

FORWORD EC2-LOGS INTO CLOUDWATCH

Create ec2 with below shown configuration

Name and tags [Info](#)

Name
ec2-logs [Add additional tags](#)

Application and OS Images (Amazon Machine Image) [Info](#)

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

Recents **Quick Start**

Amazon Linux [aws](#) macOS [Mac](#) Ubuntu [ubuntu](#) Windows [Microsoft](#) Red Hat [Red Hat](#) SUSE Linux [SUSE](#) [Browse more AMIs](#) Including AMIs from AWS Marketplace and the Community

Amazon Linux 2023 AMI
ami-050cd642fd85388e4 (x86_64, uefi-preferred) / ami-0727d67ab27b16279 (x86_64, uefi)
Virtualization: hvm BIOS enabled: true Root device type: ebs **Free tier eligible**

Description
Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.6.20241010.0 x86_64 HVM kernel-6.1

Architecture: **G4-bit (x86)** Boot mode: **uefi-preferred** AMI ID: **ami-050cd642fd85388e4** Username: **ec2-user** **Verified provider**

Instance type [Info](#) [Get advice](#)

Instance type: **t2.micro** **Free tier eligible**
Family: t2 1 vCPU 1 GiB Memory Current generation: true On-Demand Linux base pricing: 0.0116 USD per hour

Network settings [Info](#)

VPC - required [Info](#)
vpc-0e5de924081b0145a (default) [Create new VPC](#)

Subnet [Info](#)
No preference [Create new subnet](#)

Auto-assign public IP [Info](#)
Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

☐ Create security group ☒ Select existing security group

Common security groups [Info](#)
Select security groups [Compare security](#)

Summary

Number of instances [Info](#)
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...[read more](#)
ami-050cd642fd85388e4

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
[Read the AWS Free tier limits and restrictions](#)

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

Click on Launch instance

Create IAM role for ec2 add administration access

- Open the instance
- Click on action

EC2 > Instances > i-0548292718f541006

Instance summary for i-0548292718f541006 (ec2-logs) [info](#)

Updated less than a minute ago

Instance ID
i-0548292718f541006

IPv6 address
-

Hostname type
IP name: ip-172-31-22-138.us-east-2.compute.internal

Answer private resource DNS name
IPv4 [A]

Auto-assigned IP address
3.140.187.136 [Public IP]

Public IPv4 address
3.140.187.136 | [open address](#)

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-22-138.us-east-2.compute.internal

Instance type
t2.micro

VPC ID
vpc-0e5de924081b0145a

Private IPv4 address
172.31.22.138

Public IPv4 DNS
ec2-3-140-187-136 | [open address](#)

Elastic IP addresses
-

AWS Compute Optimizer finding
[Opt-in to AWS Compute Optimizer for recommendation](#)

[Connect](#)

Instance state ▾

Actions ▲

- Connect
- Manage instance state
- Instance settings ▶
- Networking ▶
- Security ▶
- Image and templates ▶
- Monitor and troubleshoot ▶

- click on security
- select Modify IAM role

EC2 > Instances > i-0548292718f541006

Instance summary for i-0548292718f541006 (ec2-logs) [info](#)

Updated less than a minute ago

Instance ID
i-0548292718f541006

IPv6 address
-

Hostname type
IP name: ip-172-31-22-138.us-east-2.compute.internal

Answer private resource DNS name
IPv4 [A]

Public IPv4 address
3.140.187.136 | [open address](#)

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-22-138.us-east-2.compute.internal

Instance type
t2.micro

Private IPv4 address
172.31.22.138

Public IPv4 DNS
ec2-3-140-187-136 | [open address](#)

Elastic IP addresses
-

[Connect](#)

Instance state ▾

Actions ▲

- Connect
- Manage instance state
- Instance settings ▶
- Networking ▶
- Security ▶
- Image and templates ▶
- Monitor and troubleshoot ▶


- add your IAM ROLE HERE
- then update iam role

EC2 > Instances > i-0548292718f541006 > Modify IAM role

Modify IAM role [Info](#)


Attach an IAM role to your instance.



Instance ID

 i-0548292718f541006 (ec2-logs)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.



 [Create new IAM role](#) 

[Cancel](#) [Update IAM role](#)

- connect to instance

- switch to root user

sudo su -

```
[ec2-user@ip-172-31-22-138 ~]$ sudo su -  
[root@ip-172-31-22-138 ~]#
```

- install cloud watch agent

yum install amazon-cloudwatch-agent -y

```
[root@ip-172-31-22-138 ~]# yum install amazon-cloudwatch-agent -y
Last metadata expiration check: 0:04:42 ago on Thu Oct 24 07:42:17 2024.
Dependencies resolved.

Package                               Architecture Version                                R
Installing:
amazon-cloudwatch-agent              x86_64    1.300044.0-1.amzn2023                a

Transaction Summary
-----
Install 1 Package

Total download size: 135 M
Installed size: 445 M
Downloading Packages:
amazon-cloudwatch-agent-1.300044.0-1.amzn2023.x86_64.rpm

Total
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing      :
  Running scriptlet: amazon-cloudwatch-agent-1.300044.0-1.amzn2023.x86_64

i-0548292718f541006 (ec2-logs)
PublicIPs: 3.140.187.136 PrivateIPs: 172.31.22.138
```

- Edit this file or create file using below command

`vi /opt/aws/amazon-cloudwatch-agent/bin/config.json`

```
[root@ip-172-31-19-9 ~]# vi /opt/aws/amazon-cloudwatch-agent/bin/config.json
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
[root@ip-172-31-19-9 ~]#
```

- Add below json config details
- Change the highlighted details based on your requirement

```
{
  "agent": {
    "metrics_collection_interval": 60,
    "run_as_user": "root"
  },
  "logs": {
    "logs_collected": {
      "files": {
        "collect_list": [
```

```
        {
            "file_path": "/var/log/*",
            "log_group_class": "STANDARD",
            "log_group_name": "aws",
            "log_stream_name": "devops",
            "retention_in_days": 1
        }
    ]
}

},
"metrics": {
    "aggregation_dimensions": [
        [
            "InstanceId"
        ]
    ],
    "append_dimensions": {
        "AutoScalingGroupName": "${aws:AutoScalingGroupName}",
        "ImageId": "${aws:ImageId}",
        "InstanceId": "${aws:InstanceId}",
        "InstanceType": "${aws:InstanceType}"
    },
    "metrics_collected": {
        "cpu": {
            "measurement": [
                "cpu_usage_idle",
                "cpu_usage_iowait",
                "cpu_usage_user",
```

```
        "cpu_usage_system"
    ],
    "metrics_collection_interval": 60,
    "resources": [
        "*"
    ],
    "totalcpu": false
},
"disk": {
    "measurement": [
        "used_percent",
        "inodes_free"
    ],
    "metrics_collection_interval": 60,
    "resources": [
        "*"
    ]
},
"diskio": {
    "measurement": [
        "io_time"
    ],
    "metrics_collection_interval": 60,
    "resources": [
        "*"
    ]
},
"mem": {
    "measurement": [
```

```
        "mem_used_percent"
    ],
    "metrics_collection_interval": 60
},
"statsd": {
    "metrics_aggregation_interval": 60,
    "metrics_collection_interval": 10,
    "service_address": ":8125"
},
"swap": {
    "measurement": [
        "swap_used_percent"
    ],
    "metrics_collection_interval": 60
}
},
"traces": {
    "buffer_size_mb": 3,
    "concurrency": 8,
    "insecure": false,
    "region_override": "us-east-2",
    "traces_collected": {
        "xray": {
            "bind_address": "127.0.0.1:2000",
            "tcp_proxy": {
                "bind_address": "127.0.0.1:2000"
            }
        }
    }
}
```

```

    }
  }
}

```

- Enter below command for start the cloudwatch and creation of log group in cloud watch console based on config.json file

```

sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch-agent/bin/config.json -s

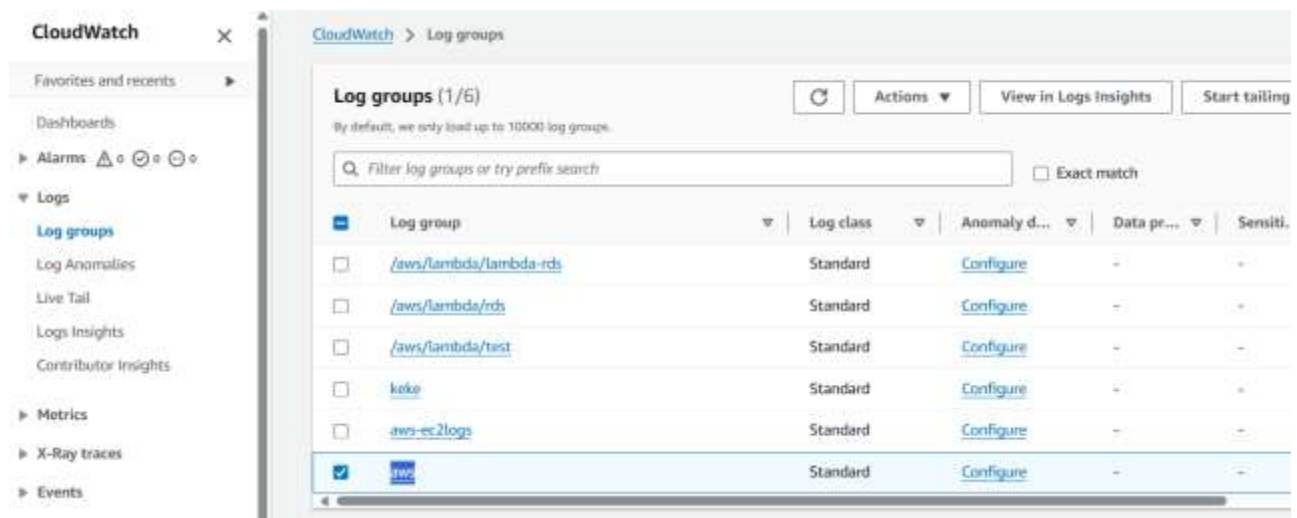
```

```

[root@ip-172-31-19-9 ~]# sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c file:/opt/aws/amazon-cloudwatch-agent
n/config.json -s
***** processing amazon-cloudwatch-agent *****
I! Trying to detect region from ec2 D! [EC2] Found active network interface I! imds retry client will retry 1 timesSuccessfully fetched the config and saved i
/opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp
Start configuration validation...
2024/10/24 08:50:51 Reading json config file path: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp ...
2024/10/24 08:50:51 I! Valid json input schema.
2024/10/24 08:50:51 D! ec2tagger processor required because append dimensions is set
2024/10/24 08:50:51 D! delta processor required because metrics with diskio or net are set
2024/10/24 08:50:51 D! ec2tagger processor required because append dimensions is set
2024/10/24 08:50:51 Configuration validation first phase succeeded
I! Detecting run as user...
I! Trying to detect region from ec2
D! [EC2] Found active network interface
I! imds retry client will retry 1 times
/opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent -schematest -config /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.toml
Configuration validation second phase succeeded
Configuration validation succeeded
amazon-cloudwatch-agent has already been stopped
Created symlink /etc/systemd/system/multi-user.target.wants/amazon-cloudwatch-agent.service -> /etc/systemd/system/amazon-cloudwatch-agent.service.

```

- Open your cloud watch and check



-----thank you-----

--thank you