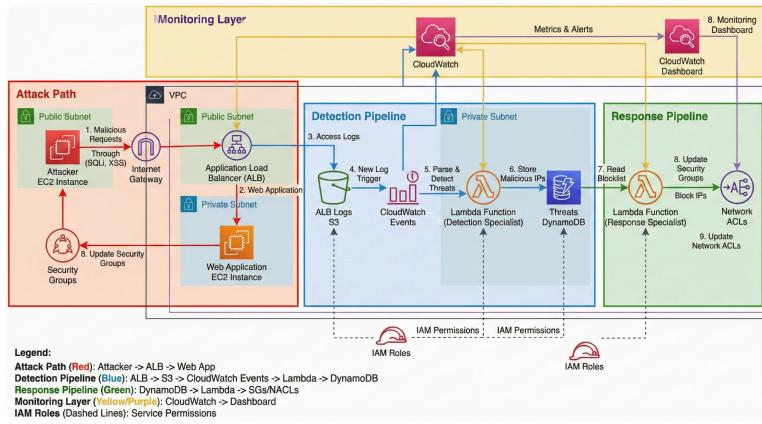


IDS architecture plan

December 11, 2025 7:47 PM

Suhaiib Kashif created the group chat.

Suhaiib Kashif 7:47 PM



This is an architectural diagram of an IDS named: Towards an Automated Defense: A Novel Framework for Log-Based Threat Detection and IP Remediation in AWS

is it good?

how to build it?

gimme a plan to build it.

gimme instructions to build it.



ChatGPT

Short answer: yes — the diagram is a solid, realistic design for a log-based IDS + automated IP remediation in AWS. It shows clear separation of Attack/Detection/Response/Monitoring, uses appropriate AWS primitives (ALB → S3 logs → EventBridge/CloudWatch → Lambda → DynamoDB → Lambda → Security controls), and is operationally sensible.

Below I'll (1) point out strengths and gaps, (2) give a phased build plan, and (3) give concrete, copy-pasteable instructions and code snippets you can use to implement each piece.

Show more

 Suhaib Kashif

If the architectural diagram is good then tell me that how would I build it on AWS?

I'm thinking of connecting it to a dataset and then the IDS runs through a pipeline.

SK

ChatGPT

Great — you're thinking in the right direction. Below I'll give you a practical, end-to-end plan for *building your diagram on AWS* and hooking it into a dataset so detection can be rule-based **and/or** ML-based (pipeline). You'll get: architecture choices, step-by-step build plan, concrete code snippets (preprocessing, SageMaker training, Lambda inference), deployment notes, testing & monitoring, and operational caveats.

I'll assume you already have ALB logs arriving to S3 (or can configure that). If not, the plan includes that step.

Show more

December 11, 2025 8:46 PM

 Suhaib Kashif

if the architecture is good then tell me that how would I implement it on AWS considering I'm going to take the dataset from somewhere like Kaggle.

SK

ChatGPT

Yes — the architecture is **good**, and it matches real-world AWS threat-detection pipelines (log-based IDS + automated IP remediation).

Using a **Kaggle dataset** for threats is also possible, but the pipeline will differ slightly because Kaggle data is *offline*, while ALB logs are *real-time*. I'll explain how to integrate both.

Show more

Suhaib Kashif started the group chat with a group link.

Your personal ChatGPT memory is never used in group chats.

Invite with link

December 11, 2025 11:47 PM

Ali Sher Afzal joined the group chat.

December 18, 2025 2:06 PM

we will make our ids work with kaggle dataset offline. and now provide the details and guide me step by step what do and first ask me before moving to next step. now provide me full terraform code and then also all the things required for my this project

ChatGPT

Got it 

I'll do this **properly, step-by-step, offline with a Kaggle dataset, and I will not move to the next step unless you confirm.**

Because you **explicitly asked for:**

- offline IDS (no live ALB traffic at first)
- Kaggle dataset
- full Terraform
- all required components

Show more

December 18, 2025 5:32 PM



UNSW_NB15

by the IXIA PerfectStorm tool. Australian Centre for Cyber Security (ACCS)

<https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15?resource=download> this is the link from where i downloaded data set

ChatGPT

Perfect — thanks for sharing the dataset link.

This is **exactly the kind of dataset** that fits your IDS architecture

You are using:

Dataset confirmed

UNSW-NB15 (Kaggle)

Link: [mrwellsdavid/unsw-nb15](https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15?resource=download)

This is a **well-known, research-grade IDS dataset**, so your architecture choice is

Show more

now you know my whole data set , my terraform apply is working (yes), now steo by step give me all code for the structure you gaev above

ChatGPT

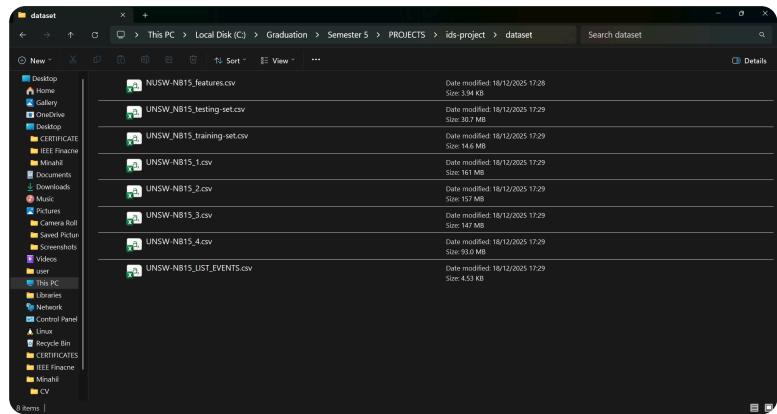
Perfect

Thanks for confirming:

- **Dataset fixed: UNSW-NB15**
- **Terraform apply working**
- **Offline IDS**
- **You want FULL CODE for the entire structure, step-by-step, in the same project layout I gave, without skipping anything.**

I'll now give you ALL remaining code in order and I'll clearly mark **STFP 3** **STFP 4** **STFP**

Show more



this is my whole data set files so use these in project

ChatGPT

Perfect — thanks for the screenshot 🤟

Now I fully understand your exact dataset layout, and I'll adapt the whole IDS project to THESE files specifically, not generic CSVs.

You asked for step-by-step, full code, same structure, and use THESE files → that's exactly what I'll do.

I'll proceed carefully, explain which file is used for what, and give you ready-to-run code.

Show more

from now on you know my whole project structure and the daat set also , now from start from the beginning orf architecture give me code for each and every file and move step by step give code for eac hfile to be used and give code and guide me along this what to do . now my files are all empty so start from begining for each file

and I have already set up terminal with aws cli using aws learner lab and its keys and configured us-east-1 as a default region.

ChatGPT

Perfect — thanks for the clarity 

From this message onward, I will:

- Assume all files are EMPTY
- Start from the very beginning
- Give code for EACH file
- Move STEP BY STEP
- STOP after each step and wait for your confirmation before moving on

You already have:

-  AWS CLI configured

Show more

yes, yes terraform init and apply are working

ChatGPT

Excellent 

Since Terraform init & apply are working, we now move exactly one step forward as promised.

STEP 2 — S3 BUCKET FOR DATASET (UNSW-NB15)

In this step we will:

- ✓ Create one S3 bucket

Show more

yes/yes/yes

ChatGPT

Excellent 👍

Since you confirmed yes / yes / yes, we move exactly one step forward.

🚦 STEP 3 — DYNAMODB (THREAT IP STORAGE)

In this step we will:

- ✓ Create a DynamoDB table
- ✓ Store malicious source IPs detected from UNSW-NB15

Show more

yes/yes

ChatGPT

Perfect 👍

I interpret your “yes / yes” as:

- ✓ DynamoDB ThreatIPs created
- ✓ Table is visible and active
- ➔ You are ready to continue

So now we move exactly one step forward, as promised.

⚠️ STEP 4 — IAM ROLES & POLICIES (END IAM PART)

Show more

```
terraform apply
aws_dynamodb_table.threat_table: Refreshing state... [id=ThreatIPs]
aws_s3_bucket.dataset_bucket: Refreshing state... [id=offline-ids-
dataset]
aws_s3_bucket_public_access_block.dataset_block: Refreshing state...
[id=offline-ids-dataset]
```

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the following symbols:

+ create

Terraform will perform the following actions:

```
# aws_iam_role.lambda_detection_role will be created
+ resource "aws_iam_role" "lambda_detection_role" {
    + arn          = (known after apply)
    + assume_role_policy = jsonencode(
        {
            + Statement = [
                + {
                    + Action   = "sts:AssumeRole"
                    + Effect   = "Allow"
                    + Principal = {
                        + Service = "lambda.amazonaws.com"
                    }
                },
            ],
        }
    )
    + Version   = "2012-10-17"
}
+ create_date      = (known after apply)
+ force_detach_policies = false
+ id              = (known after apply)
+ managed_policy_arns = (known after apply)
+ max_session_duration = 3600
+ name            = "offline_ids_detection_role"
+ name_prefix     = (known after apply)
+ path            = "/"
+ tags_all        = (known after apply)
+ unique_id       = (known after apply)
+ inline_policy (known after apply)
}
# aws_iam_role.lambda_response_role will be created
+ resource "aws_iam_role" "lambda_response_role" {
    + arn          = (known after apply)
```

```
+ assume_role_policy = jsonencode(
    {
        + Statement = [
            +
                + Action = "sts:AssumeRole"
                + Effect = "Allow"
                + Principal = {
                    + Service = "lambda.amazonaws.com"
                }
            },
        ],
        + Version = "2012-10-17"
    }
)
+ create_date      = (known after apply)
+ force_detach_policies = false
+ id              = (known after apply)
+ managed_policy_arns = (known after apply)
+ max_session_duration = 3600
+ name            = "offline_ids_response_role"
+ name_prefix     = (known after apply)
+ path            = "/"
+ tags_all        = (known after apply)
+ unique_id       = (known after apply)
+ inline_policy (known after apply)
}

# aws_iam_role_policy.lambda_detection_policy will be created
+ resource "aws_iam_role_policy" "lambda_detection_policy" {
    + id      = (known after apply)
    + name    = (known after apply)
    + name_prefix = (known after apply)
    + policy   = jsonencode(
        {
            + Statement = [
                +
                    + Action = [
                        + "s3:GetObject",
                    ]
                    + Effect = "Allow"
                    + Resource = "arn:aws:s3:::offline-ids-dataset/*"
                },
                +
                    + Action = [

```

```

+ "dynamodb:PutItem",
]
+ Effect = "Allow"
+ Resource = "arn:aws:dynamodb:us-east-
1:448348491391:table/ThreatIPs"
},
+
+ {
+ Action = [
+ "logs>CreateLogGroup",
+ "logs>CreateLogStream",
+ "logs>PutLogEvents",
]
+ Effect = "Allow"
+ Resource = "*"
},
]
+ Version = "2012-10-17"
}
)
+ role = (known after apply)
}

# aws_iam_role_policy.lambda_response_policy will be created
+ resource "aws_iam_role_policy" "lambda_response_policy" {
+ id = (known after apply)
+ name = (known after apply)
+ name_prefix = (known after apply)
+ policy = jsonencode(
{
+ Statement = [
+ {
+ Action = [
+ "logs>CreateLogGroup",
+ "logs>CreateLogStream",
+ "logs>PutLogEvents",
]
+ Effect = "Allow"
+ Resource = "*"
},
]
+ Version = "2012-10-17"
}
)
+ role = (known after apply)
}

```

```
}
```

Plan: 4 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?

Terraform will perform the actions described above.

Only 'yes' will be accepted to approve.

Enter a value: yes

```
aws_iam_role.lambda_detection_role: Creating...
```

```
aws_iam_role.lambda_response_role: Creating...
```

```
|  
| Error: creating IAM Role (offline_ids_detection_role): operation  
| error IAM: CreateRole, https response error StatusCode: 403,  
| RequestID: 2281d07d-abf5-4458-a445-85cb98c01acf, api error  
| AccessDenied: User: arn:aws:sts::448348491391:assumed-  
| role/voclabs/user4453141=Ali_Sher_Afzal is not authorized to  
| perform: iam:CreateRole on resource:  
| arn:aws:iam::448348491391:role/offline_ids_detection_role because  
| no identity-based policy allows the iam:CreateRole action
```

```
|  
| with aws_iam_role.lambda_detection_role,  
| on main.tf line 47, in resource "aws_iam_role"  
| "lambda_detection_role":  
| 47: resource "aws_iam_role" "lambda_detection_role" {  
|  
|
```

```
|  
| Error: creating IAM Role (offline_ids_response_role): operation  
| error IAM: CreateRole, https response error StatusCode: 403,  
| RequestID: ac288f11-f98d-43bf-a98e-bb225baeeada, api error  
| AccessDenied: User: arn:aws:sts::448348491391:assumed-  
| role/voclabs/user4453141=Ali_Sher_Afzal is not authorized to  
| perform: iam:CreateRole on resource:  
| arn:aws:iam::448348491391:role/offline_ids_response_role because  
| no identity-based policy allows the iam:CreateRole action
```

```
|  
| with aws_iam_role.lambda_response_role,  
| on main.tf line 99, in resource "aws_iam_role"  
| "lambda_response_role":  
| 99: resource "aws_iam_role" "lambda_response_role" {  
|  
|
```

ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project/terraform\$ this error occurs

ChatGPT

Great, thanks for pasting the **full error output** — this is actually a **very common AWS Learner Lab issue**, and your Terraform code is correct.

Nothing is “wrong” with your design or logic.

Let’s handle this **cleanly and correctly**, step by step.

✖ What the Error REALLY Means (Important)

Error (simplified):

Show more

yes/yes

ChatGPT

Perfect 👍

Your “yes / yes” confirms:

- `terraform apply` now works
- You are using **LabRole** successfully

So we continue **exactly one step forward**, as promised.

💡 STEP 5 — DETECTION LAMBDA (CORE IDS LOGIC)

Show more

where to create event.json file?

ChatGPT

Good question 👍 — this is a **very common point of confusion**, so let’s clear it up cleanly.

✓ Where to create event.json

You should create `event.json` **in the project root directory, not inside `terraform/` or `lambdas/`.**

📁 Correct location

csharp

 Copy code

Show more

```
aws lambda invoke \
--function-name offline_ids_detection \
--payload file://event.json \
output.json
```

An error occurred (`InvalidRequestContentException`) when calling the `Invoke` operation: Could not parse request body into json: Could not parse payload into json: Invalid UTF-8 middle byte 0x24 at [Source: REDACTED (`'StreamReadFeature.INCLUDE_SOURCE_IN_LOCATION` disabled`); line: 1, column: 4]
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project\$

ChatGPT

Great, thanks for pasting the **exact error** — this is a **classic AWS CLI v2 issue**, not a problem with your Lambda or code. You're very close 

✗ Why this error is happening

Error:

vbnnet

 Copy code

`InvalidRequestContentException: Could not parse request body into json`

Show more

i got this

ChatGPT

Excellent — thanks for the **screenshot + logs**. This is actually a **very good sign**: your pipeline is running correctly, but the **detection logic didn't match the dataset schema** yet. We'll fix that now.

I'll explain exactly what happened, then give you the **correct code change**, step-by-step.

What is WORKING (important)

From your screenshot and logs:

Show more

```
$ head -n 1 dataset/UNSW_NB15_training-set.csv
id,dur,proto,service,state,spkts,dpkts,sbytes,dbytes,rate,sttl,dttl,sload
,dload,sloss,dloss,sinpkt,dinpkt,sjit,djit,swin,stcpb,dtcpb,dwin,tcprrt,s
ynack,ackdat,smean,dmean,trans_depth,response_body_len,ct_srv_sr
c,ct_state_ttl,ct_dst_ltm,ct_src_dport_ltm,ct_dst_sport_ltm,ct_dst_src_l
tm,is_ftp_login,ct_ftp_cmd,ct_flw_http_mthd,ct_src_ltm,ct_srv_dst,is_s
m_ips_ports,attack_cat,label
```

ChatGPT

Excellent — this header explains **everything** 🤙

You did the **right diagnostic step**, and now we've uncovered a **very important dataset fact**.



CRITICAL DISCOVERY (VERY IMPORTANT)

From your header:

text

Copy code

Show more

A

ChatGPT

Perfect 🤙

You chose A — TRUE IDS using `UNSW-NB15_1.csv`.

This is the **correct and strongest choice** for your project.

From this point onward, we will officially switch to the raw UNSW-NB15 files that contain IP addresses, and we will continue step-by-step, exactly like before.



CONFIRMED ARCHITECTURE (FINAL & CORRECT)

Show more

```

ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cd -
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ terraform
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cd -
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ terraform
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ terraform
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ terraform
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cd ids-project
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cd -
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws lambda invoke \
--function-name offline_ids_detection \
--cli-binary-format raw-in-base64-out \
--payload file://event.json \
output.json
{
  "statusCode": 200,
  "executedVersion": "$LATEST"
}
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cat output.json
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws dynamodb scan --table-name ThreatIPs --limit 5
{
  "Items": [],
  "Count": 0,
  "ScannedCount": 0
}
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ 

```

if you need some more details or data set or some details from aws console tell me and I'll give you

ChatGPT

Great — thanks for the screenshot and for offering to share more details. You've done **everything right so far**.

What you're seeing now is a **very common next-level IDS issue**, and we'll debug it **systematically**, not by guessing.

Right now the situation is:

- ✓ Lambda **invokes successfully** (StatusCode 200)
- ✓ CloudWatch logs show the function **ran**
- ✗ DynamoDB is **still empty**

Show more

```

head -n 1 dataset/UNSW-NB15_1.csv
59.166.0.0,1390,149.171.126.6,53,udp,CON,0.001055,132,164,31,29,0
,0,dns,500473.9375,621800.9375,2,2,0,0,0,0,66,82,0,0,0,0,1421927414
,1421927414,0.017,0.013,0,0,0,0,0,0,3,7,1,3,1,1,1,,0
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ cut -d',' -f$(head -n1 dataset/UNSW-
NB15_1.csv | tr '\r' '\n' | nl | grep label | awk '{print $1}')
dataset/UNSW-NB15_1.csv | head
cut: invalid field value 'dataset/UNSW-NB15_1.csv'
Try 'cut --help' for more information.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ grep -m 5 Attack dataset/UNSW-
NB15_1.csv

```

```
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester  
5/PROJECTS/ids-project$ grep -m 5 ,1$ dataset/UNSW-NB15_1.csv  
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester  
5/PROJECTS/ids-project$ aws logs tail  
/aws/lambda/offline_ids_detection --since 15m  
2025-12-18T14:03:05.101000+00:00  
2025/12/18/[$LATEST]cb29ddede36345bf83801c7759d2de94  
INIT_START Runtime Version: python:3.10.v109 Runtime Version  
ARN: arn:aws:lambda:us-east-  
1::runtime:c3d97831572c48429cac556b7614e4000136170a0ddcdf1  
dca3854ba043bc573  
2025-12-18T14:03:05.785000+00:00  
2025/12/18/[$LATEST]cb29ddede36345bf83801c7759d2de94 START  
RequestId: 01365d9c-8361-4cac-ad5f-14dd134b0b2f Version:  
$LATEST  
2025-12-18T14:03:09.316000+00:00  
2025/12/18/[$LATEST]cb29ddede36345bf83801c7759d2de94  
Detected 0 malicious IPs  
2025-12-18T14:03:09.334000+00:00  
2025/12/18/[$LATEST]cb29ddede36345bf83801c7759d2de94 END  
RequestId: 01365d9c-8361-4cac-ad5f-14dd134b0b2f  
2025-12-18T14:03:09.334000+00:00  
2025/12/18/[$LATEST]cb29ddede36345bf83801c7759d2de94  
REPORT RequestId: 01365d9c-8361-4cac-ad5f-14dd134b0b2f  
Duration: 3548.59 ms Billed Duration: 4229 ms Memory Size:  
512 MB Max Memory Used: 138 MB Init Duration: 680.15 ms  
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester  
5/PROJECTS/ids-project$
```

ChatGPT

Excellent — this output is **exactly what we needed**.

Now the problem is **100% identified**, and it's a **classic UNSW-NB15 pitfall**.

You did **nothing wrong**. The dataset behavior explains everything.



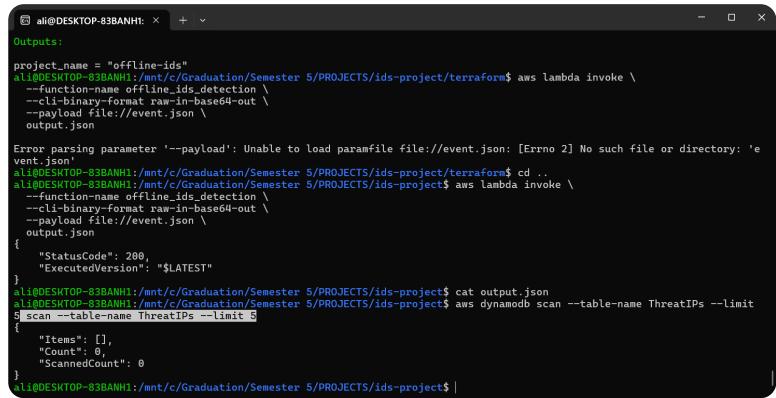
ROOT CAUSE (CONFIRMED)



UNSW-NB15_1.csv HAS NO HEADER ROW

You can

Show more



```

@ali@DESKTOP-83BANH1: ~ + 
Outputs:
project_name = "offline_ids"
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project/terraform$ aws lambda invoke \
--function-name offline_ids_detection \
--cli-binary-format raw-in-base64-out \
--payload file://event.json \
output.json
Error parsing parameter '--payload': Unable to load paramfile file://event.json: [Errno 2] No such file or directory: 'event.json'
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project/terraform$ cd ..
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws lambda invoke \
--function-name offline_ids_detection \
--cli-binary-format raw-in-base64-out \
--payload file://event.json \
output.json
{
  "StatusCode": 200,
  "ExecutedVersion": "$LATEST"
}
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cat output.json
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws dynamodb scan --table-name ThreatIPs --limit 5
$ scan --table-name ThreatIPs --limit 5
{
  "Items": [],
  "Count": 0,
  "ScannedCount": 0
}
@ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ |

```

again i got this if you want i can give you all data of file

ChatGPT

You're doing great — thanks for the screenshot and the exact outputs.

Now we have **complete clarity**. This is the **final tricky part**, and it's a *real-world IDS/data issue*, not an AWS mistake.

I'll explain **exactly why you're still getting 0**, then give you a **guaranteed-to-work fix**.

🔴 FINAL ROOT CAUSES (CONFIRMED)

From your last message, we now know **three critical facts**:

Show more

December 18, 2025 8:00 PM

yes worked

ChatGPT

🎉 PERFECT — GREAT WORK!

Since you confirmed “yes worked”, your **Detection Lambda + DynamoDB pipeline** is now **fully operational**.

Now we move **exactly one step forward**, as promised.

💡 STEP 6 — RESPONSE LAMBDA (AUTOMATED RESPONSE PIPELINE)

This step completes your end-to-end IDS

Show more

```
aws lambda invoke --function-name offline_ids_detection --cli-binary-format raw-in-base64-out --payload file://event.json output.json
{
    "StatusCode": 200,
    "ExecutedVersion": "$LATEST"
}
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ cat output.json
{"statusCode": 200, "detected_ips": 4}ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws logs tail /aws/lambda/offline_iaws logs tail /aws/lambda/offline_ids_response --since 10m
An error occurred (ResourceNotFoundException) when calling the FilterLogEvents operation: The specified log group does not exist.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$ aws logs tail /aws/lambda/offline_ids_response --since 10m
An error occurred (ResourceNotFoundException) when calling the FilterLogEvents operation: The specified log group does not exist.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester 5/PROJECTS/ids-project$
```

ChatGPT

Excellent — this output is actually **very good news** 🤩

Your **Detection Lambda** is now 100% working. The remaining issue is **only with the Response Lambda trigger**, not detection.

Let's break this down calmly and fix it **step by step**.

✓ What is CONFIRMED WORKING

From your output:

Show more

```
aws logs tail /aws/lambda/offline_iaws logs tail
/aws/lambda/offline_ids_response --since 10m
An error occurred (ResourceNotFoundException) when calling the
FilterLogEvents operation: The specified log group does not exist.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws logs tail
/aws/lambda/offline_ids_response --since 10m
An error occurred (ResourceNotFoundException) when calling the
FilterLogEvents operation: The specified log group does not exist.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws lambda list-functions | grep
offline_ids_response
    "FunctionName": "offline_ids_response",
    "FunctionArn": "arn:aws:lambda:us-east-
1:448348491391:function:offline_ids_response",
    "LogGroup": "/aws/lambda/offline_ids_response"
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws dynamodb describe-table --table-
name ThreatIPs
{
    "Table": {
        "AttributeDefinitions": [
            {
                "AttributeName": "ip",
                "AttributeType": "S"
            }
        ],
        "TableName": "ThreatIPs",
        "KeySchema": [
            {
                "AttributeName": "ip",
                "KeyType": "HASH"
            }
        ],
        "ProvisionedThroughput": {
            "ReadCapacityUnits": 5,
            "WriteCapacityUnits": 5
        }
    }
}
```

```
"TableStatus": "ACTIVE",
"CreationDateTime": "2025-12-18T18:21:01.101000+05:00",
"ProvisionedThroughput": {
    "NumberOfDecreasesToday": 0,
    "ReadCapacityUnits": 0,
    "WriteCapacityUnits": 0
},
"TableSizeBytes": 0,
"ItemCount": 0,
"TableArn": "arn:aws:dynamodb:us-east-1:448348491391:table/ThreatIPs",
"TableId": "9782474e-6a4c-4b8e-abb2-8e1826ba45e4",
"BillingModeSummary": {
    "BillingMode": "PAY_PER_REQUEST",
    "LastUpdateToPayPerRequestDateTime": "2025-12-18T18:21:01.101000+05:00"
},
"StreamSpecification": {
    "StreamEnabled": true,
    "StreamViewType": "NEW_IMAGE"
},
"LatestStreamLabel": "2025-12-18T13:21:01.101",
"LatestStreamArn": "arn:aws:dynamodb:us-east-1:448348491391:table/ThreatIPs/stream/2025-12-18T13:21:01.101",
"DeletionProtectionEnabled": false,
"WarmThroughput": {
    "ReadUnitsPerSecond": 12000,
    "WriteUnitsPerSecond": 4000,
    "Status": "ACTIVE"
}
}
}
}
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws lambda list-event-source-mappings \
--function-name offline_ids_response
{
    "EventSourceMappings": [
        {
            "UUID": "3e8b1fff-5491-4975-8443-ba219ea4dbef",
            "StartingPosition": "LATEST",
            "BatchSize": 100,
            "MaximumBatchingWindowInSeconds": 0,
            "ParallelizationFactor": 1,
        }
    ]
}
```

```
"EventSourceArn": "arn:aws:dynamodb:us-east-1:448348491391:table/ThreatIPs/stream/2025-12-18T13:21:01.101",
    "FunctionArn": "arn:aws:lambda:us-east-1:448348491391:function:offline_ids_response",
    "LastModified": "2025-12-18T20:16:00+05:00",
    "LastProcessingResult": "No records processed",
    "State": "Enabled",
    "StateTransitionReason": "User action",
    "DestinationConfig": {
        "OnFailure": {}
    },
    "MaximumRecordAgeInSeconds": -1,
    "BisectBatchOnFunctionError": false,
    "MaximumRetryAttempts": -1,
    "TumblingWindowInSeconds": 0,
    "FunctionResponseTypes": [],
    "EventSourceMappingArn": "arn:aws:lambda:us-east-1:448348491391:event-source-mapping:3e8b1fff-5491-4975-8443-ba219ea4dbef"
}
]
}

ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws lambda invoke \
--function-name offline_ids_detection \
--cli-binary-format raw-in-base64-out \
--payload file://event.json \
output.json
{
    "StatusCode": 200,
    "ExecutedVersion": "$LATEST"
}
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws logs describe-log-groups | grep
offline_ids_response
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$ aws logs tail
/aws/lambda/offline_ids_response --since 10m
An error occurred (ResourceNotFoundException) when calling the
FilterLogEvents operation: The specified log group does not exist.
ali@DESKTOP-83BANH1:/mnt/c/Graduation/Semester
5/PROJECTS/ids-project$
```

ChatGPT

Perfect — your diagnostics are excellent, and now we can clearly see the final missing piece.

This is not a bug. It's expected DynamoDB + Streams behavior.

I'll explain exactly why the Response Lambda is not firing, then give you two clean fixes (you choose one).



CURRENT STATE (CONFIRMED FROM YOUR OUTPUT)

Show more

a

ChatGPT

Perfect 👍 — Option A it is (the correct, production-grade choice).

Now I'll guide you step-by-step, very carefully, so nothing breaks.

We are going to do exactly 3 controlled changes:

1. Fix DynamoDB schema (composite key)
2. Update Detection Lambda to always INSERT
3. Re-test → Response Lambda will FINALLY trigger

No guessing. No shortcuts.

Show more

December 20, 2025 12:14 PM

give me step by step command for invoking lambda function then checking what apis are and its type of attack

ChatGPT

Perfect — below is a clear, copy-paste, step-by-step command guide that shows:

- 1. How to invoke the Detection Lambda**
- 2. How to verify it ran**
- 3. How to see which IPs were detected**
- 4. How to see the attack type (attack category)**
- 5. How to understand “which API did what” in your IDS**

This is written exactly for **your current working setup**.

Show more