LOGZILLA DOCUMENTATION

AWS CloudWatch and Kinesis Setup

Deliver AWS CloudWatch logs to LogZilla using Kinesis Data Firehose over HTTPS to the /firehose endpoint with access token authentication

Receiving Data in LogZilla · Generated April 27, 2026 · logzilla.ai/docs/receiving-data/aws-cloudwatch-kinesis-setup

AWS CloudWatch and Kinesis Data Firehose

AWS CloudWatch logs can be forwarded to LogZilla using Kinesis Data Firehose over HTTP/HTTPS. Firehose delivers logs to LogZilla's `/firehose` endpoint with authentication via access tokens.

Prerequisites

- A LogZilla access token, refer to the LogZilla API documentation (https://www.logzilla.ai/docs/logzilla-api).
- AWS account with CloudWatch logs and Kinesis Data Firehose access.
- Network access from AWS to the LogZilla server on HTTP/HTTPS ports.

Configure Kinesis Data Firehose

Create delivery stream

In the AWS Console, navigate to Kinesis Data Firehose.
Select `Create delivery stream`.
Set:

- Source: `Direct PUT` or `Kinesis Data Streams`
- Destination: `HTTP Endpoint`

Enter a delivery stream name (for example, `logzilla-firehose`).

Configure destination settings

Set the HTTP endpoint details:

- HTTP endpoint URL: `https://LOGZILLA_HOST/firehose`
- Access key: the LogZilla access token
- Content encoding: `GZIP` (recommended)
- Retry duration: `3600` seconds (or as required)

Configure backup settings

Select an S3 bucket for failed delivery backup. This is required by Firehose for error handling and replay.

Create the stream

Review settings and create the delivery stream. AWS will validate the endpoint and begin processing.

Connect CloudWatch logs

Create subscription filter

In CloudWatch, select the log group to forward.
Create a subscription filter:

- Destination: the Kinesis Data Firehose delivery stream
- Filter pattern: leave empty to forward all logs, or specify a pattern
- Role: create or select an IAM role with Firehose write permissions

Verify delivery

Check LogZilla reception

Search LogZilla for events with:

- Program: `firehose` (default for Firehose events)
- Recent timestamps matching the CloudWatch log entries

Note: If a LogZilla app is installed for the log type (such as AWS VPC Flow), the app may set different program names and add Vendor/Product tags.

Test with curl

Send a test event to verify the `/firehose` endpoint:

```
url="https://LOGZILLA_HOST/firehose"
apikey="YOUR_TOKEN_HERE"

# Create test payload (CloudWatch log format)
payload='{"logGroup":"/test/logs","logStream":"test-stream","logEvents":[{"message":"Test log entry
from curl"}]}'
test_data=$(echo "$payload" | gzip | base64 -w 0)

curl -X POST "$url" \
  -H "Content-Type: application/json" \
```

```
  -H "X-Amz-Firehose-Access-Key: $apikey" \
  -d "{\"requestId\": \"test-123\", \"records\": [{\"data\": \"$test_data\"}]}"
```

The test event should appear in LogZilla with program `firehose`.

Troubleshooting

Common issues

- **403 Forbidden**: Check the access token and ensure it matches the Firehose configuration.
- **404 Not Found**: Verify the endpoint URL uses `/firehose` (not `/incoming`).
- **No events in LogZilla**: Check CloudWatch subscription filter and Firehose delivery metrics in AWS.

Firehose delivery metrics

Monitor Firehose delivery in the AWS Console:

- Delivery success/failure rates
- Error logs for failed deliveries
- S3 backup bucket for failed events

Network troubleshooting

For packet-level checks, see Testing And Verification (https://www.logzilla.ai/docs/data-transforms/testing-and-verification).

AWS VPC Flow Logs

For AWS VPC Flow Logs, install the **AWS CloudWatch VPC Flow** app from the LogZilla appstore (`Settings` → `App store`).
The app provides specialized parsing, dashboards, and security triggers for VPC flow log data.

See the app's documentation for configuration details and features.