROGRESS	

Screenshot of the AWS CloudFormation console showing the Stacks page.

The page displays two stacks:

Stack name	Status	Created time	Description
wordpress-devprod	CREATE_COMPLETE	2026-02-18 15:31:23 UTC-0800	WordPress Prod and Dev/Test Stack on Amazon Linux 2023 with Monitoring
stack-2079977	CREATE_COMPLETE	2026-02-18 12:18:35 UTC-0800	-

Filter status: Active. View nested checkbox is checked.

Screenshot of the AWS EC2 console showing the Instances page.

The page displays two running instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
wp-prod	i-0a13d0db3deb0d02a	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c
wp-dev	i-0cb1058ef81dfc6f2	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

Filter: Instance state = running. Clear filters button is present.

Actions dropdown menu includes: Connect, Instance state, Actions, Launch instances.

Left sidebar navigation:

- EC2
- Dashboard
- AWS Global View
- Events
- Instances
  - Instances
  - Instance Types
  - Launch Templates
  - Spot Requests
  - Savings Plans
  - Reserved Instances
  - Dedicated Hosts
  - Capacity Reservations
  - Capacity Manager [New](#)
- Images
  - AMIs
  - AMI Catalog
- Elastic Block Store
  - Volumes
  - Snapshots
  - Lifecycle Manager

```
MariaDB [(none)]>
MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| wordpress_dev   |
+-----+
4 rows in set (0.000 sec)

MariaDB [(none)]> SELECT user, host FROM mysql.user;
+-----+-----+
| User      | Host     |
+-----+-----+
| mariadb.sys | localhost |
| mysql       | localhost |
| root        | localhost |
| wpuser      | localhost |
+-----+-----+
4 rows in set (0.001 sec)
```

Second Time Link changed.  
Link: <http://3.210.198.207/wordpress/>

The screenshot shows the WordPress dashboard at [3.219.29.207/wordpress/wp-admin/](http://3.219.29.207/wordpress/wp-admin/). The top navigation bar includes links for AOL Journey Portal, Harmony AOL, Voice to Text, Shakti Awakening..., Google Calendar, ParentVUE, Parent Portal Q, Dashboard | Learn..., Soundarya Lahari..., and Gemini. A "Finish update" button is also present.

The dashboard sidebar on the left lists: Home, Updates, Posts, Media, Pages, Comments, Appearance, Plugins, Users, Tools, Settings, and Collapse Menu. The "Dashboard" tab is selected.

The main content area features a large "Welcome to WordPress!" banner with a "Learn more about the 6.9.1 version." link. Below the banner are three cards:

- Author rich content with blocks and patterns**: Includes a "Add a new page" button.
- Customize your entire site with block themes**: Includes a "Open site editor" button.
- Switch up your site's look & feel with Styles**: Includes a "Edit styles" button.

Two side panels are visible:

- Site Health Status**: Shows "No information yet..." and a note about site health checks.
- Quick Draft**: Fields for "Title" and "Content" with placeholder text "What's on your mind?"

```
[ec2-user@ip-172-31-10-184 ~]$ sudo mysql -u root -p
|Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 24
Server version: 10.5.29-MariaDB MariaDB Server

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

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

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| wordpress       |
+-----+
4 rows in set (0.002 sec)

MariaDB [(none)]> SELECT user, host FROM mysql.user;
+-----+-----+
| User      | Host     |
+-----+-----+
| mariadb.sys | localhost |
| mysql       | localhost |
| root        | localhost |
| wpuser      | localhost |
+-----+-----+
4 rows in set (0.002 sec)
```

Link: <http://3.210.198.207/wordpress>

Not Secure 3.210.198.207/wordpress/wp-admin/setup-config.php?step=1 Gemini

AOLF Journey Portal Harmony AOL Voice to Text Shakti Awakening... Google Calendar ... ParentVUE Parent Portal Q Dashboard | Learn... Soundarya Lahari... Finish update

Below you should enter your database connection details. If you are not sure about these, contact your host.

**Database Name**  The name of the database you want to use with WordPress.

**Username**  Your database username.

**Password**   Your database password.

**Database Host**  You should be able to get this info from your web host, if localhost does not work.

**Table Prefix**  If you want to run multiple WordPress installations in a single database, change this.

Not Secure 3.210.198.207/wordpress/wp-admin/install.php?language=en\_US Gemini

AOLF Journey Portal Harmony AOL Voice to Text Shakti Awakening... Google Calendar ... ParentVUE Parent Portal Q Dashboard | Learn... Soundarya Lahari... Finish update

## Welcome

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**

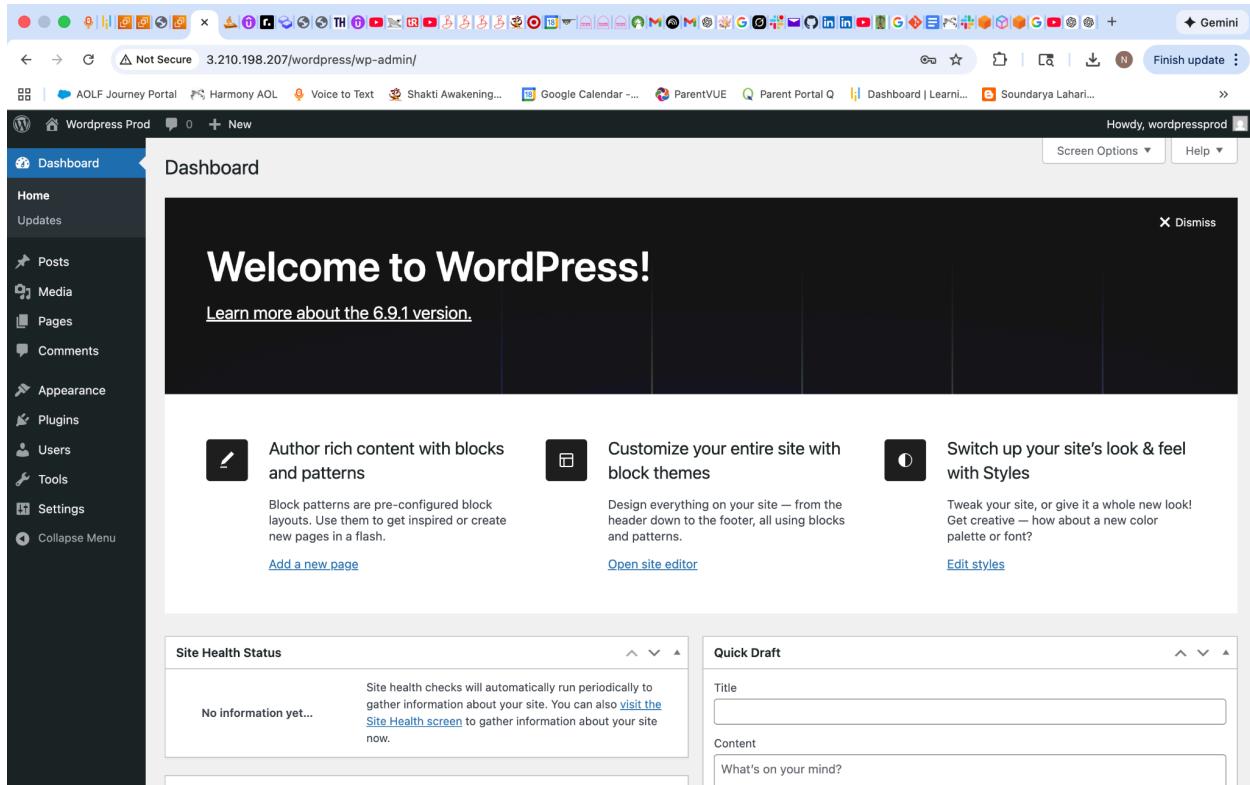
**Username**  Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

**Password**   Very weak

**Confirm Password**  Confirm use of weak password

**Your Email**  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.



## Step 3 - Create IAM Role EC2-CloudWatch-Role attached to all instances

IAM->Roles->Create Role (name:EC2-CloudWatch-Role)

Search and attach:

CloudWatchAgentServerPolicy

This policy allows:

- Sending logs
- Sending custom metrics (memory, disk, etc.)

**Identity and Access Management (IAM)**

**Roles (1/11) Info**

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
<a href="#">EC2-CloudWatch-Role</a>	AWS Service: ec2	-

**Roles Anywhere Info**

Authenticate your non AWS workloads and securely provide access to AWS services.

- Access AWS from your non AWS workloads**
- X.509 Standard**
- Temporary credentials**

IAM Roles attached to Dev and Prod instances

**EC2-CloudWatch-Role:**

Role is now attached to dev & prod EC2 instances.

## Step 4 - Create Launch Template, set Prod AMI and dev AMI respectively.

Purpose: Capture a ready-to-use image of the instance (with WordPress, PHP, Apache, MariaDB configured) for auto-scaling or backups.

Steps:

1. Go to EC2 → Instances → Select your WordPress instance (Dev or Prod)
  2. Click Actions → Image → Create Image
- AMIs created for both Dev and Prod instances:
    - Prod AMI: wordpress-prod-ami
    - Dev AMI: wordpress-dev-ami

Screenshot of the AWS EC2 'Create launch template' wizard.

**Software Image (AMI):** wordpress-prod-ami  
ami-080c36e3747b40dfe

**Virtual server type (instance type):** -

**Firewall (security group):** -

**Storage (volumes):** 1 volume(s) - 8 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 the internet. Data transfer charges are not included as part of the free tier allowance. Charges may apply depending

**Launch template name and description:**

- Launch template name - required: wordpress-prod-template
- Template version description: Prod wordpress
- Auto Scaling guidance: Select this if you intend to use this template with EC2 Auto Scaling
- Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

**Template tags:** No template tags are currently applied to this template. Add a template tag to apply it to the launch template.

**Source template:**

**Create launch template**

Screenshot of the AWS EC2 'Launch templates' page showing the 'wordpress-prod-template' details.

**Launch template details:**

- Launch template ID: lt-027b5ef0d3dfd48e5
- Launch template name: wordpress-prod-template
- Default version: 1
- Owner: arnaws:iam::992382775032:user/odl\_user\_2080071

**Launch template version details:**

Version	Description	Date created	Created by
1 (Default)	Prod wordpress	2026-02-19T05:37:57.000Z	arnaws:iam::992382775032:user/odl_user_2080071

**Instance details:**

- AMI ID: ami-080c36e3747b40dfe
- Instance type: t2.micro
- Key pair name: wpdp-key

**Resource tags:** -

**Network interfaces:** -

**Advanced details:** -

**Availability Zone:** -

**Security group IDs:** sg-0144ba35663157ef9

**Actions:** Actions ▾, Delete template, Delete template version

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 3 - optional  
 Integrate with other services

Step 4 - optional  
 Configure group size and scaling

Step 5 - optional  
 Add notifications

Step 6 - optional  
 Add tags

Step 7  
 Review

**Name**

**Auto Scaling group name**  
Enter a name to identify the group.  
  
Must be unique to this account in the current Region and no more than 255 characters.

**Launch template** Info

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

**Launch template**  
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.  
 (C)

**Create a launch template** L

**Version**  
Default (1) (C)

**Description**  
Prod wordpress

**Launch template**  
[wordpress-prod-template L](#)  
lt-027b5ef0d3dfd48e5

**AMI ID**  
ami-080c36e3747b40dfe

**Instance type**  
t2.micro

**Security groups**  
-

**Request Spot Instances**  
No

## Notifications

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

AOLF Journey Portal Harmony AOL Voice to Text Shakti Awakening... Google Calendar ... ParentVUE Parent Portal Q Dashboard | Learn... Soundarya Lahari...

aws Launch Wizard Ask Amazon Q

United States (N. Virginia) Simplilearn-Account-314 (9923-8277-5032) od\_user\_2080071

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1 Choose launch template

Step 2 Choose instance launch options

Step 3 - optional  
 Integrate with other services

Step 4 - optional  
 Configure group size and scaling

Add notifications

Step 6 - optional  
 Add tags

Step 7  
 Review

**Add notifications - optional** Info

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

**Notification 1** Remove

**Send a notification to**

**With these recipients**

**Use existing topic**

**Event types**  
Notify subscribers whenever instances

Launch  
 Terminate  
 Replace root volume  
 Fail to launch  
 Fail to terminate  
 Fail to replace root volume

**Add notification**

**Cancel** **Skip to review** **Previous** **Next**

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Step 5 - Configure Auto Scaling to Launch New WordPress Instances

Purpose: Ensure your environment can scale automatically (for high availability or testing scaling behavior).

Steps:

1. Create a Launch Template:

- AMI: Select the AMI you just created
- Instance Type: t2.micro (or your choice)
- Key Pair: Same as your original instance
- Security Group: Use the same WordPress SG
- IAM Role: CloudWatch monitoring role
- Optional: Add UserData script if you want extra configuration at boot

The screenshot shows the AWS CloudFormation Create Stack Wizard Step 7: Review screen. The page title is "Create stack". The main content area is titled "Review" and shows the following configuration:

- VPC:** Choose the VPC that defines the virtual network for your Auto Scaling group. Selected: `vpc-0459fa8f63e52077` (172.31.0.0/16, Default). Action: `Create a VPC`.
- Availability Zones and subnets:** Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC. Selected: `use1-az2 (us-east-1d) | subnet-062b8b4bdd5673c1e` (172.31.80.0/20, Default). Action: `Create a subnet`.
- Availability Zone distribution - new:** Auto Scaling automatically balances instances across Availability Zones. If launch failures occur in a zone, select a strategy.
  - Balanced best effort**: If launches fail in one Availability Zone, Auto Scaling will attempt to launch in another healthy Availability Zone.
  - Balanced only**: If launches fail in one Availability Zone, Auto Scaling will continue to attempt to launch in the unhealthy Availability Zone to preserve balanced distribution.

At the bottom, there are navigation buttons: `Cancel`, `Next` (highlighted in orange), `Previous`, and `Review`.

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

AOL Journey Portal Harmony AOL Voice to Text Shakti Awakening... Google Calendar ... ParentVUE Parent Portal Q Dashboard | Learn... Soundarya Lahari... Gemini

Finish update

Simplilearn-Account-314 (9923-8277-5032) odll\_user\_2080071

aws Search [Option+S] United States (N. Virginia) ▾

EC2 > Auto Scaling groups

wordpress-dev-asg created successfully

Last updated less than a minute ago

Auto Scaling groups (2) Info

Launch configurations Launch templates Actions Create Auto Scaling group

Search your Auto Scaling groups

Name	Launch template/configuration	Instances	Status	Desired capacity	Min	Max	Availability Zone
<a href="#">wordpress-dev-asg</a>	<a href="#">wordpress-dev-template</a>   Version Default	1	-	1	1	1	use1-az2 (us-east-1)
<a href="#">wordpress-prod-template</a>	<a href="#">wordpress-prod-template</a>   Version Default	1	-	1	1	2	use1-az1 (us-east-1)

0 Auto Scaling groups selected

Select an Auto Scaling group

CloudShell Feedback Console Mobile App

© 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Create Scheduled Actions for dev ASG

The screenshot shows the AWS CloudWatch Metrics console interface. At the top, there's a navigation bar with various links like AOL Journey Portal, Harmony AOL, Voice to Text, Shakti Awakening, Google Calendar, and a search bar. Below the navigation bar, the main content area has two sections:

- Predictive scaling policies (0) Info**: This section includes an "Evaluation period" dropdown set to "Evaluation based on 2 days". It features a table header with columns: Name, Metric pair, Forecast and scale, Recommendation, Chart, Availability impact, and Cost impact. A message below the table states, "No predictive scaling policies have been created. Predictive scaling policies use historical data to scale out your group ahead of forecasted hourly load." A blue "Create predictive scaling policy" button is at the bottom.
- Scheduled actions (0) Info**: This section includes a "Filter scheduled actions" input field. It features a table header with columns: Name, Start time, End time, Recurrence, Time zone, Desired capa..., Min, and Max. A message below the table states, "No scheduled actions are currently specified." A blue "Create scheduled action" button is at the bottom.

This screenshot is identical to the one above, but it shows the results of creating scheduled actions. In the "Scheduled actions" section, the count has increased to (2). The table now lists two entries:

Name	Start time	End time	Recurrence	Time zone	Desired capa...	Min	Max
Dev-Start-9AM	2026 February 1...		0 9 * * *	Etc/UTC	1		
Dev-Stop-6pm	2026 February 1...		0 18 * * *	Etc/UTC	0		

## Step 6 - Cloudwatch for Monitoring

### 1. Confirm IAM Role Attached to Prod

## Step 7 - Verify EC2 Instance Launched by ASG

- Terminate any running ASG-managed Prod instance (if needed)
- ASG will automatically launch a new instance using latest Launch Template version
- Confirm IAM Role column → should now show EC2-CloudWatch-Role

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with navigation links like Dashboard, Events, Instances (with sub-links for Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, and Capacity Manager), Images, and Elastic Block Store. The main content area has a header with 'Instances (1/3) Info' and filters for 'Instance state = running'. Below this is a table listing three instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
wp-prod	i-0ca32524019c4a277	Running	t2.micro	2/2 checks pass	View alarms	us-east-1c
	i-04186957f4a147f74	Running	t2.micro	2/2 checks pass	View alarms	us-east-1c
<input checked="" type="checkbox"/>	i-039df72d0f2619d04	Running	t2.micro	2/2 checks pass	View alarms	us-east-1c

Below the table, a specific instance is selected: **i-039df72d0f2619d04**. The details for this instance are shown in a expanded view:

Details		
Instance ID	Public IPv4 address	Private IPv4 addresses
i-039df72d0f2619d04	98.92.138.129   open address	172.31.4.92
IPv6 address	Instance state	Public DNS
-	Running	ec2-98-92-138-129.compute-1.amazonaws.com   open address
Hostname type	Private IP DNS name (IPv4 only)	
ip-172-31-4-92.ec2.internal	172.31.4.92	

At the bottom of the page, there are links for CloudShell, Feedback, and Console Mobile App, along with copyright information: © 2026, Amazon Web Services, Inc. or its affiliates.

Screenshot of the AWS EC2 Instances page showing the 'Stop instance' dialog for instance i-039df72d0f2619d04.

The dialog box contains the following information:

- Stop instance**: Stopping your instance allows you to reduce costs, modify settings, and troubleshoot problems.
- Instance ID**: i-039df72d0f2619d04
- Stop protection**: Disabled
- Result**: Can stop

Alerts and notes:

- Associated resources**: You will continue to incur charges for these resources while the instance is stopped.
- You will be billed for associated resources**: After you stop the instance, you are no longer charged usage or data transfer fees for it. However, you will still be billed for associated Elastic IP addresses and EBS volumes.
- Skip OS shutdown**: This option skips the graceful OS shutdown process. Use only when your instance must be stopped immediately, such as during an emergency or failover.

Buttons: Cancel, Stop

Screenshot of the AWS EC2 Instances page showing the successful initiation of stopping instance i-039df72d0f2619d04.

The message at the top of the page reads: "Successfully initiated stopping of i-039df72d0f2619d04".

The main table shows two instances:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
wp-prod	i-0ca32524019c4a277	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c
	i-04186957f4a147f74	Running	t2.micro	2/2 checks passed	View alarms +	us-east-1c

Details for instance i-039df72d0f2619d04:

- Public IPv4 address**: 98.92.138.129 | open address ↗
- Private IP DNS name (IPv4 only)**: 172.31.1.99 | open address ↗
- Instance state**: Stopping
- Private IPv4 addresses**: 172.31.4.92
- Public DNS**: ec2-98-92-138-129.compute-1.amazonaws.com | open address ↗

The ASG is automatically launching a new prod instance (named: ASG-prod2).

The screenshot shows the AWS EC2 Instances page. On the left, there's a sidebar with 'EC2' selected. The main area has a header 'Instances (3) Info' with a search bar 'Find Instance by attribute or tag (case-sensitive)' and a dropdown 'All states'. Below the header is a table with columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. The table contains three rows:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
wp-prod	i-0ca32524019c4a277	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>	us-east-1c
	i-04186957f4a147f74	Running	t2.micro	2/2 checks passed	<a href="#">View alarms</a>	us-east-1c
	i-08ccf22bf47ae4466	Running	t2.micro	Initializing	<a href="#">View alarms</a>	us-east-1c

## Step 8 — Install CloudWatch Agent

SSH into ASG-launched Prod instance:

```
sudo dnf install amazon-cloudwatch-agent -y
```

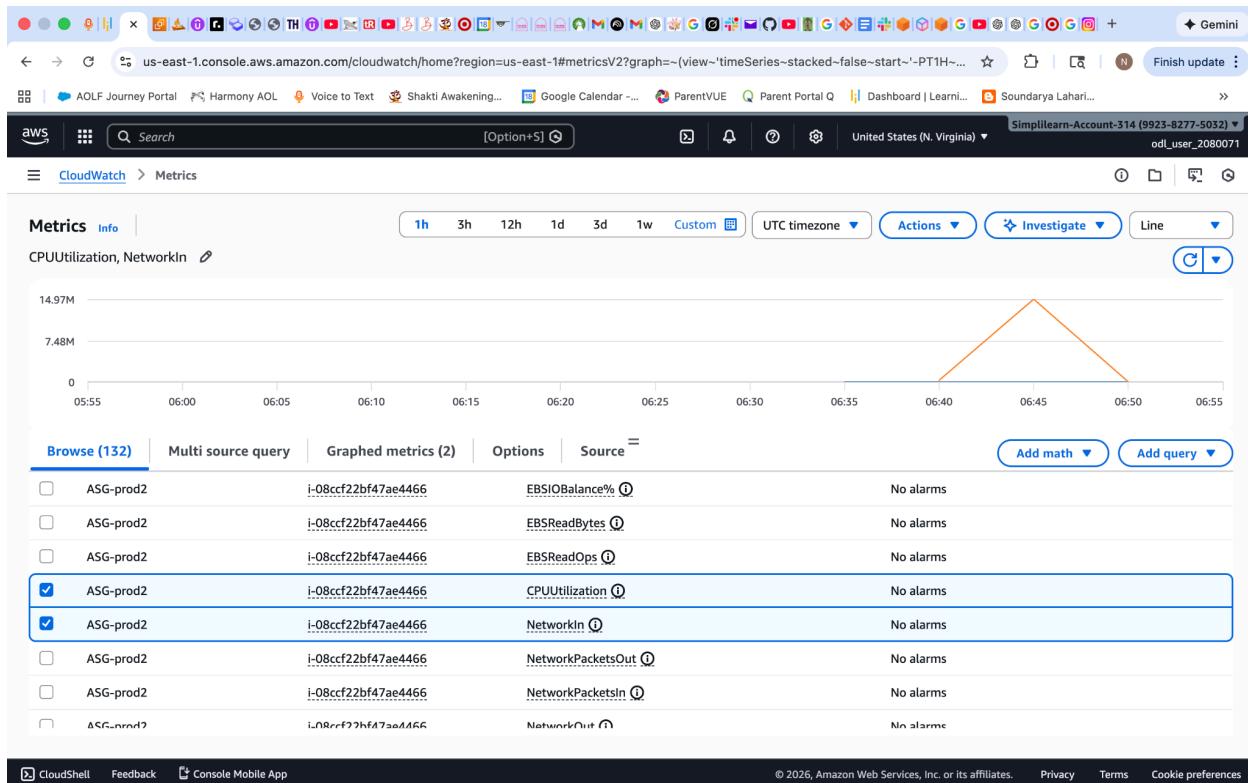
```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard
```

```
sudo systemctl start amazon-cloudwatch-agent
```

```
sudo systemctl enable amazon-cloudwatch-agent
```

Now metrics (memory, disk) will push automatically.

```
        "InstanceId"
    ],
    "append_dimensions": {
        "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
        "ImageId": "${aws:ImageId}",
        "InstanceId": "${aws:InstanceId}",
        "InstanceType": "${aws:InstanceType}"
    },
    "metrics_collected": {
        "collectd": {
            "metrics_aggregation_interval": 60
        },
        "disk": {
            "measurement": [
                "used_percent"
            ],
            "metrics_collection_interval": 1,
            "resources": [
                "*"
            ]
        },
        "mem": {
            "measurement": [
                "mem_used_percent"
            ],
            "metrics_collection_interval": 1
        },
        "statsd": {
            "metrics_aggregation_interval": 60,
            "metrics_collection_interval": 10,
            "service_address": ":8125"
        }
    }
}
Please check the above content of the config.
The config file is also located at /opt/aws/amazon-cloudwatch-agent/bin/config.json.
Edit it manually if needed.
Do you want to store the config in the SSM parameter store?
1: yes
2: no
default choice: [1]:
2
Program exits now.
[ec2-user@ip-172-31-5-19 ~]$ sudo systemctl start amazon-cloudwatch-agent
sudo systemctl enable amazon-cloudwatch-agent
Created symlink /etc/systemd/system/multi-user.target.wants/amazon-cloudwatch-agent.service → /etc/systemd/system/amazon-cloudwatch-agent.service.
[ec2-user@ip-172-31-5-19 ~]$
```



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

AOL Journey Portal Harmony AOL Voice to Text Shakti Awakening... Google Calendar ... ParentVUE Parent Portal Q Dashboard | Learn... Soundarya Lahari...

CloudWatch Metrics

EC2 Instances

Instances (1/6) Info Connect Instance state Actions Launch instances

Find Instance by attribute or tag (case-sensitive)

All states

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
wp-prod	i-0ca32524019c4a277	Running ⓘ ⓘ	t2.micro	2/2 checks passec	View alarms +	us-east-1c
	i-04186957f4a147f74	Running ⓘ ⓘ	t2.micro	2/2 checks passec	View alarms +	us-east-1c
wp-dev	i-0b58937a8302f7ba2	Stopped ⓘ ⓘ	t2.micro	-	View alarms +	us-east-1c
<b>ASG-prod2</b>	<b>i-08ccf22bf47ae4466</b>	<b>Running ⓘ ⓘ</b>	<b>t2.micro</b>	<b>2/2 checks passec</b>	<b>View alarms +</b>	<b>us-east-1c</b>
	i-039df72d0f2619d04	Terminated ⓘ ⓘ	t2.micro	-	View alarms +	us-east-1c
	i-084c81bed8343ee01	Terminated ⓘ ⓘ	t2.micro	-	View alarms +	us-east-1d

**i-08ccf22bf47ae4466 (ASG-prod2)**

Details Status and alarms Monitoring Security Networking Storage Tags

Instance summary

Instance ID: i-08ccf22bf47ae4466	Public IPv4 address: 3.239.18.128   open address ⓘ	Private IPv4 addresses: 172.31.5.19
IPv6 address: -	Instance state: Running	Public DNS: ec2-3-239-18-128.compute-1.amazonaws.com   open address ⓘ
Hostname type: -	Private IP DNS name (IPv4 only): -	

CloudShell Feedback Console Mobile App © 2026, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

**Course-End Project is fully functional:**

- Dev and Prod WordPress instances up and running - ✓
- Local MySql database configured and WordPress accessible - ✓
- IAM role attached for CloudWatch - ✓
- CloudWatch monitoring working (CPU, memory, disk) - ✓
- Dev instance scheduled (either via EventBridge or ASG) - ✓
- Optional Auto Scaling implemented for Prod - ✓
- Security groups - ✓