

**Project in AWS
Practice Lab**

Implement Advanced CloudWatch Monitoring for a Web Server

Andra-Diana Popescu

2025

ABOUT THIS LAB

CloudWatch Logs centralizes the logs from all of your systems, applications, and AWS services that you use, in a single, highly scalable service. In this lab, you will configure an EC2 instance to stream its Apache web server error logs to CloudWatch Logs. You will configure the agent and then log in to the CloudWatch Logs console to make sure the logs are streamed correctly. By the end of this lab, you will understand how to install the CloudWatch Logs agent and configure it to stream a log to the service.

LEARNING OBJECTIVES

- Download and Run the CloudWatch Logs Installer
- Configure CloudWatch Logs
- Log In to the CloudWatch Logs Website

AWS Documentation about CloudWatch:

<https://aws.amazon.com/cloudwatch/faqs/#topic-0>

https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch_concepts.html

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/CloudWatch-Agent-Configuration-File-Details.html>

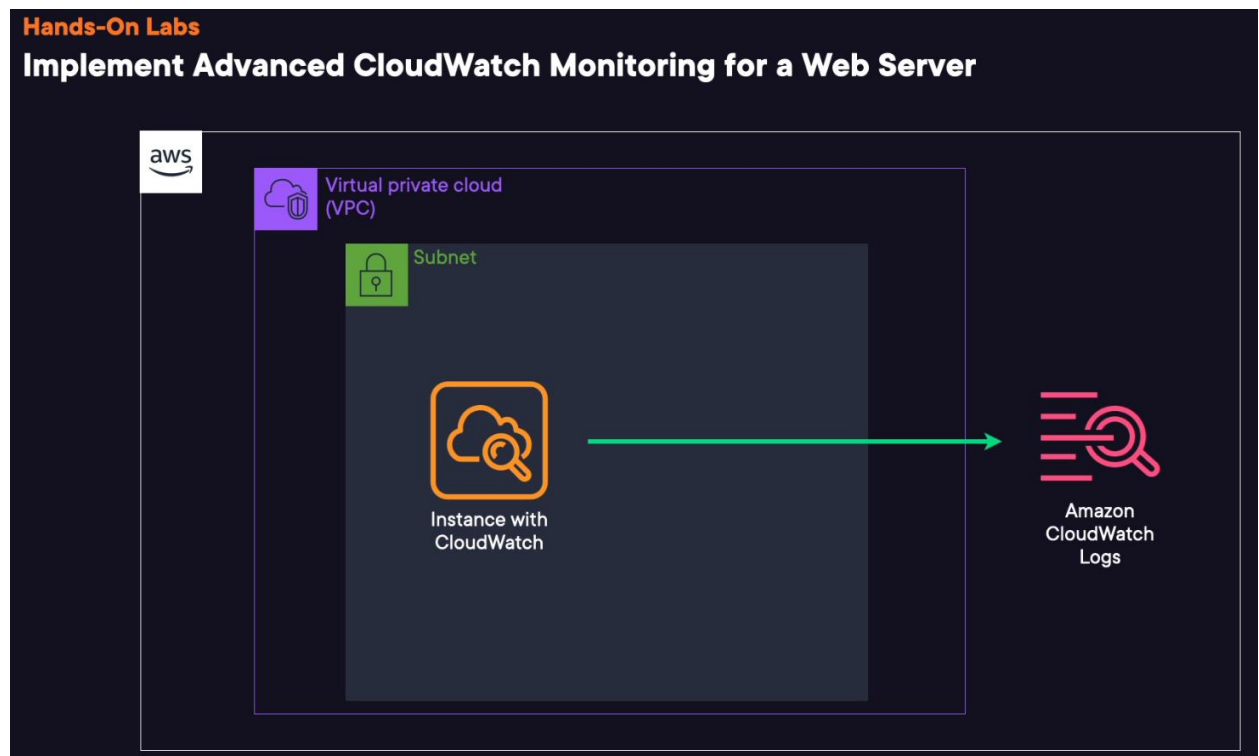
<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/create-cloudwatch-agent-configuration-file-wizard.html>

Source: <https://learn.acloud.guru/course/certified-solutions-architect-associate/>

Table of Contents

Lab Diagrams.....	4
Log in to your AWS account	5
1. Download and Run the CloudWatch Logs Installer	5
2. Configure CloudWatch Logs	7
3. Log In to the CloudWatch Logs Website.....	10

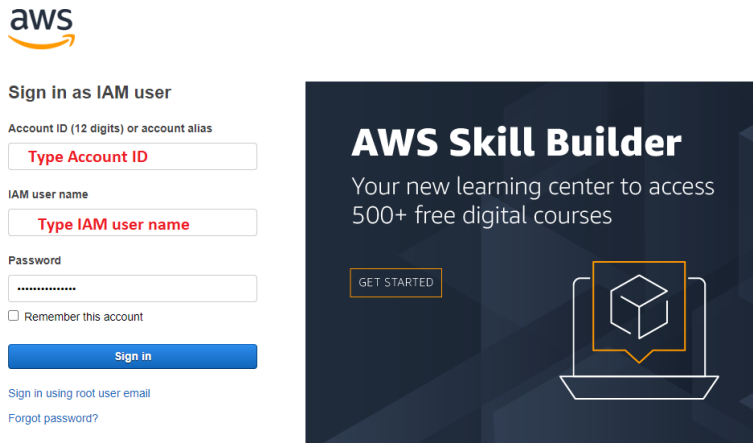
Lab Diagrams



We have the AWS account in **us-east-1** Region. In this lab, you're setting up a new web server for a company which will host a critical application. You've been asked to store the Apache Web logs in an Amazon CloudWatch log group in case any problems occur on the server.

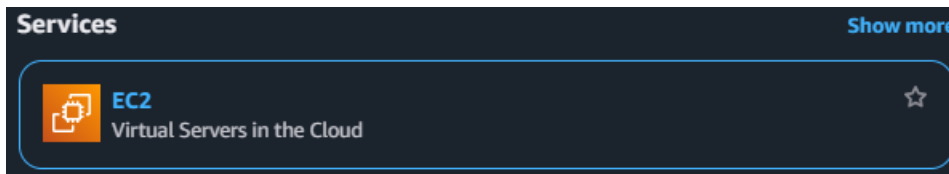
You'll be logging into the EC2 instance and installing the CloudWatch Logs agent. Once installed, you'll be configuring the CloudWatch logs agent to send copies of the Apache Web logs to the CloudWatch Logs group.

Log in to your AWS account

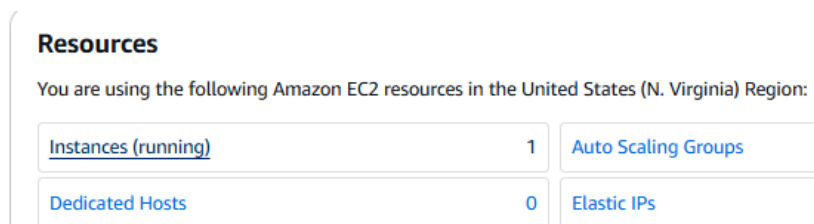


1. Download and Run the CloudWatch Logs Installer

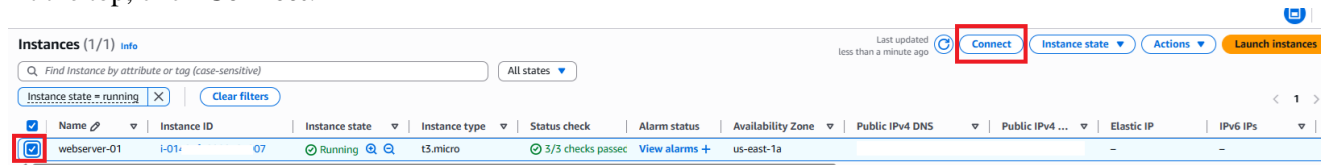
1. Once you are logged in to the AWS Management Console, navigate to **EC2**.



2. Click **Instances (running)**.



3. Click the checkbox next to **webserver-01**. Please give the lab an extra few minutes before connecting to **webserver-01**.
4. At the top, click **Connect**.



5. At the bottom of the page, click **Connect** to access the CLI.

Connect info

Connect to an instance using the browser-based client.

EC2 Instance Connect

Session Manager

SSH client

EC2 serial console

Instance ID

i-07 (webserver-01)

☒ Connect using a Public IP

☐ Connect using a Private IP

Connect using a public IPv4 or IPv6 address

Connect using a private IP address and a VPC endpoint

☒ Public IPv4 address

☐ IPv6 address

Public IPv4 address

4 6

Username

Enter the username defined in the AMI used to launch the instance. If you didn't define a custom username, use the default username, ubuntu.

Q ubuntu X

Note: In most cases, the default username, ubuntu, is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

Cancel

Connect

- Run the following commands to install the CloudWatch Logs agent:

```
wget -O awslogs-agent-setup.py https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-setup.py
```

Note: This command fetches the aws-logs-agent-setup file from the S3 bucket where it's stored.

```
ubuntu@ip-10.0.0.1:~$ wget -O awslogs-agent-setup.py https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-setup.py
--2025-06-02 14:52:59-- https://s3.amazonaws.com/aws-cloudwatch/downloads/latest/awslogs-agent-setup.py
Resolving s3.amazonaws.com (s3.amazonaws.com)... 54.230.149.101
Connecting to s3.amazonaws.com (s3.amazonaws.com)|54.230.149.101|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 58225 (57K) [text/x-python]
Saving to: 'awslogs-agent-setup.py'

awslogs-agent-setup.py 100%[=====] 56.86K --.-KB/s in 0.001s

2025-06-02 14:52:59 (63.9 MB/s) - 'awslogs-agent-setup.py' saved [58225/58225]

ubuntu@ip-10.0.0.1:~$
```

```
sudo python ./awslogs-agent-setup.py --region us-east-1
```

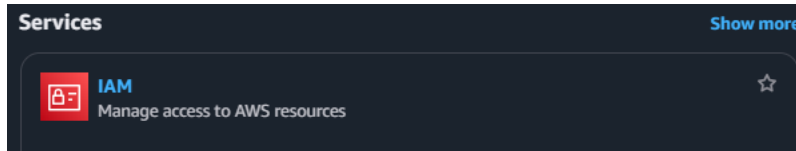
```
ubuntu@ip-10.0.0.1:~$ sudo python ./awslogs-agent-setup.py --region us-east-1
Launching interactive setup of CloudWatch Logs agent ...
downloading AgentDependencies.tar.gz with urllib
AgentDependencies/virtualenv-15.1.0/docs/makefile
AgentDependencies/virtualenv-15.1.0/docs/conf.py
AgentDependencies/virtualenv-15.1.0/docs/changes.rst
AgentDependencies/virtualenv-15.1.0/docs/installation.rst
AgentDependencies/virtualenv-15.1.0/docs/make.bat
AgentDependencies/pip-6.1.1.tar.gz

Step 1 of 5: Installing pip ...libyaml-dev does not exist in system python-dev does not exist in system DONE
Step 2 of 5: Downloading the latest CloudWatch Logs agent bits ... DONE
Step 3 of 5: Configuring AWS CLI ...
AWS Access Key ID [None]:
```

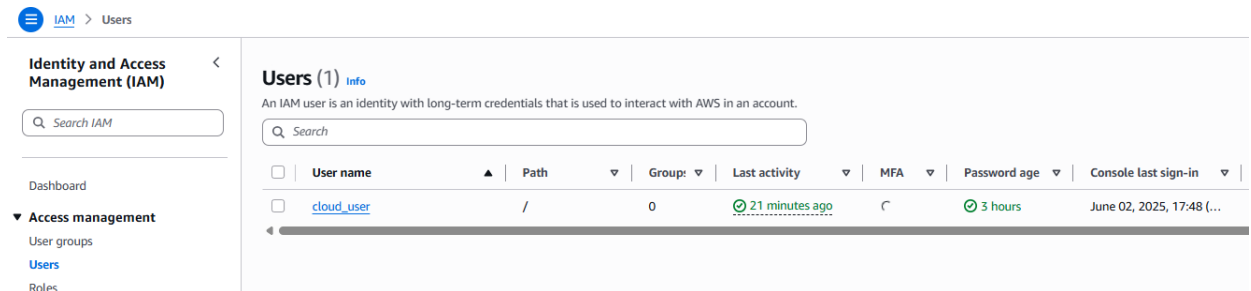
Note: This command executes the setup.py file and targets the server to the default region of us-east-1. Once a CloudWatch Logs agent has finished downloading, you'll move on to **step 3 of 5**, which is where you'll need an AWS access key and a secret access key to complete the configuration. Our cloud user account currently does not have an access key we can use.

2. Configure CloudWatch Logs

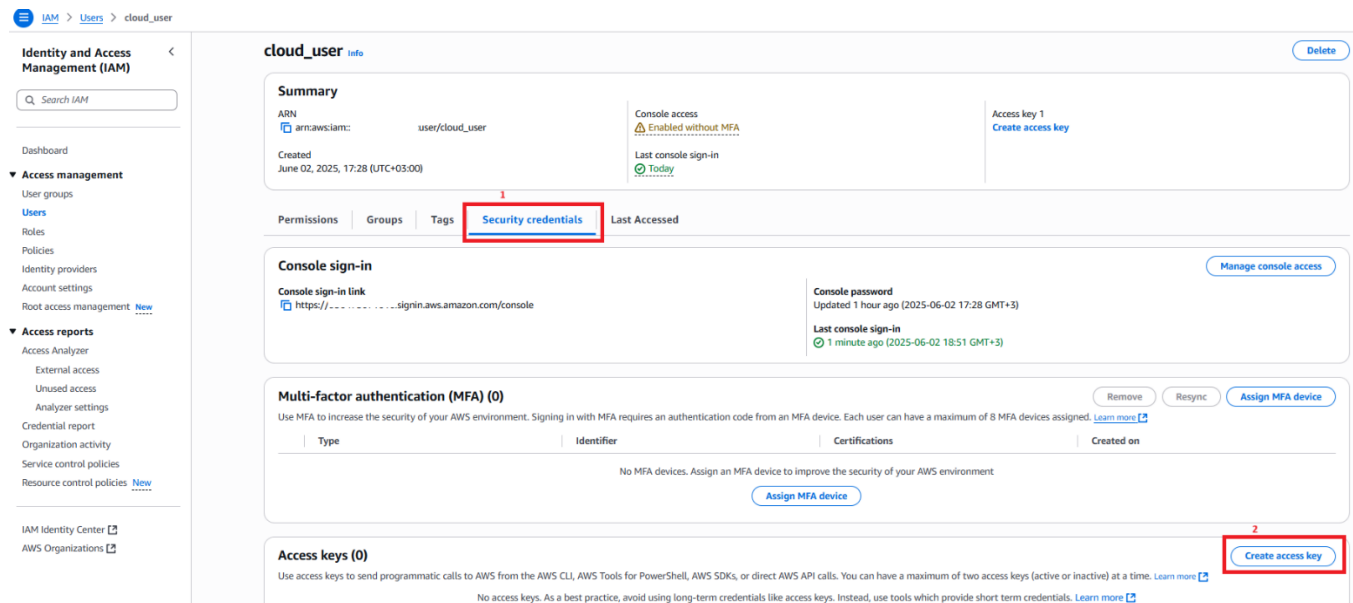
1. Go back to the AWS Management Console, and open **IAM** in a separate tab.



2. Click **Users**.
3. Click **cloud_user**.



4. Select the **Security credentials** tab.
5. Click **Create access key**.



6. Select **Command Line Interface (CLI)**.
7. Check the acknowledgment checkbox at the bottom of the page and click **Next**.

IAM > Users > cloud_user > Create access key

Step 1
☒ Access key best practices & alternatives
 Step 2 - optional
☐ Set description tag
 Step 3
☐ Retrieve access keys

Access key best practices & alternatives info

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

Use case

☒ **Command Line Interface (CLI)**
 You plan to use this access key to enable the AWS CLI to access your AWS account.

☐ **Local code**
 You plan to use this access key to enable application code in a local development environment to access your AWS account.

☐ **Application running on an AWS compute service**
 You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

☐ **Third-party service**
 You plan to use this access key to enable access for a third-party application or service that monitors or manages your AWS resources.

☐ **Application running outside AWS**
 You plan to use this access key to authenticate workloads running in your data center or other infrastructure outside of AWS that needs to access your AWS resources.

☐ **Other**
 Your use case is not listed here.

Alternatives recommended

- Use [AWS CloudShell](#), a browser-based CLI, to run commands. [Learn more](#)
- Use the [AWS CLI V2](#) and enable authentication through a user in IAM Identity Center. [Learn more](#)

Confirmation

☒ I understand the above recommendation and want to proceed to create an access key.

Cancel **Next**

8. For **Description tag value**, enter *clouduseraccesskey*.

9. Click **Create access key**.

Step 1
☐ Access key best practices & alternatives
 Step 2 - optional
☒ **Set description tag**
 Step 3
☐ Retrieve access keys

Set description tag - optional info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value
 Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidentially later.

clouduseraccesskey

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: _ . : / = + - @

Cancel Previous **Create access key**

10. Copy the **Access key**.

Access key created
 This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

Step 1
☐ Access key best practices & alternatives
 Step 2 - optional
☐ Set description tag
 Step 3
☒ **Retrieve access keys**

Retrieve access keys info

Access key
 If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key | Secret access key

AKIA VT | ***** **Show**

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

Download .csv file Done

11. Go back to the CLI (keep IAM open in the other tab), paste the **Access key ID**, and press **Enter**.


```
Step 3 of 5: Configuring AWS CLI ...
AWS Access Key ID [None]: AKIA
```

12. Go back to IAM and, in the **Secret access key** on the right, click **Show**. Then, copy the **Secret access key**.

Access key created
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

Step 1
● Access key best practices & alternatives

Step 2 - optional
● Set description tag

Step 3
● **Retrieve access keys**

Retrieve access keys [Info](#)

Access key
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIA VT	lvv 3k Hide

Access key best practices

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

[Download .csv file](#) [Done](#)

13. Go back to the CLI, paste the **Secret access key**, and press **Enter**.

```
AWS Secret Access Key [None]: lvn8
Default region name [us-east-1]:
```

14. Press **Enter** twice to accept the default region name and output format.

```
Default region name [us-east-1]:
Default output format [None]:

Step 4 of 5: Configuring the CloudWatch Logs Agent ...
Path of log file to upload [/var/log/syslog]:
```

15. Copy and paste the below path for the log file to upload:

/var/log/apache2/error.log

```
Step 4 of 5: Configuring the CloudWatch Logs Agent ...
Path of log file to upload [/var/log/syslog]: /var/log/apache2/error.log
Destination Log Group name [/var/log/apache2/error.log]:
```

16. Press **Enter** to keep the current Destination Log Group name.
17. Press **Enter** to accept the default Log Stream name.
18. Press **Enter** to keep the default Log Event timestamp format.
19. Press **1** to select **From start of file as** the initial position of upload.
20. Type **N** to complete the configuration.

```

Destination Log Group name [/var/log/apache2/error.log]:

Choose Log Stream name:
  1. Use EC2 instance id.
  2. Use hostname.
  3. Custom.
Enter choice [1]:

Choose Log Event timestamp format:
  1. %b %d %H:%M:%S (Dec 31 23:59:59)
  2. %d/%b/%Y:%H:%M:%S (10/Oct/2000:13:55:36)
  3. %Y-%m-%d %H:%M:%S (2008-09-08 11:52:54)
  4. Custom
Enter choice [1]:

Choose initial position of upload:
  1. From start of file.
  2. From end of file.
Enter choice [1]: 1
More log files to configure? [Y]: n

Step 5 of 5: Setting up agent as a daemon ...DONE

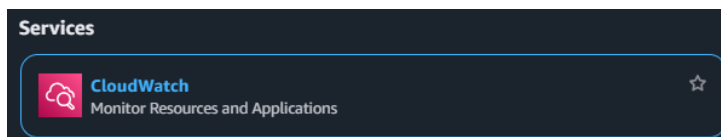
-----
- Configuration file successfully saved at: /var/awslogs/etc/awslogs.conf
- You can begin accessing new log events after a few moments at https://console.aws.amazon.com/cloudwatch/home?region=us-east-1#logs:
- You can use 'sudo service awslogs start|stop|status|restart' to control the daemon.
- To see diagnostic information for the CloudWatch Logs Agent, see /var/log/awslogs.log
- You can rerun interactive setup using 'sudo python ./awslogs-agent-setup.py --region us-east-1 --only-generate-config'
-----
ubuntu@ip-10-0-1-10:~$

```

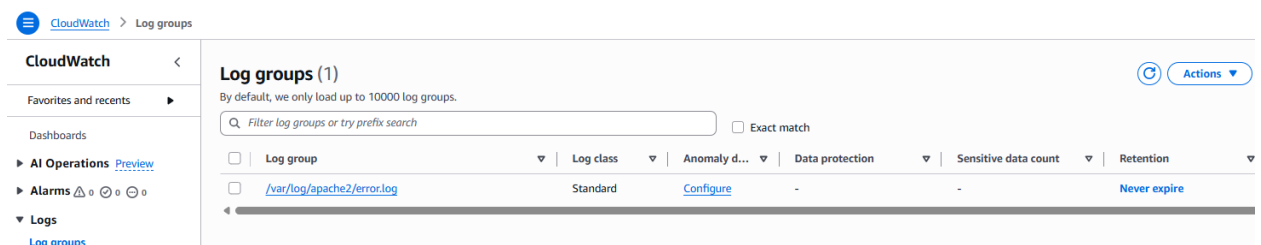
Note: The agent's been configured, you can access the log events through the URL provided or type CloudWatch in the search bar.

3. Log In to the CloudWatch Logs Website

1. In the search bar, enter **CloudWatch**, and right-click to open it in a new tab.



2. Click **Log groups**.
3. Click **/var/log/apache2/error.log**; if you don't see it yet, click the **Refresh** button.



4. Under **Log streams**, click the link for the instance identifier. You will see the contents of your error log with two events logged.

The screenshot shows the AWS CloudWatch console interface. The breadcrumb navigation at the top indicates the path: CloudWatch > Log groups > /var/log/apache2/error.log. The left sidebar contains navigation links for Dashboards, AI Operations, Alarms, Logs, Metrics, X-Ray traces, Events, Application Signals, Network Monitoring, and Insights. The main content area is titled '/var/log/apache2/error.log' and includes a 'Log group details' section with fields for Log class, ARN, Creation time, Retention, and Stored bytes. Below this, a horizontal tab bar shows 'Log streams' as the active tab. The 'Log streams (1)' section displays a single log stream named 'i-05' with a last event time of 2025-06-02 14:32:17 (UTC). The 'Log stream' checkbox is checked, and the 'i-05' label is highlighted with a red box.

As you can see, we already have two log file entries already uploaded to our log stream. We can confirm these are the only lines already in our log file.

The screenshot shows the 'Log events' view in the AWS CloudWatch console. The breadcrumb navigation is: CloudWatch > Log groups > /var/log/apache2/error.log > i-05. The 'Log events' section includes a search bar and a table of log events. The table has two columns: 'Timestamp' and 'Message'. The first event is from Mon Jun 02 14:32:17.934795 2025, with a message about Apache/2.4.41 (Ubuntu) configured. The second event is from Mon Jun 02 14:32:17.934923 2025, with a message about the command line: '/usr/sbin/apache2'.

- You can also view the contents of the error log in the CLI by running the following command:

sudo cat /var/log/apache2/error.log

```
ubuntu@ip-10-0-1-10:~$ sudo cat /var/log/apache2/error.log
Mon Jun 02 14:32:17.934795 2025] [mpm_event:notice] [pid 1855:tid 139865400147008] AH00489: Apache/2.4.41 (Ubuntu) configured -- resuming normal operations
Mon Jun 02 14:32:17.934923 2025] [core:notice] [pid 1855:tid 139865400147008] AH00094: Command line: '/usr/sbin/apache2'
```

We can see here we have the same two log file lines as we've already been displayed in our log stream.