

Setting up VPC Flow logs

With CloudWatch and S3+Athena

The screenshot shows the AWS VPC dashboard under the 'Your VPCs' section. A specific VPC, 'DemoProjectVPC', is selected. In the 'Flow logs' tab, two flow logs are listed: 'DP_FlowLogs_S3' and 'DP_FlowLogs_CW'. The 'Flow logs' tab is highlighted with a red oval. The 'CW' log is associated with the CloudWatch Logs log group 'DP_FlowLogs_LogGroup'.

Short Summary:

Flow logs With S3+Athena

- To use with Athena, with this we can eliminate ETL(Extract, transform, load) pipelines, infrastructure setup, scaling, and ongoing management (but we still need to define a schema in Athena), which makes log analysis far simpler and faster because we can directly query flow logs without building heavy data pipelines.

Usecase: when you want **deep analysis, reporting, and trend insights** over time.

Flow logs With CloudWatch

- For log analysis – Observability purpose

Usecase: when you want **live monitoring and troubleshooting**.

Flow logs With S3

1. Create an S3 bucket in the Same region as VPC

The screenshot shows the AWS S3 bucket properties page for 'dp-vpc-flow-logs-1'. The bucket name is highlighted with a red oval. The 'Properties' tab is selected. Key details shown include:

- AWS Region: Asia Pacific (Singapore) ap-southeast-1
- Amazon Resource Name (ARN): arn:aws:s3:::dp-vpc-flow-logs-1
- Creation date: January 15, 2026, 18:31:07 (UTC+05:30)

Other tabs like 'Objects', 'Metadata', 'Permissions', 'Metrics', 'Management', and 'Access Points' are also visible.

2. Create Flow logs from VPC

Flow log created with destination: S3 Bucket

The screenshot shows the AWS VPC Flow Logs Details page for a flow log named "fl-0cd681af17bb21628 / DP_FlowLogs_S3". The "Destination Type" section is highlighted with a red oval, specifically the "Destination Name" field which contains "dp-vpc-flow-logs-1". Other settings shown include "Traffic Type: All", "File Format: Plain text", and "Log Format: Default". The "Tags" section shows a single tag "Name: DP_FlowLogs_S3".

Using Athena with the logs stored in S3

1. Creating an S3 bucket to store the query results of Athena

The screenshot shows the AWS S3 Buckets page. A green success message at the top states "Successfully created folder 'Athena_DataQueries_data'." Below it, the "athena-query-data-bucket-1" bucket is listed. The "Objects" tab is selected, showing one object named "Athena_DataQueries_data/". The "Actions" dropdown menu is open, with "Upload" highlighted.

2. Mapping S3 bucket in the query location of Athena

The screenshot shows the 'Query settings' tab in the Amazon SageMaker Unified Studio. A modal window titled 'Manage query result location and encryption' is open. In the 'Location of query result - optional' section, the value 's3://athena-query-data-buck' is entered. Below it, there's a note about creating lifecycle rules and a 'Lifecycle configuration' link. In the 'Expected bucket owner - optional' section, there's a field to enter an AWS account ID and two checkboxes: 'Assign bucket owner full control over query results' (unchecked) and 'Encrypt query results' (unchecked). At the bottom right of the modal are 'Cancel' and 'Save' buttons.

3. Querying logs from S3 with Athena | Querylink: [Create a table for Amazon VPC flow logs and query it - Amazon Athena](#)

a. First creating a Structured Table(SQL) from the logs stored in S3

The screenshot shows the Amazon Athena Query Editor. On the left, the 'Data' sidebar shows the 'Tables and views' section with a table named 'vpc_flow_logs'. The table has columns: version (int), account_id (string), interface_id (string), srcaddr (string), dstaddr (string), srcport (int), dstport (int), protocol (bigint), packets (bigint), bytes (bigint), start (bigint), and end (bigint). The main editor area shows the SQL code for creating the table:

```

1 - CREATE EXTERNAL TABLE IF NOT EXISTS `vpc_flow_logs` (
2   version int,
3   account_id string,
4   interface_id string,
5   srcaddr string,
6   dstaddr string,
7   srcport int,
8   dstport int,
9   protocol bigint,
10  packets bigint,
11  bytes bigint,
12  start bigint,
13  `end` bigint,
14  action string,
15  log_status string,

```

Below the code, the status bar indicates: SQL Ln 35, Col 83, Run again, Explain, Cancel, Clear, Create, and Reuse query results up to 60 minutes ago.

b. Partitioning the table by time (day or hour is recommended for large datasets, month is fine for demo)

The screenshot shows the Amazon Athena Query Editor with three tabs open: 'Creating table from Logs i...', 'Adding month partition t...', and 'Select query - to check th...'. The main editor area contains the following SQL code:

```

1 ALTER TABLE vpc_flow_logs
2 ADD PARTITION (`date`='2026-01-15')
3 LOCATION 's3://dp-vpc-flow-logs-1/AWSLogs/438465149928/vpcflowlogs/ap-southeast-1/2026/01/15/';
4
5

```

Below the code, the status bar indicates: SQL Ln 4, Col 1, Run, Explain, Cancel, Clear, Create, and Reuse query results up to 60 minutes ago.

Result:

Tables (1)		
		< 1 >
instance_id	string	:
tcp_flags	int	:
type	string	:
pkt_srcaddr	string	:
pkt_dstaddr	string	:
region	string	:
az_id	string	:
sublocation_type	string	:
sublocation_id	string	:
pkt_src_aws_service	string	:
pkt_dst_aws_service	string	:
flow_direction	string	:
traffic_path	int	:
date	date (Partitioned)	:

c. Checking the logs with accepted action

The screenshot shows the Amazon Athena Query Editor interface. On the left, the Data sidebar displays the configuration for the query, including the Data source (AwsDataCatalog), Catalogue (None), and Database (default). Below it, the Tables (1) section shows the schema of the 'vpc_flow_logs' table, which includes columns like instance_id, tcp_flags, type, pkt_srcaddr, pkt_dstaddr, region, az_id, sublocation_type, sublocation_id, pkt_src_aws_service, pkt_dst_aws_service, flow_direction, traffic_path, and date.

The main area contains three tabs: 'Creating table from Logs i...', 'Adding month partition t...', and 'Select query - to check th...'. The third tab is active, showing the SQL query:

```
1 SELECT day_of_week(date) AS
2   day,
3   date,
4   interface_id,
5   srcaddr,
6   action,
7   protocol
8 FROM vpc_flow_logs
9 WHERE action = 'ACCEPT' AND protocol = 6
10 LIMIT 100;
```

Below the query, the results are displayed in a table titled 'Results (100)'. The table has the following columns: #, day, date, interface_id, srcaddr, action, and protocol. One row is shown, corresponding to the query results:

#	day	date	interface_id	srcaddr	action	protocol
1	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6

Results (100)						
#	day	date	interface_id	srcaddr	action	protocol
1	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
2	4	2026-01-15	eni-05f05e980dd47e002	45.78.219.190	ACCEPT	6
3	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
4	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
5	4	2026-01-15	eni-05f05e980dd47e002	61.222.211.114	ACCEPT	6
6	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
7	4	2026-01-15	eni-05f05e980dd47e002	47.128.4.216	ACCEPT	6
8	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
9	4	2026-01-15	eni-05f05e980dd47e002	91.224.92.74	ACCEPT	6
10	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
11	4	2026-01-15	eni-05f05e980dd47e002	209.38.84.119	ACCEPT	6
12	4	2026-01-15	eni-05f05e980dd47e002	10.0.0.97	ACCEPT	6
13	4	2026-01-15	eni-015eed3c53e08adf8	10.0.2.181	ACCEPT	6
14	4	2026-01-15	eni-015eed3c53e08adf8	10.0.2.181	ACCEPT	6
15	4	2026-01-15	eni-015eed3c53e08adf8	10.0.2.181	ACCEPT	6

Flowlogs With CloudWatch

1. Create an IAM role with Custom policy(When delivering flow logs to CloudWatch, we must explicitly create and attach an IAM role that VPC Flow Logs can assume.) and “[CloudWatchFullAccess](#)”

Custom Trust policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Principal": {
                "service": "vpc-flow-logs.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Role DP_FlowLogsRole_CWAccess created.

DP_FlowLogsRole_CWAccess

Summary

Creation date: January 15, 2026, 19:06 (UTC+05:30)

ARN: arn:aws:iam::438465149928:role/DP_FlowLogsRole_CWAccess

Last activity: -

Maximum session duration: 1 hour

Permissions | Trust relationships | Tags (1) | Last Accessed | Revoke sessions

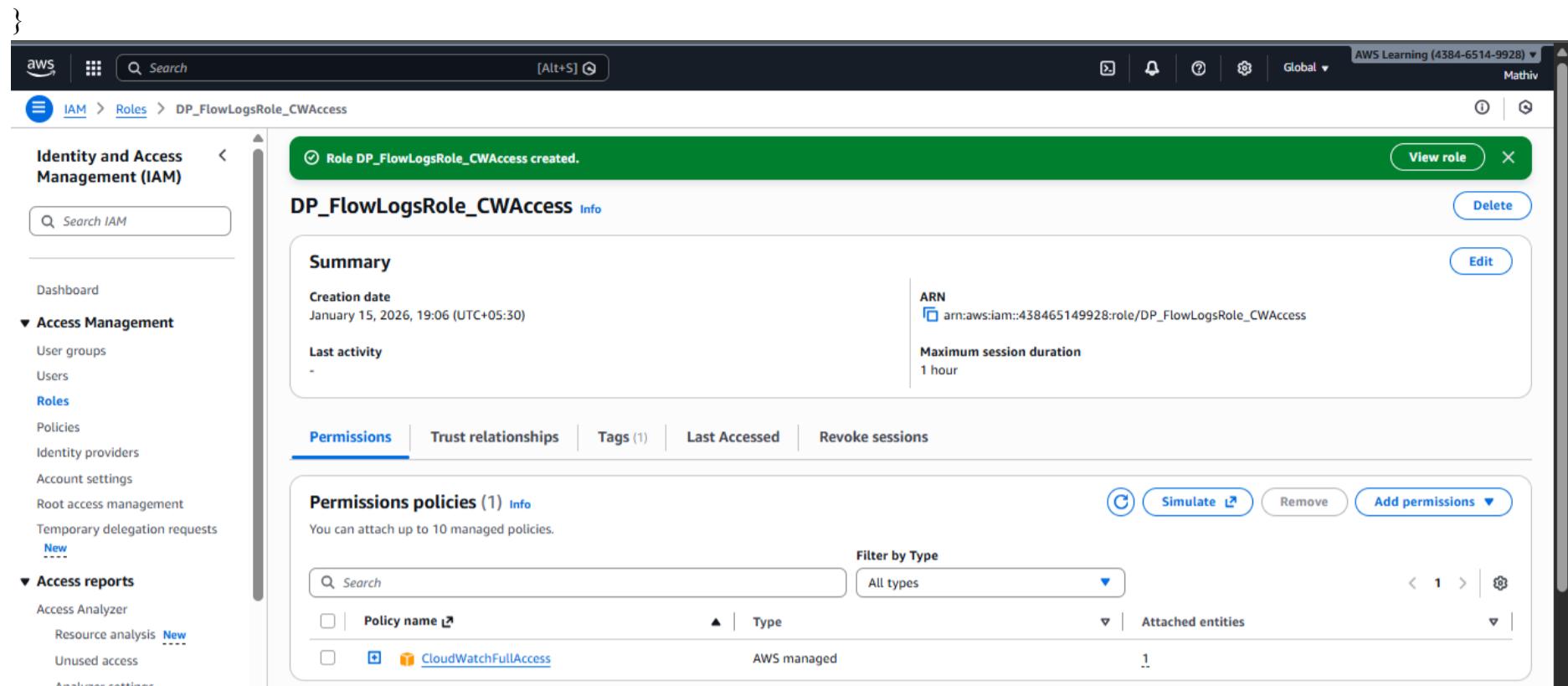
Permissions policies (1)

You can attach up to 10 managed policies.

Filter by Type: All types

Policy name	Type	Attached entities
CloudWatchFullAccess	AWS managed	1

Actions | **Simulate** | **Remove** | **Add permissions**



2. Create log group

A log group has been created with 1-day retention time (for demo; production usually uses longer retention)

Log group "DP_FlowLogs_LogGroup" has been created.

DP_FlowLogs_LogGroup

Log group details

Log class: Standard	Metric filters: 0	Data protection: -
ARN: arn:aws:logs:ap-southeast-1:438465149928:log-group:DP_FlowLogs_LogGroup:*	Subscription filters: 0	Sensitive data count: -
Creation time: 4 minutes ago	Contributor Insights rules: -	Custom field indexes: Configure
Retention: 1 day	KMS key ID: -	Transformer: Configure
Stored bytes: -	Deletion protection: <input checked="" type="checkbox"/> Off	Anomaly detection: Configure

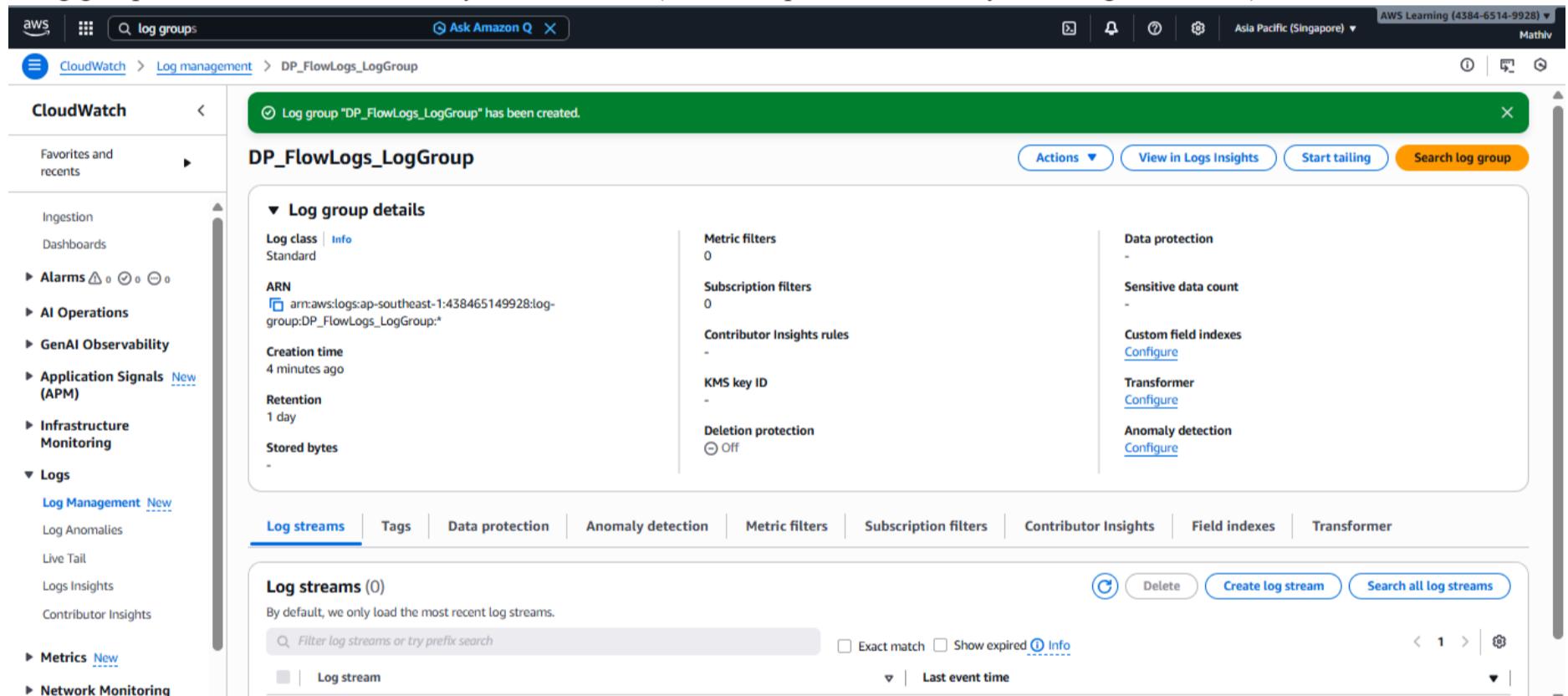
Log streams | Tags | Data protection | Anomaly detection | Metric filters | Subscription filters | Contributor Insights | Field indexes | Transformer

Log streams (0)

By default, we only load the most recent log streams.

Filter log streams or try prefix search: Exact match Show expired [Info](#)

Actions | **Delete** | **Create log stream** | **Search all log streams**



3. Flow log created with destination: CloudWatch

- attaching the created IAM role provides access to CloudWatch

Once the flow log has been set up with cloud watch, we could see the server activities are logging in the CloudWatch log group:

Note:

We could see many requests to our servers being rejected—these are unsolicited requests (bots, scanners, crawlers) blocked by security groups/NACLs, as intended.

The accepted requests are from AWS – we could confirm that by checking the IP address

Whois IP 54.197.201.248



Domains Hosting Servers Email Security Whois Deals

Enter Dom WHOIS



```
# start

NetRange:      54.144.0.0 - 54.221.255.255
CIDR:         54.160.0.0/11, 54.216.0.0/14, 54.144.0.0/12, 54.208.0.0/13, 54
NetName:       AMAZON
NetHandle:     NET-54-144-0-0-1
Parent:        NET54 (NET-54-0-0-0-0)
NetType:       Direct Allocation
OriginAS:
Organization:  Amazon Technologies Inc. (AT-88-Z)
RegDate:       2014-10-23
Updated:       2021-02-10
Ref:          https://rdap.arin.net/registry/ip/54.144.0.0
```

```
OrgName:       Amazon Technologies Inc.
OrgId:         AT-88-Z
Address:       410 Terry Ave N.
City:          Seattle
StateProv:     WA
PostalCode:    98109
Country:       US
RegDate:       2011-12-08
Updated:       2024-01-24
Comment:       All abuse reports MUST include:
Comment:       * src IP
```

.ROCKS

.ROCKS @ \$3.98 \$24.88

Introducing
WORDPRESS HOSTING

\$5.48 /mo

[VIEW MORE](#)