Send Apache Logs To Cloudwatch Log Group
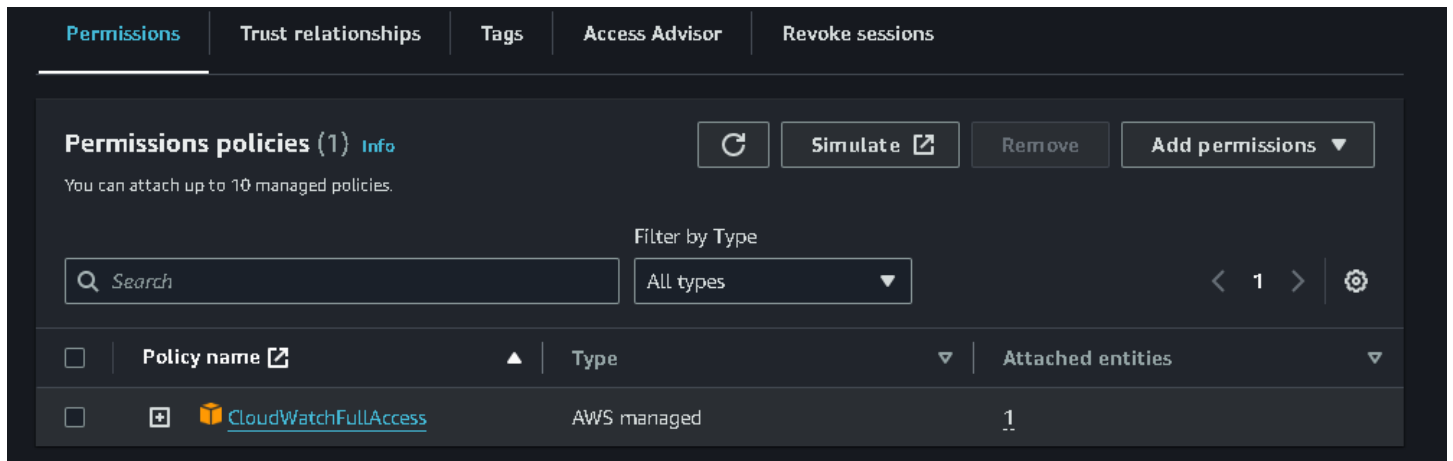
Dushyant Kumar

- **Create Iam role with policy attached to it.**

- **Create a Instance and attach an Iam role.**

- **Now run some commands**

Update and install apache in machine:
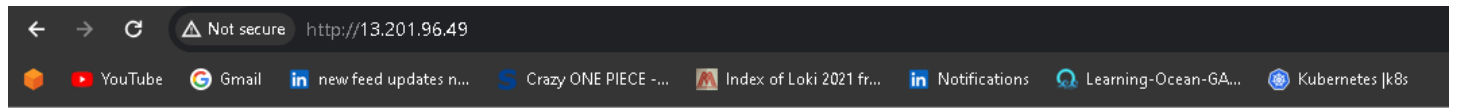
sudo apt update -y && sudo apt install apache2 -y

After the installation is complete, start the Apache service and enable it to start on boot:

sudo systemctl start apache2

sudo systemctl enable apache2

Check the output in web browser.

Download cloudwatch agent package

wget https://s3.amazonaws.com/amazoncloudwatch-agent/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb

```
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# pwd
/home/ubuntu/cloudwatch
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# ll
total 8
drwxr-xr-x 2 root   root   4096 Jan 24 18:08 ./
drwxr-x--- 5 ubuntu ubuntu 4096 Jan 24 18:08 ../
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# wget https://s3.amazonaws.com/amazoncloudwatch-agent/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb
--2024-01-24 18:13:15--  https://s3.amazonaws.com/amazoncloudwatch-agent/ubuntu/amd64/latest/amazon-cloudwatch-agent.deb
Resolving s3.amazonaws.com (s3.amazonaws.com) ... 52.216.33.88, 52.216.138.85, 52.216.222.8, ...
Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.216.33.88|:443 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 91736066 (87M) [application/octet-stream]
Saving to: 'amazon-cloudwatch-agent.deb'

amazon-cloudwatch-agent.deb      100%[===================================================================================>]  87.49M  7.93MB/s    in 12s

2024-01-24 18:13:28 (7.20 MB/s) - 'amazon-cloudwatch-agent.deb' saved [91736066/91736066]

root@ip-172-31-38-57:/home/ubuntu/cloudwatch# ls
amazon-cloudwatch-agent.deb
```

Install the CloudWatch Agent

sudo dpkg -i amazon-cloudwatch-agent.deb

```
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo dpkg -i amazon-cloudwatch-agent.deb
Selecting previously unselected package amazon-cloudwatch-agent.
(Reading database ... 65568 files and directories currently installed.)
Preparing to unpack amazon-cloudwatch-agent.deb ...
create group cwagent, result: 0
create user cwagent, result: 0
Unpacking amazon-cloudwatch-agent (1.300032.3b392-1) ...
Setting up amazon-cloudwatch-agent (1.300032.3b392-1) ...
```

configure cloudwatch agent

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

```
Do you want to monitor any log files?
1. yes
2. no
default choice: [1]:
1
Log file path:
/var/log/apache2/access.log
Log group name:
default choice: [access.log]
apache_logs
Log group class:
1. STANDARD
2. INFREQUENT_ACCESS
default choice: [1]:
1
Log stream name:
default choice: [{instance_id}]
apache_stream
```

for ubuntu debian, log file path for apache2

/var/log/apache2/access.log   /var/log/apache2/error.log

Note: also add the logstream for both path file and put in the same loggroup file

log group name : apache/logs

access stream name : apache_access_stream

log stream name : apache_error_stream

In additonal log path enter same path for error logs and enter same group name but diffrent stream name.

File path for access and error log are shown above.

systemctl start amazon-cloudwatch-agent

systemctl status-cloudwatch-agent.service

```
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo systemctl status amazon-cloudwatch-agent.service
○ amazon-cloudwatch-agent.service - Amazon CloudWatch Agent
     Loaded: loaded (/etc/systemd/system/amazon-cloudwatch-agent.service; enabled; vendor preset: enabled)
     Active: inactive (dead)

Jan 24 18:27:21 ip-172-31-38-57 systemd[1]: Started Amazon CloudWatch Agent.
Jan 24 18:27:21 ip-172-31-38-57 start-amazon-cloudwatch-agent[3201]: D! [EC2] Found active network interface
Jan 24 18:27:21 ip-172-31-38-57 start-amazon-cloudwatch-agent[3201]: I! imds retry client will retry 1 timesI! Detected the instance is EC2
Jan 24 18:27:21 ip-172-31-38-57 start-amazon-cloudwatch-agent[3201]: 2024/01/24 18:27:21 Reading json config file path: /opt/aws/amazon-cloudwatch-agent/etc/amaz
Jan 24 18:27:21 ip-172-31-38-57 start-amazon-cloudwatch-agent[3201]: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.json does not exist or cannot r
Jan 24 18:27:21 ip-172-31-38-57 systemd[1]: amazon-cloudwatch-agent.service: Deactivated successfully.
lines 1-10/10 (END)
```

sudo apt-get install collectd

```
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo apt-get install collectd
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
The following additional packages will be installed:
  alsa-topology-conf alsa-ucm-conf ca-certificates-java collectd-core default-jre-headless fontconfig fontconfig-config fonts-dejavu-core intel-cmt-cat
  java-common libabsl20210324 libasound2 libasound2-data libavahi-client3 libavahi-common-data libavahi-common3 libbson-1.0-0 libc-ares2 libcairo2 libcups2
  libdatrie1 libdbi1 libdeflate0 libesmtp6 libfdt1 libfontconfig1 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin libgdk-pixbuf2.0-common libgps28 libgraphite2-3
  libgrpc++1 libgrpc10 libharfbuzz0b libhiredis0.14 libi2c0 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 libmemcached11 libmicrohttpd12 libmodbus5
  libmongoc-1.0-0 libmongocrypt0 libmosquitto1 libmysqlclient21 libnotify4 libopenipmi0 liboping0 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
  libpcsclite1 libpixman-1-0 libpq5 libprotobuf-c1 libprotobuf23 libqpid-proton11 librabbitmq4 librdkafka1 libriemann-client0 librrd8 librte-eal22
  librte-ethdev22 librte-kvargs22 librte-mbuf22 librte-mempool22 librte-meter22 librte-net22 librte-ring22 librte-telemetry22 libsensors-config libsensors5
  libsnappy1v5 libsnmp-base libsnmp40 libthai-data libthai0 libtiff5 libvarnishapi2 libvirt0 libwebp7 libxcb-render0 libxcb-shm0 libxencall1 libxendevicemodel1
  libxenevtchn1 libxenforeignmemory1 libxengnttab1 libxenhypfs1 libxenmisc4.16 libxenstore4 libxentoolcore1 libxentoollog1 libxrender1 libyajl2 mysql-common
  openjdk-11-jre-headless rrdtool
Suggested packages:
```

sudo service collectd start

sudo service collectd enable

systemctl status collectd.service

```
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo service collectd start
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo service collectd start
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# sudo service collectd enable
Usage: /etc/init.d/collectd {start|stop|restart|force-reload|status}
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# systemctl status collectd.service
● collectd.service - Statistics collection and monitoring daemon
     Loaded: loaded (/lib/systemd/system/collectd.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2024-01-24 18:31:18 UTC; 1min 17s ago
       Docs: man:collectd(1)
             man:collectd.conf(5)
             https://collectd.org
    Process: 4756 ExecStartPre=/usr/sbin/collectd -t (code=exited, status=0/SUCCESS)
   Main PID: 4757 (collectd)
      Tasks: 12 (limit: 1121)
     Memory: 30.2M
        CPU: 179ms
     CGroup: /system.slice/collectd.service
             └─4757 /usr/sbin/collectd

Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "irq" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "load" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "memory" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "processes" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "rrdtool" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "swap" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: plugin_load: plugin "users" successfully loaded.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: Systemd detected, trying to signal readiness.
Jan 24 18:31:18 ip-172-31-38-57 systemd[1]: Started Statistics collection and monitoring daemon.
Jan 24 18:31:18 ip-172-31-38-57 collectd[4757]: Initialization complete, entering read-loop.
root@ip-172-31-38-57:/home/ubuntu/cloudwatch#
```

systemctl start amazon-cloudwatch-agent

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-172-31-38-57:/home/ubuntu/cloudwatch# systemctl start amazon-cloudwatch-agent
root@ip-172-31-38-57:/home/ubuntu/cloudwatch# 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
****** 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 in /o
pt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp
Start configuration validation ...
2024/01/24 18:34:38 Reading json config file path: /opt/aws/amazon-cloudwatch-agent/etc/amazon-cloudwatch-agent.d/file_config.json.tmp ...
2024/01/24 18:34:38 I! Valid Json input schema.
2024/01/24 18:34:38 D! ec2tagger processor required because append_dimensions is set
2024/01/24 18:34:38 D! pipeline hostDeltaMetrics has no receivers
2024/01/24 18:34:38 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
root@ip-172-31-38-57:/home/ubuntu/cloudwatch#
```
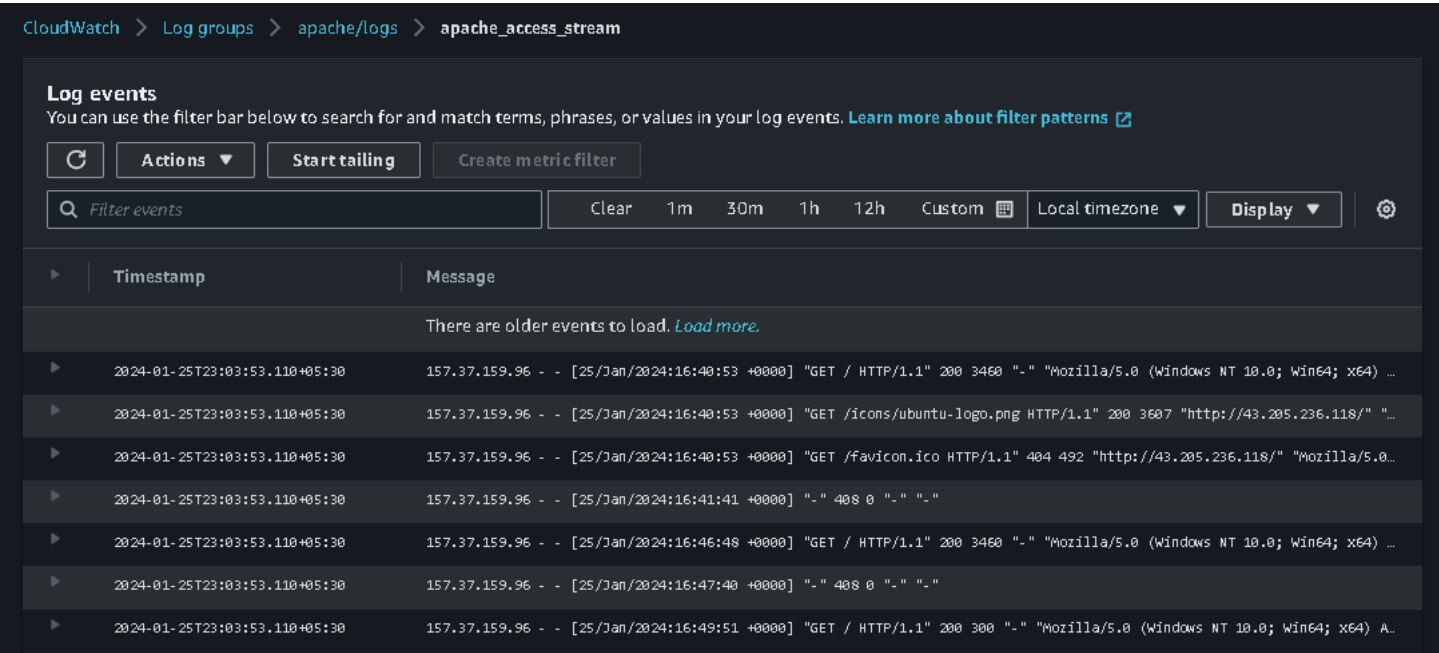
Now, we get the log group (apache/logs) in cloudwatch
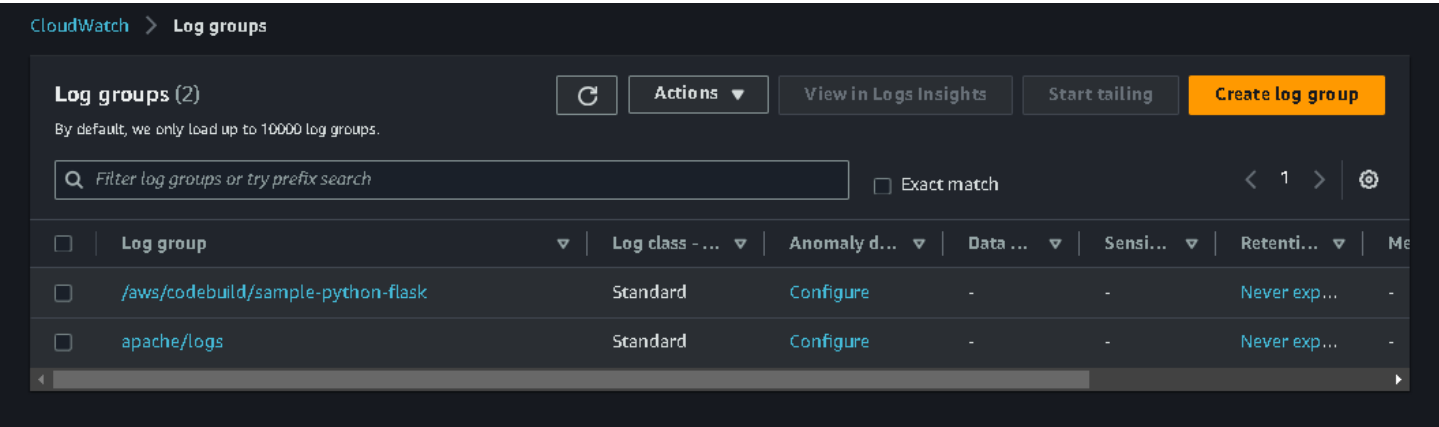
Here we get two log streams apache_access_stream and apache_error_stream
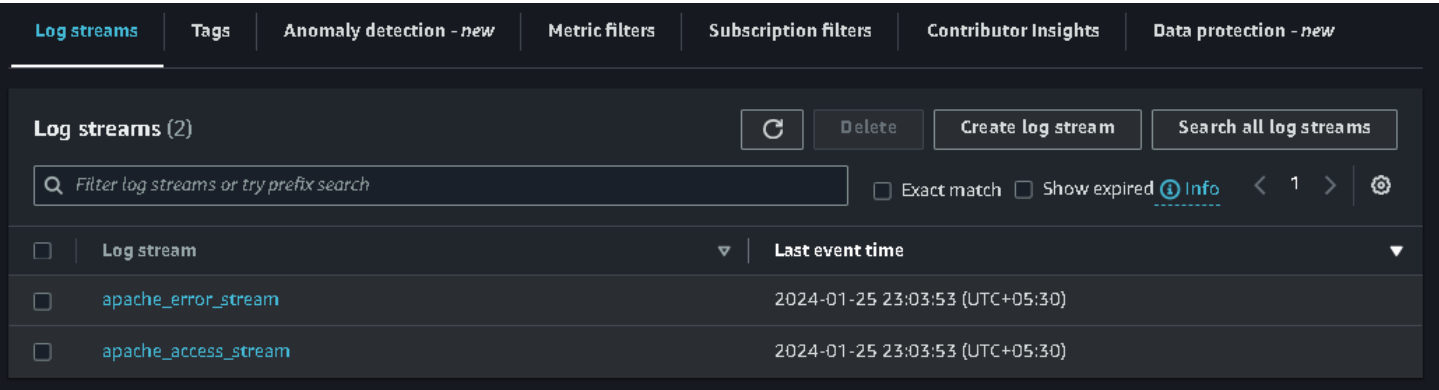
Apache access logs in cloudwatch

Apache error logs

Log events
You can use the filter bar below to search for and match terms, phrases, or values in your log events. **Learn more about filter patterns** [↗]

| Actions ▼ | Start tailing | Create metric filter |

| Q Filter events | | Clear | 1m | 30m | 1h | 12h | Custom ⊞ | Local timezone ▼ | Display ▼ | ⚙ |

►	Timestamp	Message
		There are older events to load. *Load more.*
►	2024-01-25T23:03:53.114+05:30	[Thu Jan 25 16:39:40.771737 2024] [mpm_event:notice] [pid 2319:tid 140279385950080] AH00489: Apache/2.4.52 (Ubuntu) configu…
►	2024-01-25T23:03:53.114+05:30	[Thu Jan 25 16:39:40.771835 2024] [core:notice] [pid 2319:tid 140279385950080] AH00094: Command line: '/usr/sbin/apache2'
►	2024-01-25T23:03:53.114+05:30	[Thu Jan 25 16:49:34.924751 2024] [mpm_event:notice] [pid 2319:tid 140279385950080] AH00492: caught SIGWINCH, shutting down…
►	2024-01-25T23:03:53.114+05:30	[Thu Jan 25 16:49:34.987761 2024] [mpm_event:notice] [pid 3074:tid 140013938501504] AH00489: Apache/2.4.52 (Ubuntu) configu…
		No more records within selected time range *Auto retry paused.* *Resume*

Installing and running the CloudWatch agent on your servers

Monitor your AWS services, collect and track metrics from AWS services and your own applications, and set alarms...

📦 Amazon CloudWatch

To get config json file:

cat /opt/aws/amazon-cloudwatch-agent/bin/config.json

{

    "agent": {

        "metrics_collection_interval": 60,

        "run_as_user": "root"

```json
    },
    "logs": {

        "logs_collected": {

            "files": {

                "collect_list": [

                    {

                        "file_path": "/var/log/apache2/access.log",

                        "log_group_class": "STANDARD",

                        "log_group_name": "apache/logs",

                        "log_stream_name": "apache_access_stream",

                        "retention_in_days": -1

                    },

                    {

                        "file_path": "/var/log/apache2/error.log",

                        "log_group_class": "STANDARD",

                        "log_group_name": "apache/logs",

                        "log_stream_name": "apache_error_stream",

                        "retention_in_days": -1

                    }

                ]

            }

        }

    },

    "metrics": {

        "aggregation_dimensions": [
```

```json
        [

            "InstanceId"

        ]

    ],

    "append_dimensions": {

        "AutoScalingGroupName": "${aws:AutoScalingGroupName}",

        "ImageId": "${aws:ImageId}",

        "InstanceId": "${aws:InstanceId}",

        "InstanceType": "${aws:InstanceType}"

    },

    "metrics_collected": {

        "disk": {

            "measurement": [

                "used_percent"

            ],

            "metrics_collection_interval": 60,

            "resources": [

                "*"

            ]

        },

        "mem": {

            "measurement": [

                "mem_used_percent"

            ],

            "metrics_collection_interval": 60
```

```
        }

    }

}
```