Deliverables (Students must submit)
Deliverable A — “Audit Evidence Pack” (one folder)

学生は audit-pack/ フォルダを提出。

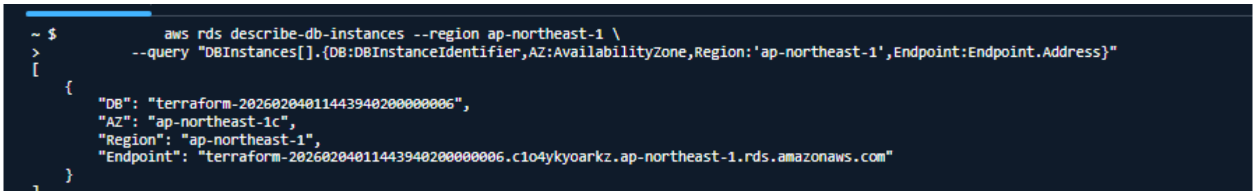
```
audit-pack/
├── 00_architecture-summary.md
├── 01_data-residency-proof.txt
├── 02_edge-proof-cloudfront.txt
├── 03_waf-proof.txt
├── 04_cloudtrail-change-proof.txt
├── 05_network-corridor-proof.txt
└── evidence.json   (Malgus scripts output)
```

Deliverable B — One paragraph “auditor narrative”
“この設計が APPI 的に安全で、なぜ DB を海外に置けないか”を 8〜12 行で説明。

Verification Commands (CLI proof students can paste)
1) Data residency proof (RDS only in Tokyo)

Tokyo: RDS exists

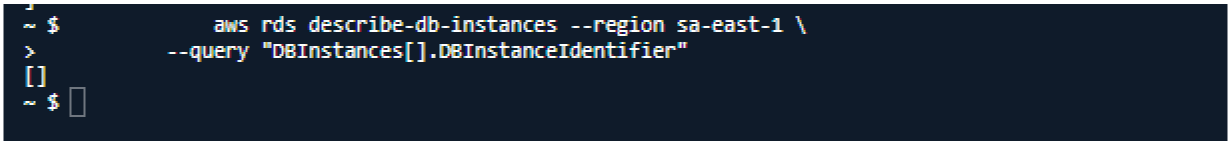
```
    aws rds describe-db-instances --region ap-northeast-1 \
  --query
"DBInstances[].{DB:DBInstanceIdentifier,AZ:AvailabilityZone,Region:'ap-northeast-1',Endpoint:Endpoint.Address}"
```



```
~ $           aws rds describe-db-instances --region ap-northeast-1 \
>         --query "DBInstances[].{DB:DBInstanceIdentifier,AZ:AvailabilityZone,Region:'ap-northeast-1',Endpoint:Endpoint.Address}"
[
    {
        "DB": "terraform-20260204011443940200000006",
        "AZ": "ap-northeast-1c",
        "Region": "ap-northeast-1",
        "Endpoint": "terraform-20260204011443940200000006.c1o4ykyoarkz.ap-northeast-1.rds.amazonaws.com"
    }
```

São Paulo: No RDS

```
    aws rds describe-db-instances --region sa-east-1 \
  --query "DBInstances[].DBInstanceIdentifier"
```

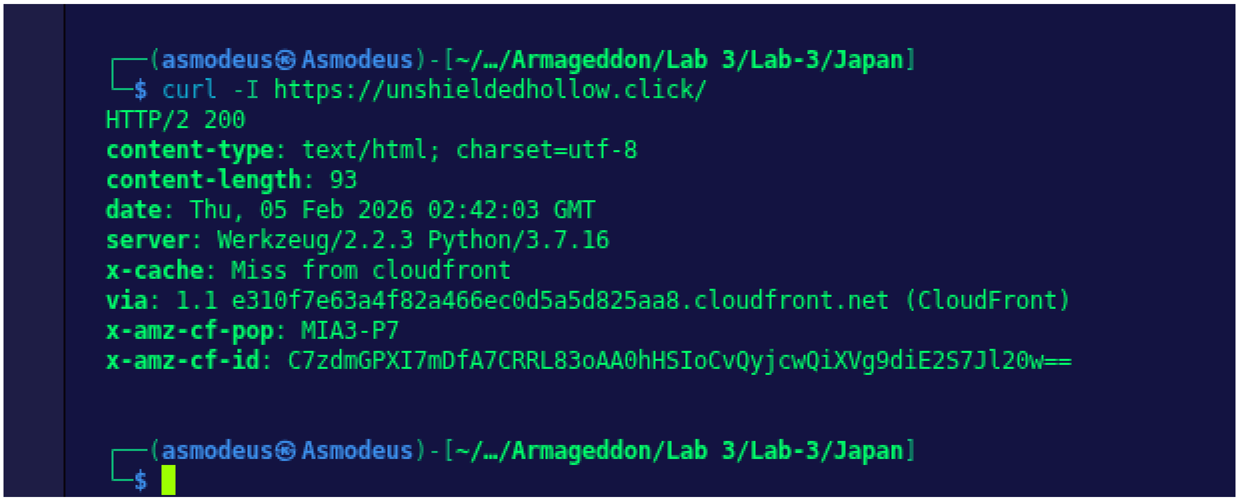


```
~ $           aws rds describe-db-instances --region sa-east-1 \
>         --query "DBInstances[].DBInstanceIdentifier"
[]
~ $ 
```

2) Edge proof (CloudFront logs show cache + access)

Students capture request headers:

curl -I https://chewbacca-growls.com/api/public-feed



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Japan]
└─$ curl -I https://unshieldedhollow.click/
HTTP/2 200
content-type: text/html; charset=utf-8
content-length: 93
date: Thu, 05 Feb 2026 02:42:03 GMT
server: Werkzeug/2.2.3 Python/3.7.16
x-cache: Miss from cloudfront
via: 1.1 e310f7e63a4f82a466ec0d5a5d825aa8.cloudfront.net (CloudFront)
x-amz-cf-pop: MIA3-P7
x-amz-cf-id: C7zdmGPXI7mDfA7CRRL83oAA0hHSIoCvQyjcwQiXVg9diE2S7Jl20w==


┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Japan]
└─$ 
```

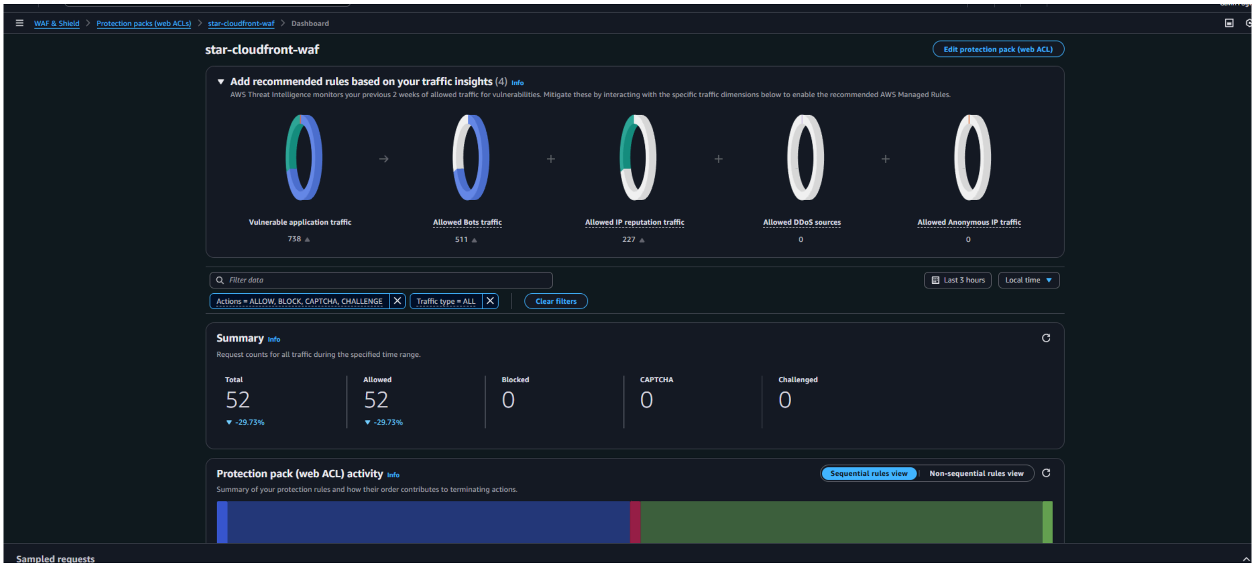
And/or submit CloudFront standard log evidence (Hit/Miss/RefreshHit)

3) WAF proof

Provide:

WAF log snippet or Insights summary

WAF logging destination options are documented



☰ WAF & Shield > Protection packs (web ACLs) > star-cloudfront-waf > Dashboard
Edit protection pack (web ACL)
Info
Filter data
Last 3 hours
Local time ▼
Clear filters
Sequential rules view
Non-sequential rules view
Sampled requests

star-cloudfront-waf

▼ Add recommended rules based on your traffic insights (4)

AWS Threat Intelligence monitors your previous 2 weeks of allowed traffic for vulnerabilities. Mitigate these by interacting with the specific traffic dimensions below to enable the recommended AWS Managed Rules.

Vulnerable application traffic	Allowed Bots traffic	Allowed IP reputation traffic	Allowed DDoS sources	Allowed Anonymous IP traffic
738 ▲	511 ▲	227 ▲	0	0

Actions = ALLOW, BLOCK, CAPTCHA, CHALLENGE ✕

Traffic type = ALL ✕

Summary

Request counts for all traffic during the specified time range.

Total	Allowed	Blocked	CAPTCHA	Challenged
52 ▼ -29.73%	52 ▼ -29.73%	0	0	0

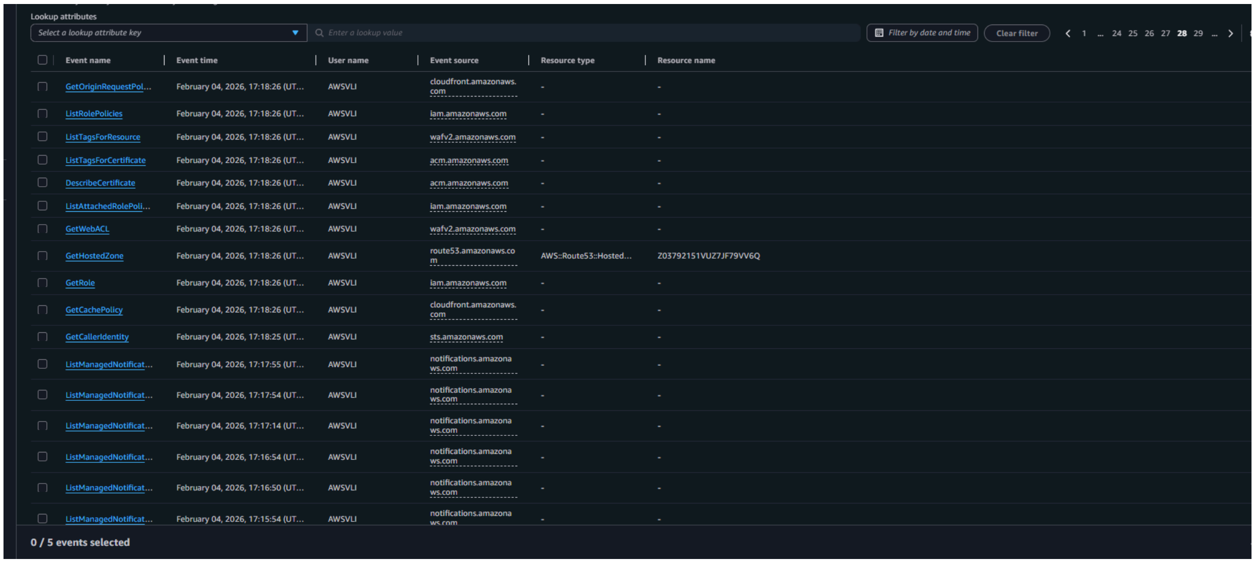
Protection pack (web ACL) activity

Summary of your protection rules and how their order contributes to terminating actions.

4) Change proof (CloudTrail)
CloudTrail has event history with a 90-day immutable record of management events

Students capture:

--> “who changed SG / TGW route / WAF / CloudFront config”



Lookup attributes

☐	Event name	Event time	User name	Event source	Resource type	Resource name
☐	GetOriginRequestPol…	February 04, 2026, 17:18:26 (UT…	AWSVLI	cloudfront.amazonaws.com	-	-
☐	ListRolePolicies	February 04, 2026, 17:18:26 (UT…	AWSVLI	iam.amazonaws.com	-	-
☐	ListTagsForResource	February 04, 2026, 17:18:26 (UT…	AWSVLI	wafv2.amazonaws.com	-	-
☐	ListTagsForCertificate	February 04, 2026, 17:18:26 (UT…	AWSVLI	acm.amazonaws.com	-	-
☐	DescribeCertificate	February 04, 2026, 17:18:26 (UT…	AWSVLI	acm.amazonaws.com	-	-
☐	ListAttachedRolePoli…	February 04, 2026, 17:18:26 (UT…	AWSVLI	iam.amazonaws.com	-	-
☐	GetWebACL	February 04, 2026, 17:18:26 (UT…	AWSVLI	wafv2.amazonaws.com	-	-
☐	GetHostedZone	February 04, 2026, 17:18:26 (UT…	AWSVLI	route53.amazonaws.com	AWS::Route53::Hosted…	Z03792151VUZ7JF79VV6Q
☐	GetRole	February 04, 2026, 17:18:26 (UT…	AWSVLI	iam.amazonaws.com	-	-
☐	GetCachePolicy	February 04, 2026, 17:18:26 (UT…	AWSVLI	cloudfront.amazonaws.com	-	-
☐	GetCallerIdentity	February 04, 2026, 17:18:25 (UT…	AWSVLI	sts.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:17:55 (UT…	AWSVLI	notifications.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:17:54 (UT…	AWSVLI	notifications.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:17:14 (UT…	AWSVLI	notifications.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:16:54 (UT…	AWSVLI	notifications.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:16:50 (UT…	AWSVLI	notifications.amazonaws.com	-	-
☐	ListManagedNotificat…	February 04, 2026, 17:15:54 (UT…	AWSVLI	notifications.amazonaws.com	-	-

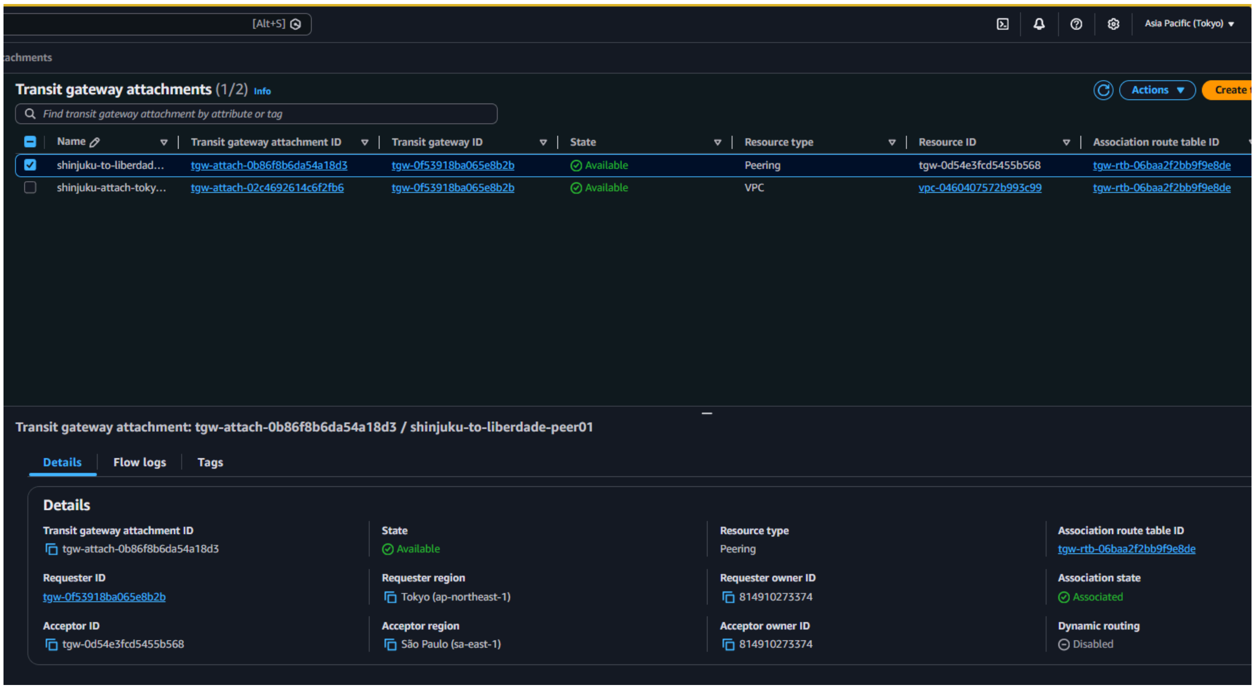
0 / 5 events selected

5) Network corridor proof (TGW)
Students prove:

TGW attachments exist in both regions

routes point cross-region CIDRs to TGW

Tokyo



Transit gateway attachments (1/2) Info

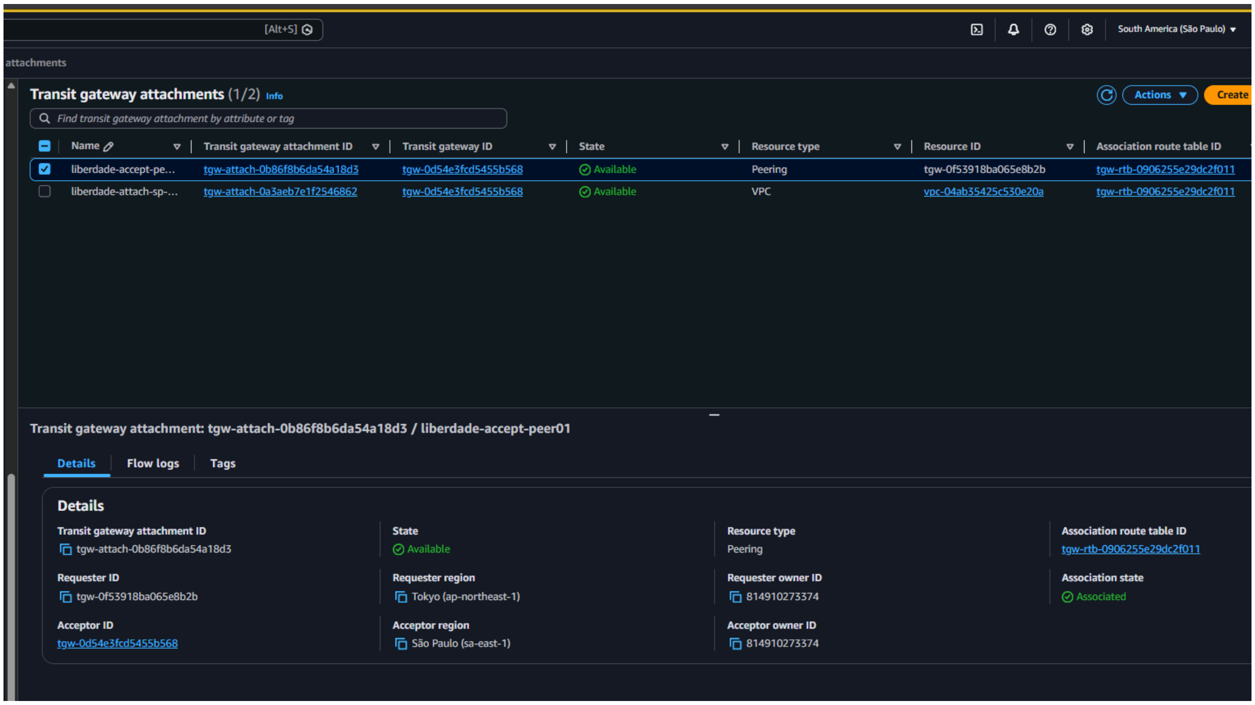
☐	Name	Transit gateway attachment ID	Transit gateway ID	State	Resource type	Resource ID	Association route table ID
☑	shinjuku-to-liberdad…	tgw-attach-0b86f8b6da54a18d3	tgw-0f53918ba065e8b2b	Available	Peering	tgw-0d54e3fcd5455b568	tgw-rtb-06baa2f2bb9f9e8de
☐	shinjuku-attach-toky…	tgw-attach-02c4692614c6f2fb6	tgw-0f53918ba065e8b2b	Available	VPC	vpc-0460407572b993c99	tgw-rtb-06baa2f2bb9f9e8de

Transit gateway attachment: tgw-attach-0b86f8b6da54a18d3 / shinjuku-to-liberdade-peer01

Details

Transit gateway attachment ID tgw-attach-0b86f8b6da54a18d3	State Available	Resource type Peering	Association route table ID tgw-rtb-06baa2f2bb9f9e8de
Requester ID tgw-0f53918ba065e8b2b	Requester region Tokyo (ap-northeast-1)	Requester owner ID 814910273374	Association state Associated
Acceptor ID tgw-0d54e3fcd5455b568	Acceptor region São Paulo (sa-east-1)	Acceptor owner ID 814910273374	Dynamic routing Disabled

Sao-Paulo



Transit gateway attachments (1/2) Info

Find transit gateway attachment by attribute or tag

	Name	Transit gateway attachment ID	Transit gateway ID	State	Resource type	Resource ID	Association route table ID
☑	liberdade-accept-pe...	tgw-attach-0b86f8b6da54a18d3	tgw-0d54e3fcd5455b568	Available	Peering	tgw-0f53918ba065e8b2b	tgw-rtb-0906255e29dc2f011
□	liberdade-attach-sp-...	tgw-attach-0a3aeb7e1f2546862	tgw-0d54e3fcd5455b568	Available	VPC	vpc-04ab35425c530e20a	tgw-rtb-0906255e29dc2f011

Transit gateway attachment: tgw-attach-0b86f8b6da54a18d3 / liberdade-accept-peer01

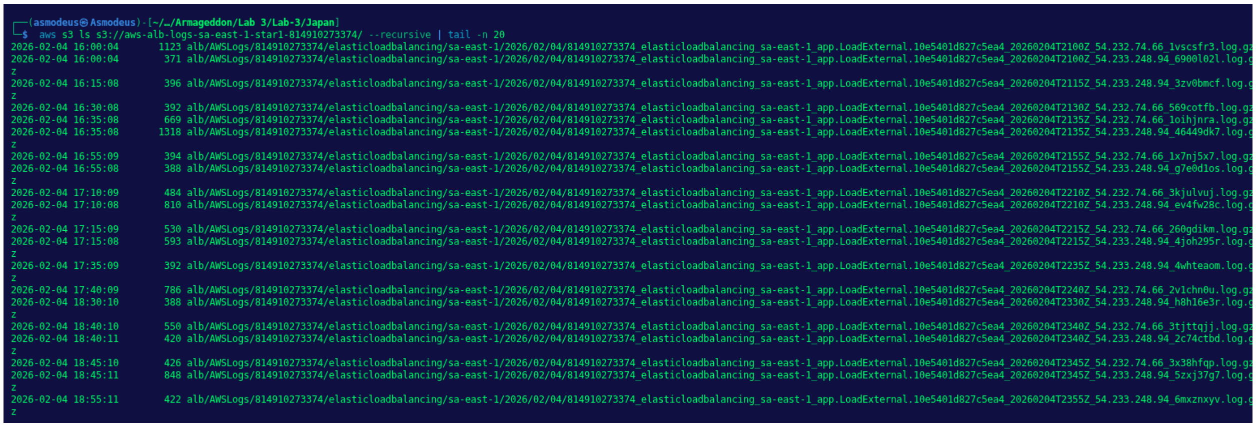
Details | Flow logs | Tags

Details

Transit gateway attachment ID tgw-attach-0b86f8b6da54a18d3	State Available	Resource type Peering	Association route table ID tgw-rtb-0906255e29dc2f011
Requester ID tgw-0f53918ba065e8b2b	Requester region Tokyo (ap-northeast-1)	Requester owner ID 814910273374	Association state Associated
Acceptor ID tgw-0d54e3fcd5455b568	Acceptor region São Paulo (sa-east-1)	Acceptor owner ID 814910273374	

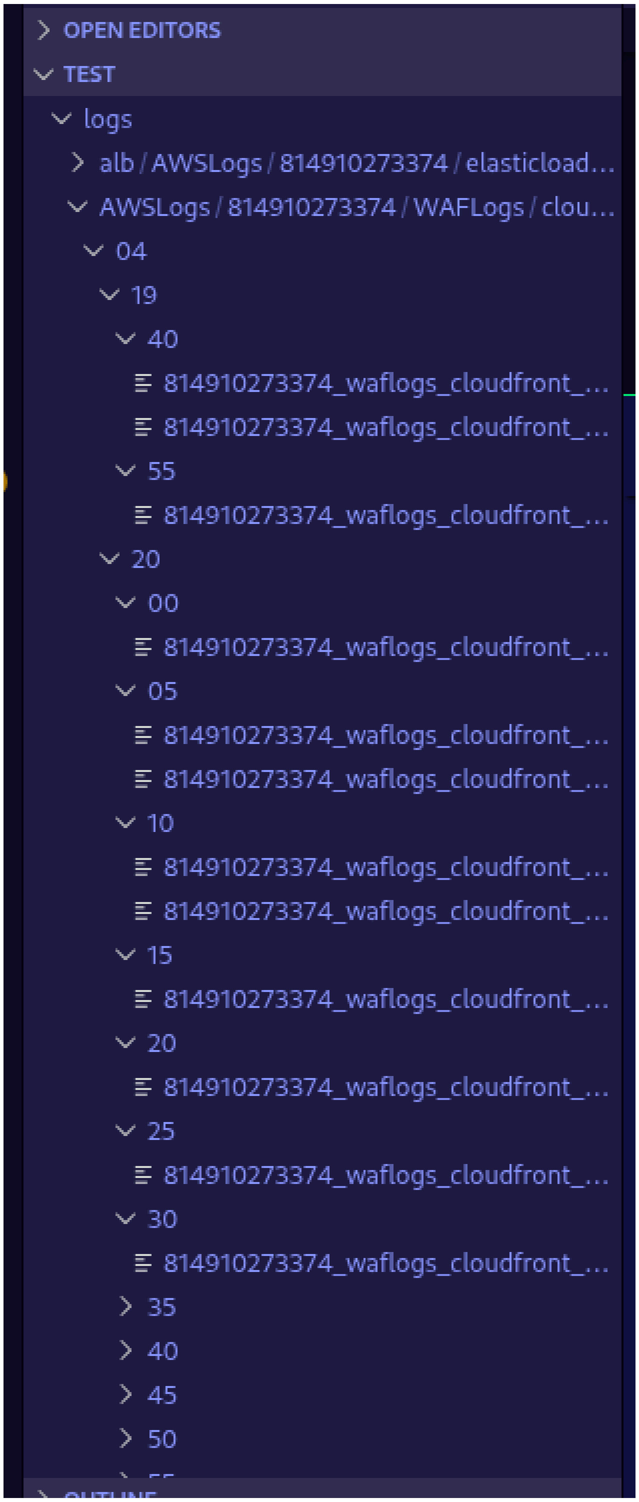
6) AWS CLI verification (students can prove the bucket/logs exist)

```
aws s3 ls s3://Class_Lab3/
# If logs are under a folder/prefix:
aws s3 ls s3://<name> / --recursive | tail -n 20
```



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Japan]
└─$  aws s3 ls s3://aws-alb-logs-sa-east-1-star1-814910273374/ --recursive | tail -n 20
2026-02-04 16:00:04       1123 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2100Z_54.232.74.66_1vscsfr3.log.gz
2026-02-04 16:00:04        371 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2100Z_54.233.248.94_6900l02l.log.g
z
2026-02-04 16:15:08        396 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2115Z_54.233.248.94_3zv0bmcf.log.g
z
2026-02-04 16:30:08        392 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2130Z_54.232.74.66_569cotfb.log.gz
2026-02-04 16:35:08        669 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2135Z_54.232.74.66_1oihjnra.log.gz
2026-02-04 16:35:08       1318 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2135Z_54.233.248.94_46449dk7.log.g
z
2026-02-04 16:55:09        394 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2155Z_54.232.74.66_1x7nj5x7.log.gz
2026-02-04 16:55:08        388 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2155Z_54.233.248.94_g7e0d1os.log.g
z
2026-02-04 17:10:09        484 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2210Z_54.232.74.66_3kjulvuj.log.gz
2026-02-04 17:10:08        810 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2210Z_54.233.248.94_ev4fw28c.log.g
z
2026-02-04 17:15:09        530 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2215Z_54.232.74.66_260gdikm.log.gz
2026-02-04 17:15:08        593 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2215Z_54.233.248.94_4joh295r.log.g
z
2026-02-04 17:35:09        392 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2235Z_54.233.248.94_4whteaom.log.g
z
2026-02-04 17:40:09        786 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2240Z_54.232.74.66_2v1chn0u.log.gz
2026-02-04 18:30:10        388 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2330Z_54.233.248.94_h8h16e3r.log.g
z
2026-02-04 18:40:10        550 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2340Z_54.232.74.66_3tjttqjj.log.gz
2026-02-04 18:40:11        420 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2340Z_54.233.248.94_2c74ctbd.log.g
z
2026-02-04 18:45:10        426 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2345Z_54.232.74.66_3x38hfqp.log.gz
2026-02-04 18:45:11        848 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2345Z_54.233.248.94_5zxj37g7.log.g
z
2026-02-04 18:55:11        422 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374_elasticloadbalancing_sa-east-1_app.LoadExternal.10e5401d827c5ea4_20260204T2355Z_54.233.248.94_6mxznxyv.log.g
z
```

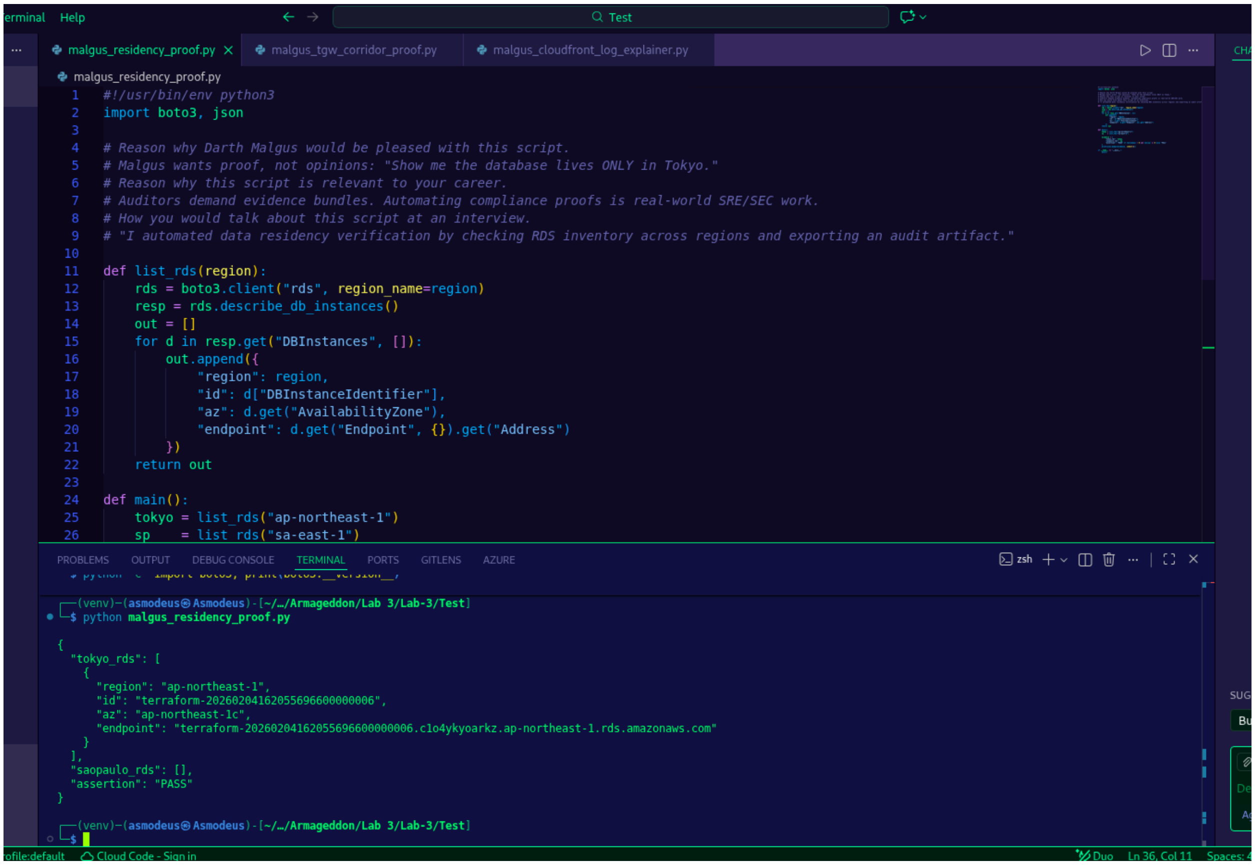
Download one file manually (sanity check):



```
> OPEN EDITORS
v TEST
  v logs
    > alb / AWSLogs / 814910273374 / elasticload…
    v AWSLogs / 814910273374 / WAFLogs / clou…
      v 04
        v 19
          v 40
              814910273374_waflogs_cloudfront_…
              814910273374_waflogs_cloudfront_…
          v 55
              814910273374_waflogs_cloudfront_…
        v 20
          v 00
              814910273374_waflogs_cloudfront_…
          v 05
              814910273374_waflogs_cloudfront_…
              814910273374_waflogs_cloudfront_…
          v 10
              814910273374_waflogs_cloudfront_…
              814910273374_waflogs_cloudfront_…
          v 15
              814910273374_waflogs_cloudfront_…
          v 20
              814910273374_waflogs_cloudfront_…
          v 25
              814910273374_waflogs_cloudfront_…
          v 30
              814910273374_waflogs_cloudfront_…
          > 35
          > 40
          > 45
          > 50
          > 55
> OUTLINE
```

```
aws s3 cp s3://<name> logs.gz .
```

Script 1 — malgus_residency_proof.py
Creates a “DB only in Tokyo” proof file.

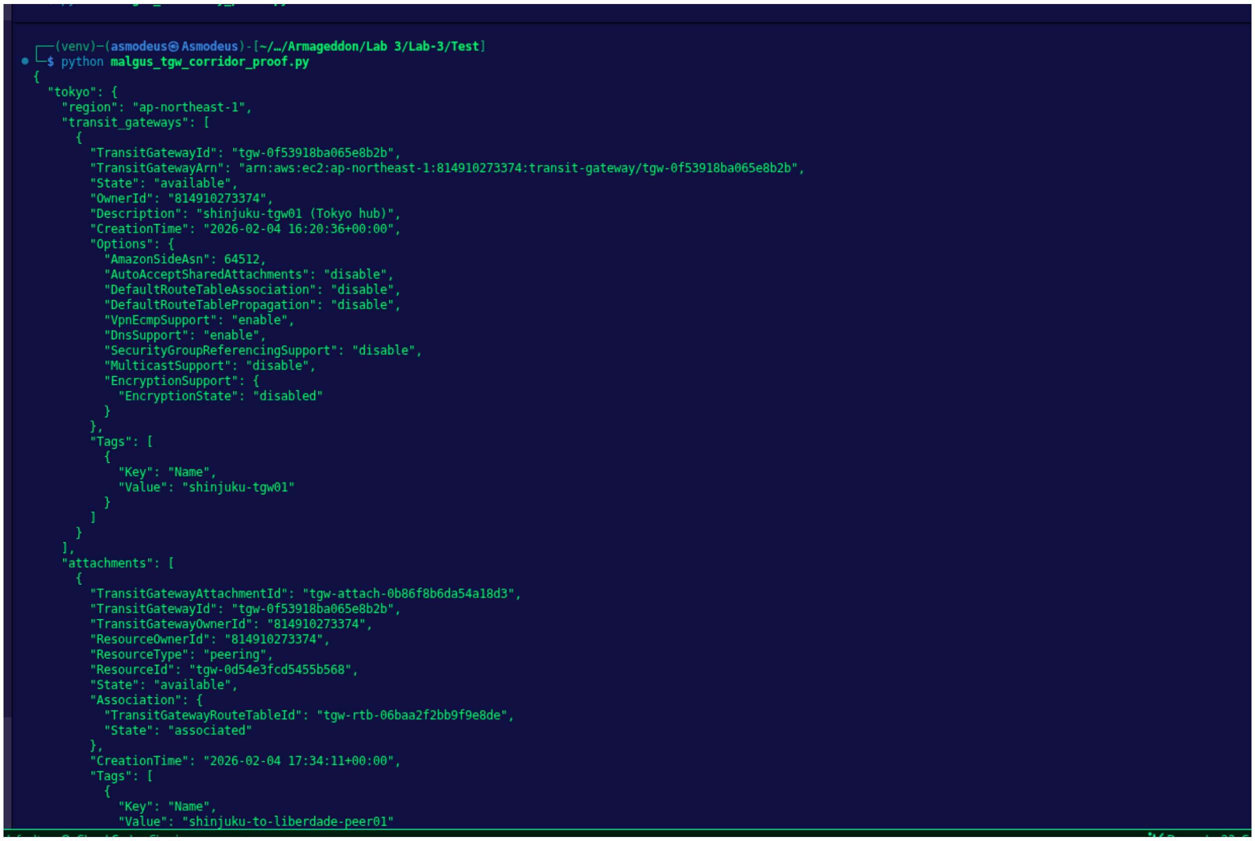


malgus_residency_proof.py

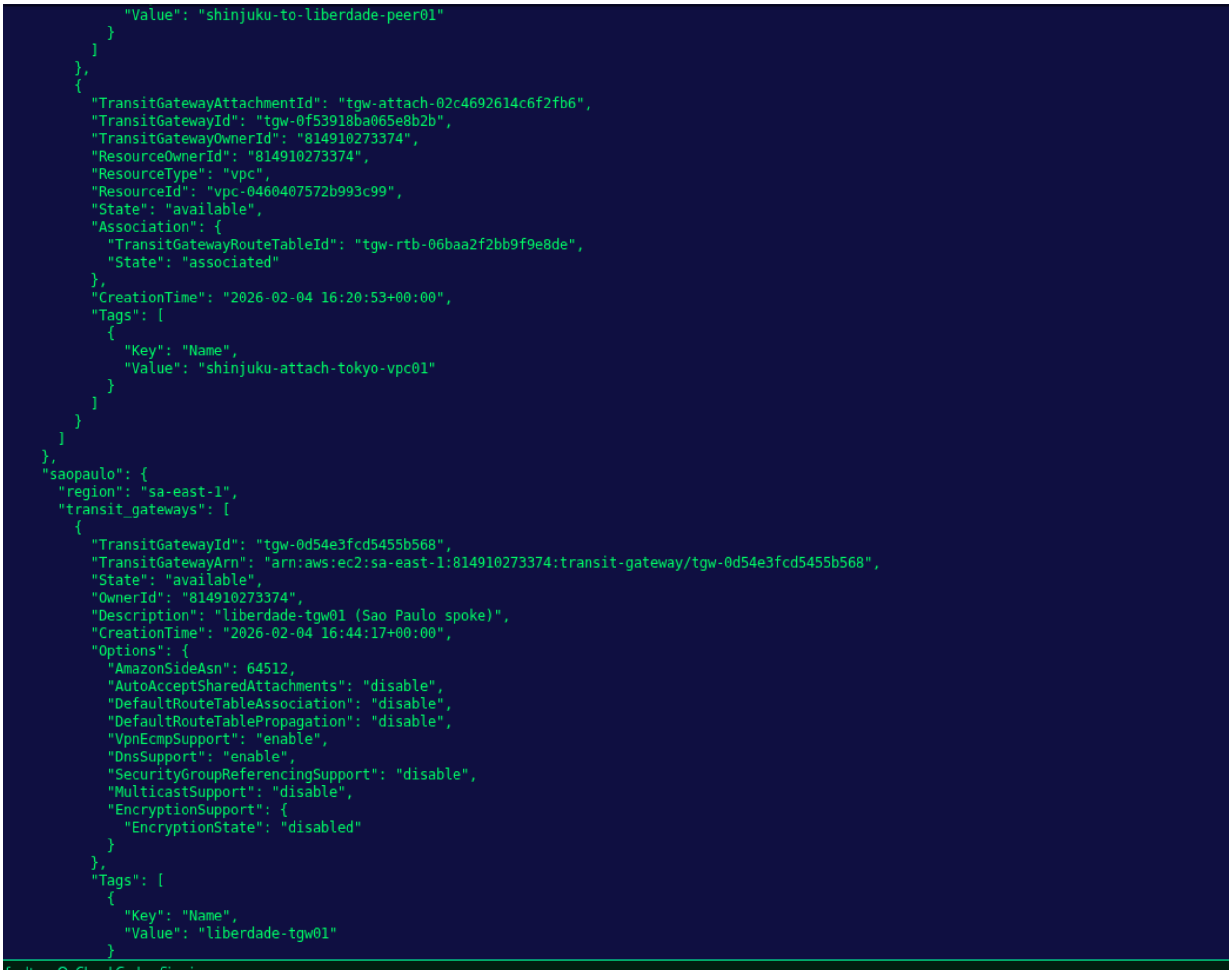
```python
#!/usr/bin/env python3
import boto3, json

# Reason why Darth Malgus would be pleased with this script.
# Malgus wants proof, not opinions: "Show me the database lives ONLY in Tokyo."
# Reason why this script is relevant to your career.
# Auditors demand evidence bundles. Automating compliance proofs is real-world SRE/SEC work.
# How you would talk about this script at an interview.
# "I automated data residency verification by checking RDS inventory across regions and exporting an audit artifact."

def list_rds(region):
    rds = boto3.client("rds", region_name=region)
    resp = rds.describe_db_instances()
    out = []
    for d in resp.get("DBInstances", []):
        out.append({
            "region": region,
            "id": d["DBInstanceIdentifier"],
            "az": d.get("AvailabilityZone"),
            "endpoint": d.get("Endpoint", {}).get("Address")
        })
    return out

def main():
    tokyo = list_rds("ap-northeast-1")
    sp    = list_rds("sa-east-1")
```

```
┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_residency_proof.py

{
  "tokyo_rds": [
    {
      "region": "ap-northeast-1",
      "id": "terraform-20260204162055696600000006",
      "az": "ap-northeast-1c",
      "endpoint": "terraform-20260204162055696600000006.clo4ykyoarkz.ap-northeast-1.rds.amazonaws.com"
    }
  ],
  "saopaulo_rds": [],
  "assertion": "PASS"
}

┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$
```

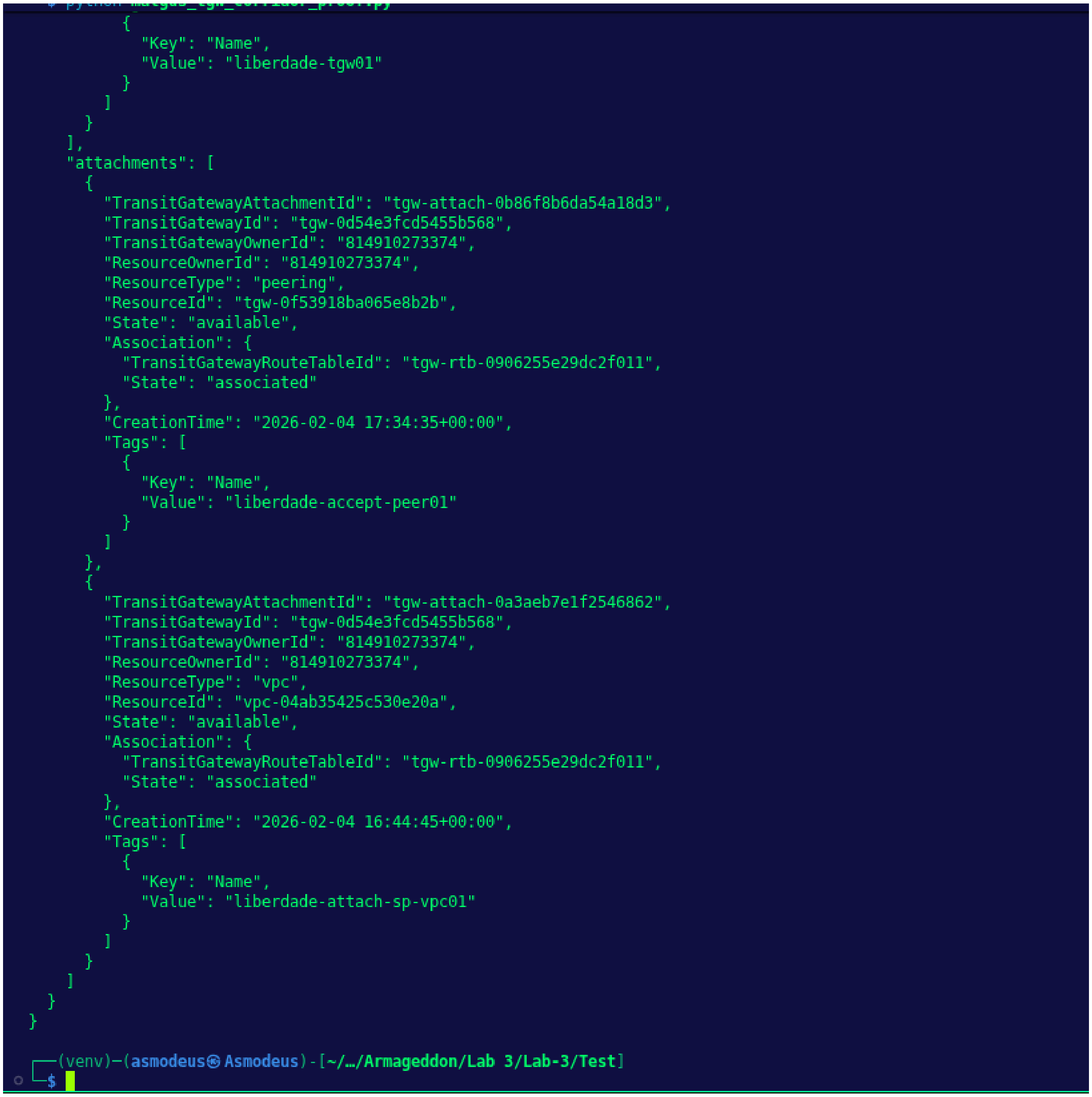
Script 2 — malgus_tgw_corridor_proof.py



```
┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_tgw_corridor_proof.py
{
  "tokyo": {
    "region": "ap-northeast-1",
    "transit_gateways": [
      {
        "TransitGatewayId": "tgw-0f53918ba065e8b2b",
        "TransitGatewayArn": "arn:aws:ec2:ap-northeast-1:814910273374:transit-gateway/tgw-0f53918ba065e8b2b",
        "State": "available",
        "OwnerId": "814910273374",
        "Description": "shinjuku-tgw01 (Tokyo hub)",
        "CreationTime": "2026-02-04 16:20:36+00:00",
        "Options": {
          "AmazonSideAsn": 64512,
          "AutoAcceptSharedAttachments": "disable",
          "DefaultRouteTableAssociation": "disable",
          "DefaultRouteTablePropagation": "disable",
          "VpnEcmpSupport": "enable",
          "DnsSupport": "enable",
          "SecurityGroupReferencingSupport": "disable",
          "MulticastSupport": "disable",
          "EncryptionSupport": {
            "EncryptionState": "disabled"
          }
        },
        "Tags": [
          {
            "Key": "Name",
            "Value": "shinjuku-tgw01"
          }
        ]
      }
    ],
    "attachments": [
      {
        "TransitGatewayAttachmentId": "tgw-attach-0b86f8b6da54a18d3",
        "TransitGatewayId": "tgw-0f53918ba065e8b2b",
        "TransitGatewayOwnerId": "814910273374",
        "ResourceOwnerId": "814910273374",
        "ResourceType": "peering",
        "ResourceId": "tgw-0d54e3fcd5455b568",
        "State": "available",
        "Association": {
          "TransitGatewayRouteTableId": "tgw-rtb-06baa2f2bb9f9e8de",
          "State": "associated"
        },
        "CreationTime": "2026-02-04 17:34:11+00:00",
        "Tags": [
          {
            "Key": "Name",
            "Value": "shinjuku-to-liberdade-peer01"
```



```
              "Value": "shinjuku-to-liberdade-peer01"
            }
          ]
        },
        {
          "TransitGatewayAttachmentId": "tgw-attach-02c4692614c6f2fb6",
          "TransitGatewayId": "tgw-0f53918ba065e8b2b",
          "TransitGatewayOwnerId": "814910273374",
          "ResourceOwnerId": "814910273374",
          "ResourceType": "vpc",
          "ResourceId": "vpc-0460407572b993c99",
          "State": "available",
          "Association": {
            "TransitGatewayRouteTableId": "tgw-rtb-06baa2f2bb9f9e8de",
            "State": "associated"
          },
          "CreationTime": "2026-02-04 16:20:53+00:00",
          "Tags": [
            {
              "Key": "Name",
              "Value": "shinjuku-attach-tokyo-vpc01"
            }
          ]
        }
      ]
    },
    "saopaulo": {
      "region": "sa-east-1",
      "transit_gateways": [
        {
          "TransitGatewayId": "tgw-0d54e3fcd5455b568",
          "TransitGatewayArn": "arn:aws:ec2:sa-east-1:814910273374:transit-gateway/tgw-0d54e3fcd5455b568",
          "State": "available",
          "OwnerId": "814910273374",
          "Description": "liberdade-tgw01 (Sao Paulo spoke)",
          "CreationTime": "2026-02-04 16:44:17+00:00",
          "Options": {
            "AmazonSideAsn": 64512,
            "AutoAcceptSharedAttachments": "disable",
            "DefaultRouteTableAssociation": "disable",
            "DefaultRouteTablePropagation": "disable",
            "VpnEcmpSupport": "enable",
            "DnsSupport": "enable",
            "SecurityGroupReferencingSupport": "disable",
            "MulticastSupport": "disable",
            "EncryptionSupport": {
              "EncryptionState": "disabled"
            }
          },
          "Tags": [
            {
              "Key": "Name",
              "Value": "liberdade-tgw01"
            }
```



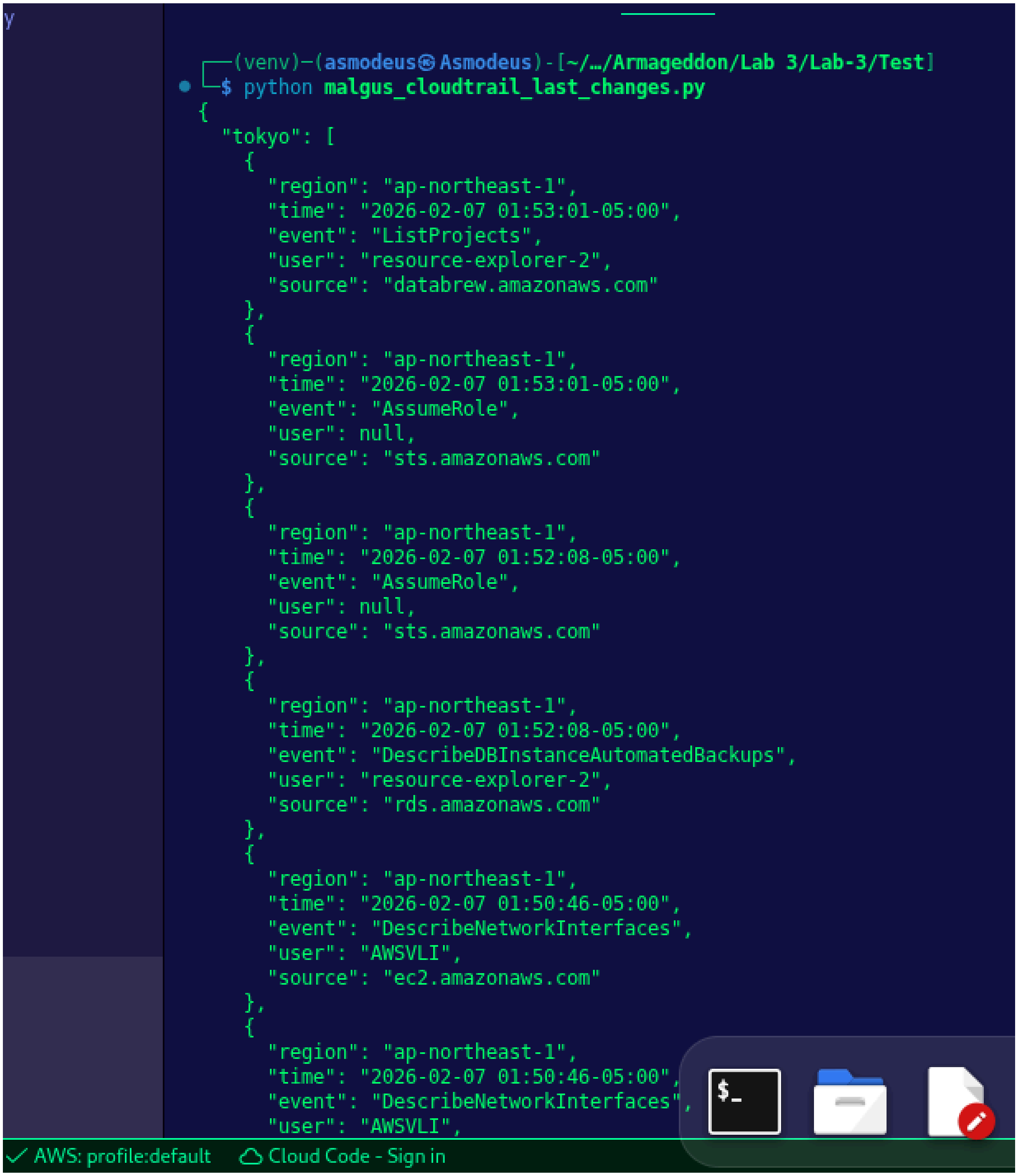
```
  $ python malgus_tgw_corridor_proof.py
          {
            "Key": "Name",
            "Value": "liberdade-tgw01"
          }
        ]
      }
    ],
    "attachments": [
      {
        "TransitGatewayAttachmentId": "tgw-attach-0b86f8b6da54a18d3",
        "TransitGatewayId": "tgw-0d54e3fcd5455b568",
        "TransitGatewayOwnerId": "814910273374",
        "ResourceOwnerId": "814910273374",
        "ResourceType": "peering",
        "ResourceId": "tgw-0f53918ba065e8b2b",
        "State": "available",
        "Association": {
          "TransitGatewayRouteTableId": "tgw-rtb-0906255e29dc2f011",
          "State": "associated"
        },
        "CreationTime": "2026-02-04 17:34:35+00:00",
        "Tags": [
          {
            "Key": "Name",
            "Value": "liberdade-accept-peer01"
          }
        ]
      },
      {
        "TransitGatewayAttachmentId": "tgw-attach-0a3aeb7e1f2546862",
        "TransitGatewayId": "tgw-0d54e3fcd5455b568",
        "TransitGatewayOwnerId": "814910273374",
        "ResourceOwnerId": "814910273374",
        "ResourceType": "vpc",
        "ResourceId": "vpc-04ab35425c530e20a",
        "State": "available",
        "Association": {
          "TransitGatewayRouteTableId": "tgw-rtb-0906255e29dc2f011",
          "State": "associated"
        },
        "CreationTime": "2026-02-04 16:44:45+00:00",
        "Tags": [
          {
            "Key": "Name",
            "Value": "liberdade-attach-sp-vpc01"
          }
        ]
      }
    ]
  }
}

┌──(venv)-(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

Shows TGW attachments + routes that form the “legal corridor”.

Script 3 — malgus_cloudtrail_last_changes.py
Pulls recent CloudTrail events for “who changed what”.

--> Event history is available by default; it provides a 90-day record of management events.

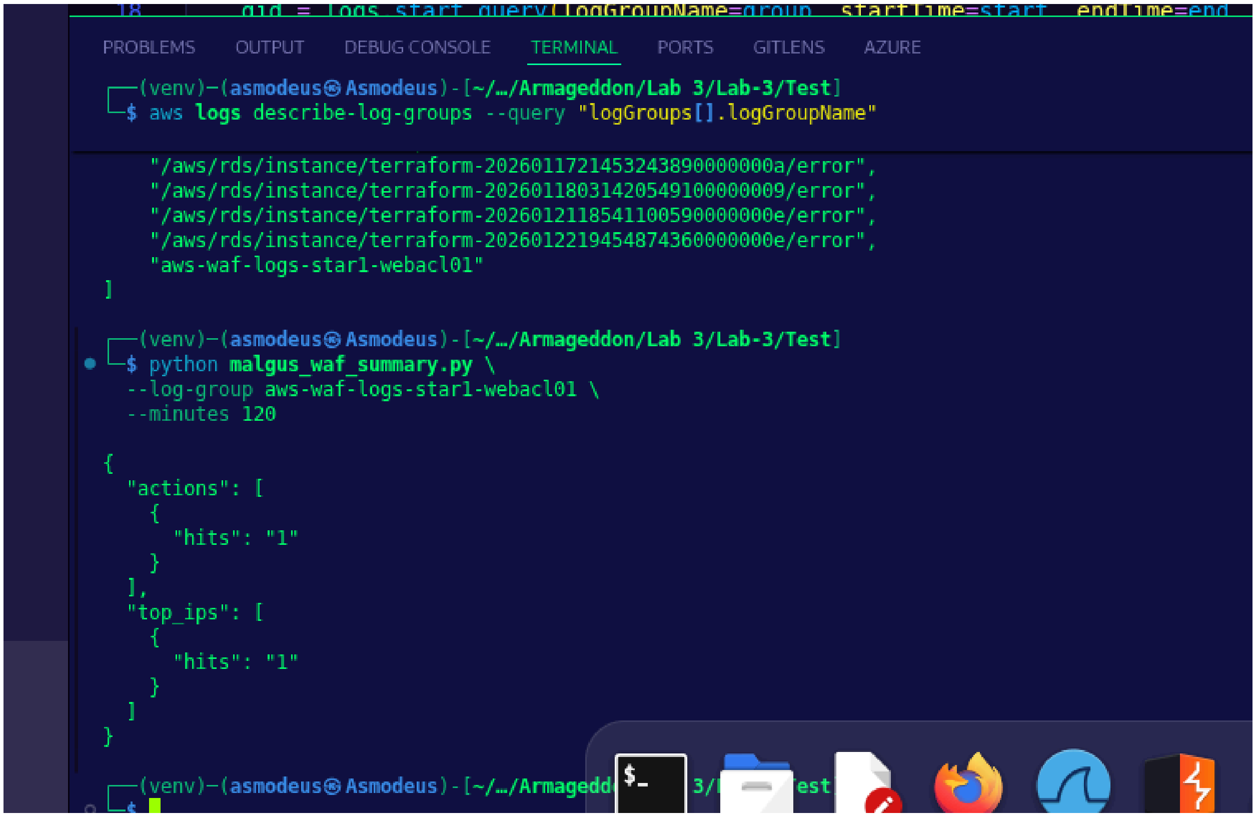
Not Created Yet



```
┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudtrail_last_changes.py
{
  "tokyo": [
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:53:01-05:00",
      "event": "ListProjects",
      "user": "resource-explorer-2",
      "source": "databrew.amazonaws.com"
    },
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:53:01-05:00",
      "event": "AssumeRole",
      "user": null,
      "source": "sts.amazonaws.com"
    },
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:52:08-05:00",
      "event": "AssumeRole",
      "user": null,
      "source": "sts.amazonaws.com"
    },
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:52:08-05:00",
      "event": "DescribeDBInstanceAutomatedBackups",
      "user": "resource-explorer-2",
      "source": "rds.amazonaws.com"
    },
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:50:46-05:00",
      "event": "DescribeNetworkInterfaces",
      "user": "AWSVLI",
      "source": "ec2.amazonaws.com"
    },
    {
      "region": "ap-northeast-1",
      "time": "2026-02-07 01:50:46-05:00",
      "event": "DescribeNetworkInterfaces",
      "user": "AWSVLI",
```

It goes for many pages

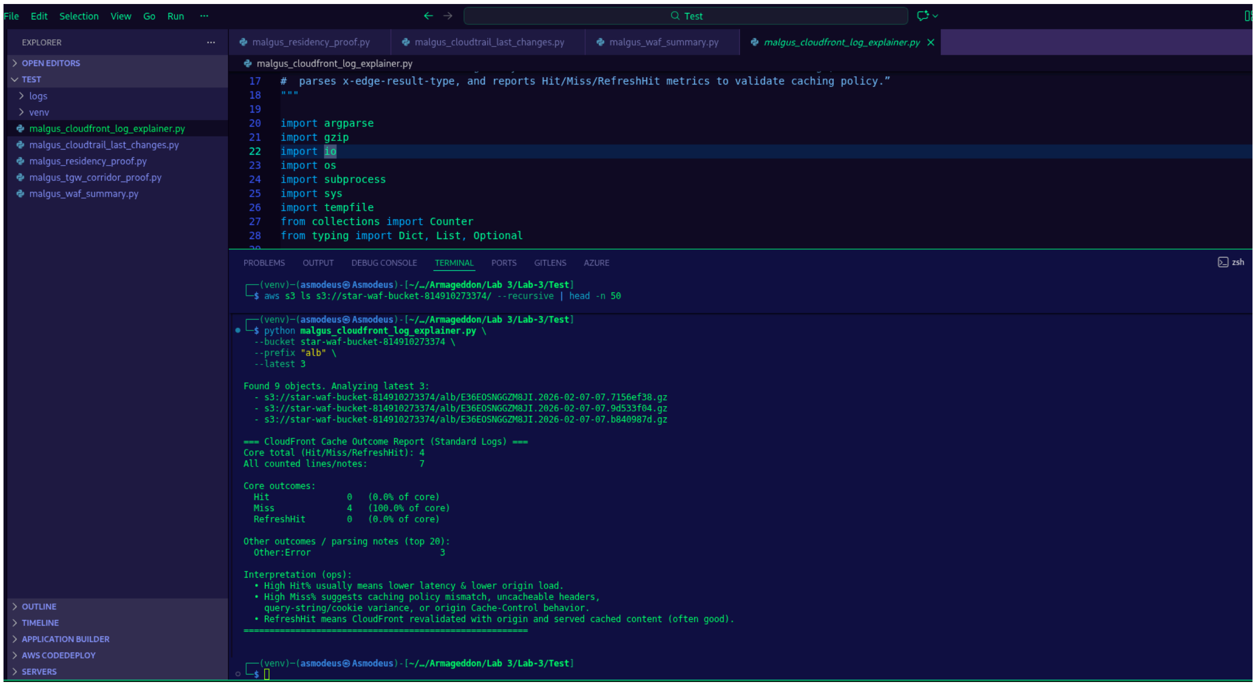
Script 4 — malgus_waf_summary.py
Summarizes WAF logs (Allow vs Block) from CloudWatch Logs destination.
WAF logging destinations: CloudWatch Logs, S3, Firehose.



```
        qid = logs.start_query(logGroupName=group, startTime=start, endTime=end
```

```
┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ aws logs describe-log-groups --query "logGroups[].logGroupName"

    "/aws/rds/instance/terraform-2026011721453243890000000a/error",
    "/aws/rds/instance/terraform-20260118031420549100000009/error",
    "/aws/rds/instance/terraform-2026012118541100590000000e/error",
    "/aws/rds/instance/terraform-2026012219454874360000000e/error",
    "aws-waf-logs-star1-webacl01"
]

┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_waf_summary.py \
  --log-group aws-waf-logs-star1-webacl01 \
  --minutes 120

{
  "actions": [
    {
      "hits": "1"
    }
  ],
  "top_ips": [
    {
      "hits": "1"
    }
  ]
}

┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armagedd          3/        est
└─$
```

Script 5 — malgus_cloudfront_log_explainer.py (optional)
If you ingest CloudFront standard logs into S3, this script reads a log file and counts Hit/Miss/RefreshHit.

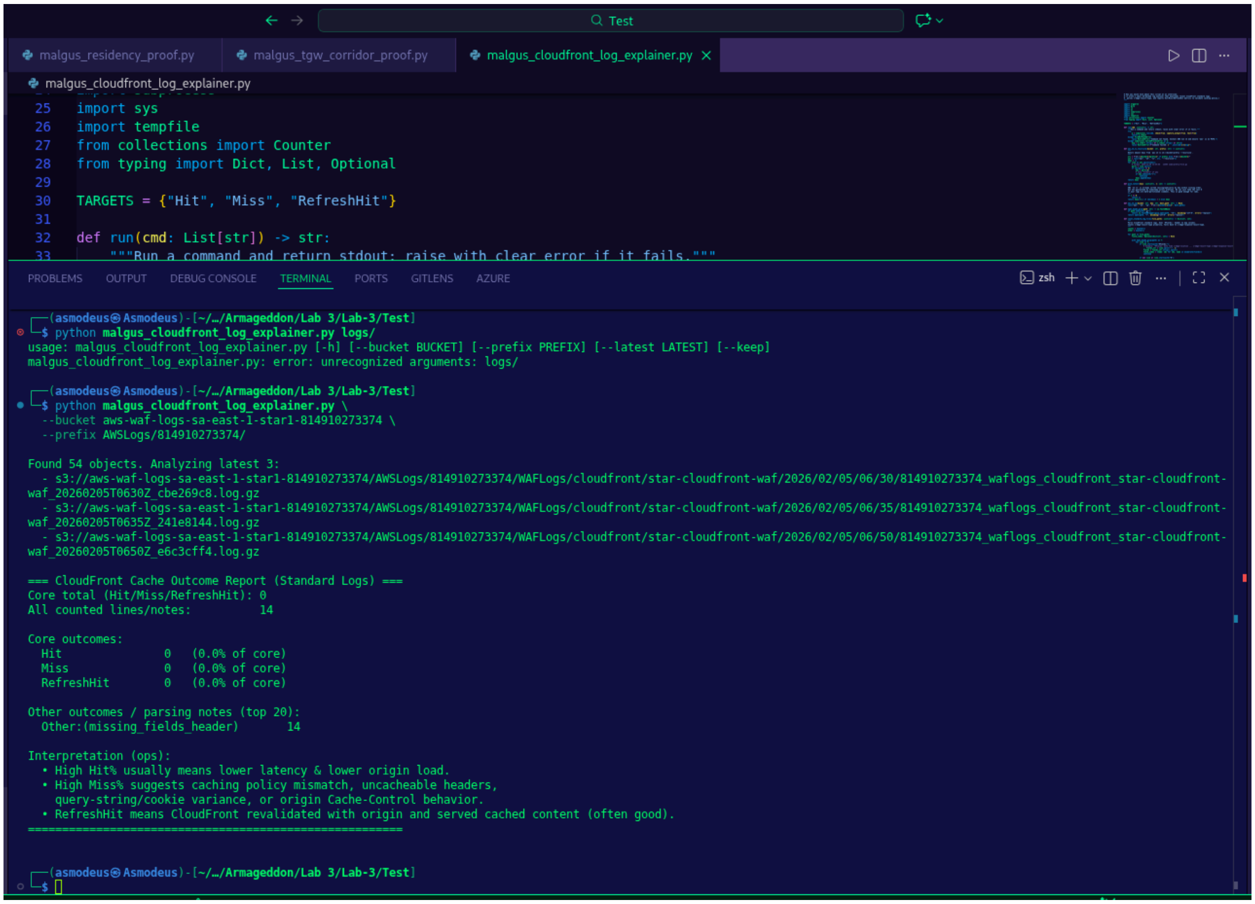

File Edit Selection View Go Run ···
Test
EXPLORER
OPEN EDITORS
TEST
logs
venv
malgus_cloudfront_log_explainer.py
malgus_cloudtrail_last_changes.py
malgus_residency_proof.py
malgus_tgw_corridor_proof.py
malgus_waf_summary.py
OUTLINE
TIMELINE
APPLICATION BUILDER
AWS CODEDEPLOY
SERVERS
malgus_residency_proof.py
malgus_cloudtrail_last_changes.py
malgus_waf_summary.py
malgus_cloudfront_log_explainer.py
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
GITLENS
AZURE
zsh

```python
#  parses x-edge-result-type, and reports Hit/Miss/RefreshHit metrics to validate caching policy.”
"""

import argparse
import gzip
import io
import os
import subprocess
import sys
import tempfile
from collections import Counter
from typing import Dict, List, Optional
```

```
┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ aws s3 ls s3://star-waf-bucket-814910273374/ --recursive | head -n 50

┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py \
  --bucket star-waf-bucket-814910273374 \
  --prefix "alb" \
  --latest 3

Found 9 objects. Analyzing latest 3:
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.7156ef38.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.9d533f04.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.b840987d.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 4
All counted lines/notes:          7

Core outcomes:
  Hit               0   (0.0% of core)
  Miss              4   (100.0% of core)
  RefreshHit        0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:Error                         3

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=======================================================

┌──(venv)─(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

CloudFront standard logs reference Hit / RefreshHit semantics.
A) Standard logs in S3 (downloaded locally)

```
	python3 malgus_cloudfront_log_explainer.py --mode standard cloudfront.log.gz
	python3 malgus_cloudfront_log_explainer.py --mode standard cloudfront_part1.log
cloudfront_part2.log
```

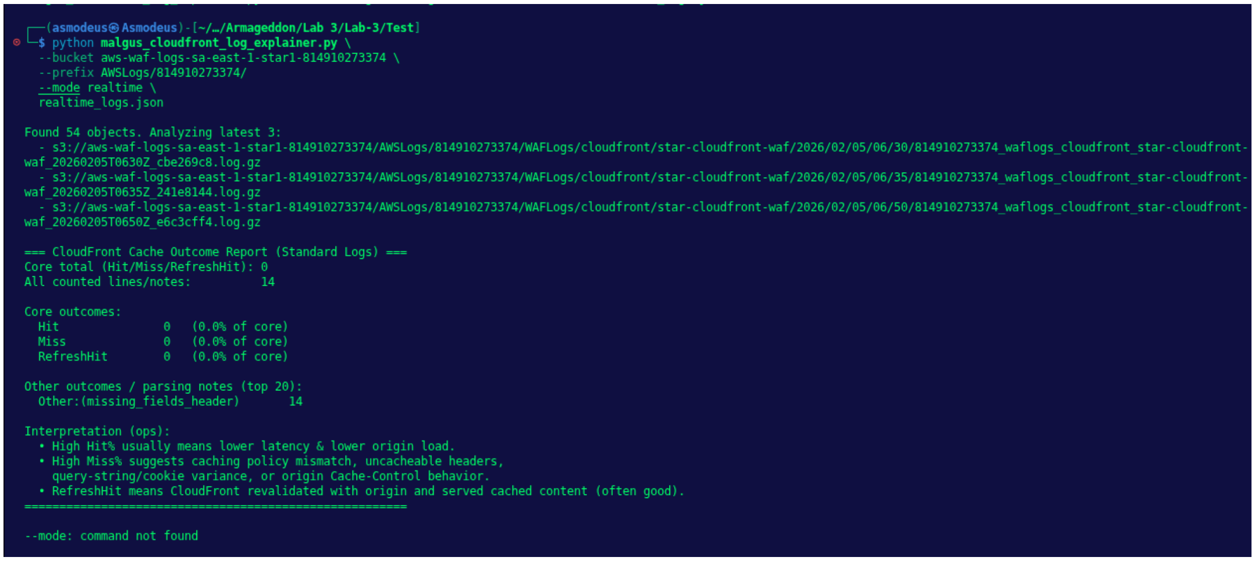

Test
malgus_residency_proof.py
malgus_tgw_corridor_proof.py
malgus_cloudfront_log_explainer.py
malgus_cloudfront_log_explainer.py
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
GITLENS
AZURE
zsh

```
import sys
import tempfile
from collections import Counter
from typing import Dict, List, Optional

TARGETS = {"Hit", "Miss", "RefreshHit"}

def run(cmd: List[str]) -> str:
    """Run a command and return stdout; raise with clear error if it fails."""
```

```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py logs/
usage: malgus_cloudfront_log_explainer.py [-h] [--bucket BUCKET] [--prefix PREFIX] [--latest LATEST] [--keep]
malgus_cloudfront_log_explainer.py: error: unrecognized arguments: logs/

┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py \
  --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
  --prefix AWSLogs/814910273374/

Found 54 objects. Analyzing latest 3:
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0630Z_cbe269c8.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0635Z_241e8144.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0650Z_e6c3cff4.log.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 0
All counted lines/notes:          14

Core outcomes:
  Hit               0   (0.0% of core)
  Miss              0   (0.0% of core)
  RefreshHit        0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:(missing_fields_header)       14

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
========================================================

┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

B) Real-time logs as JSON lines

python3 malgus_cloudfront_log_explainer.py --mode realtime realtime_logs.jsonl



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py \
  --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
  --prefix AWSLogs/814910273374/
  --mode realtime \
  realtime_logs.json

Found 54 objects. Analyzing latest 3:
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0630Z_cbe269c8.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0635Z_241e8144.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0650Z_e6c3cff4.log.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 0
All counted lines/notes:          14

Core outcomes:
  Hit               0   (0.0% of core)
  Miss              0   (0.0% of core)
  RefreshHit        0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:(missing_fields_header)       14

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
========================================================

--mode: command not found
```

Final Lab Assumptions (Locked)

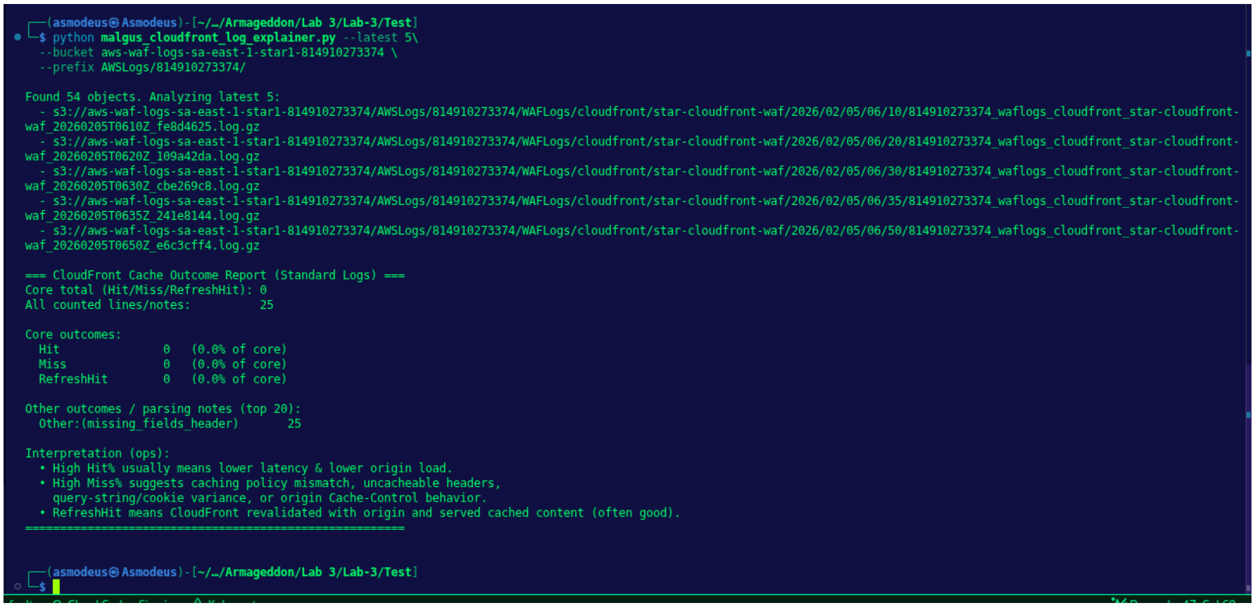
S3 Bucket: Class_Lab3

CloudFront Logs Prefix: Chwebacca-logs/ ← intentionally misspelled

AWS Account ID: 200819971986

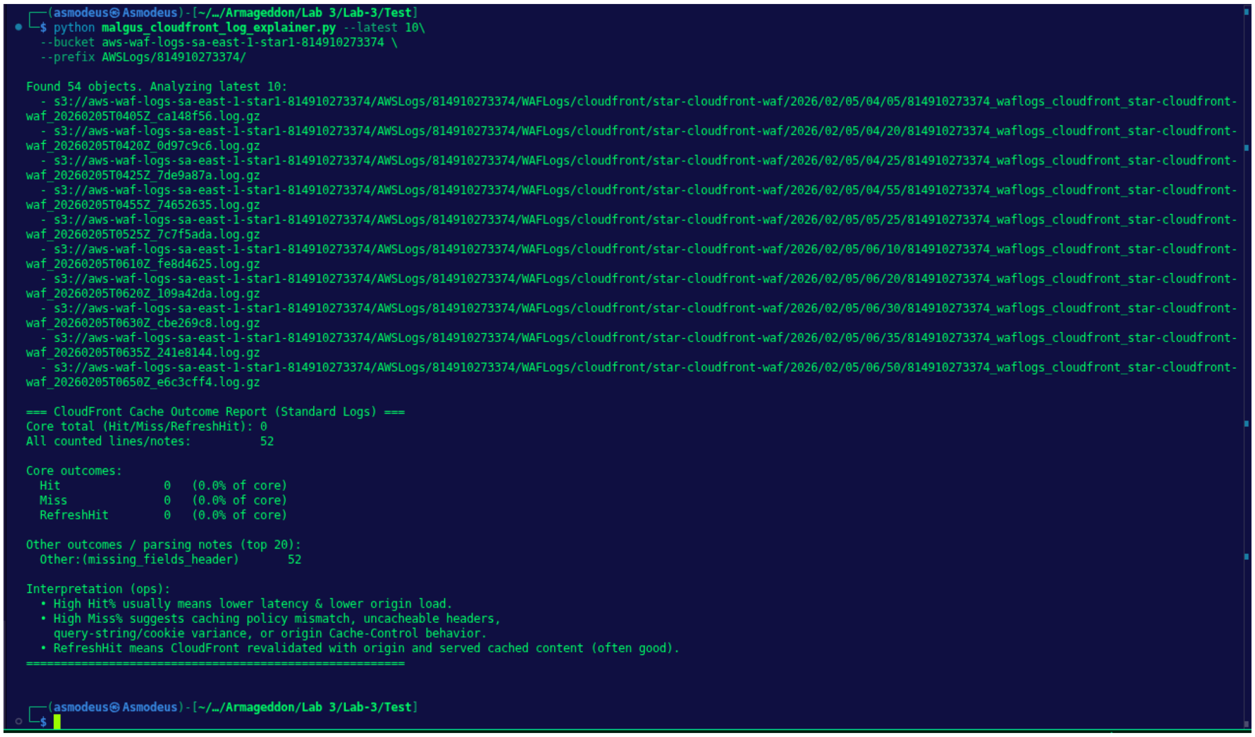
Running Scripts:

python3 malgus_cloudfront_log_explainer.py --latest 5



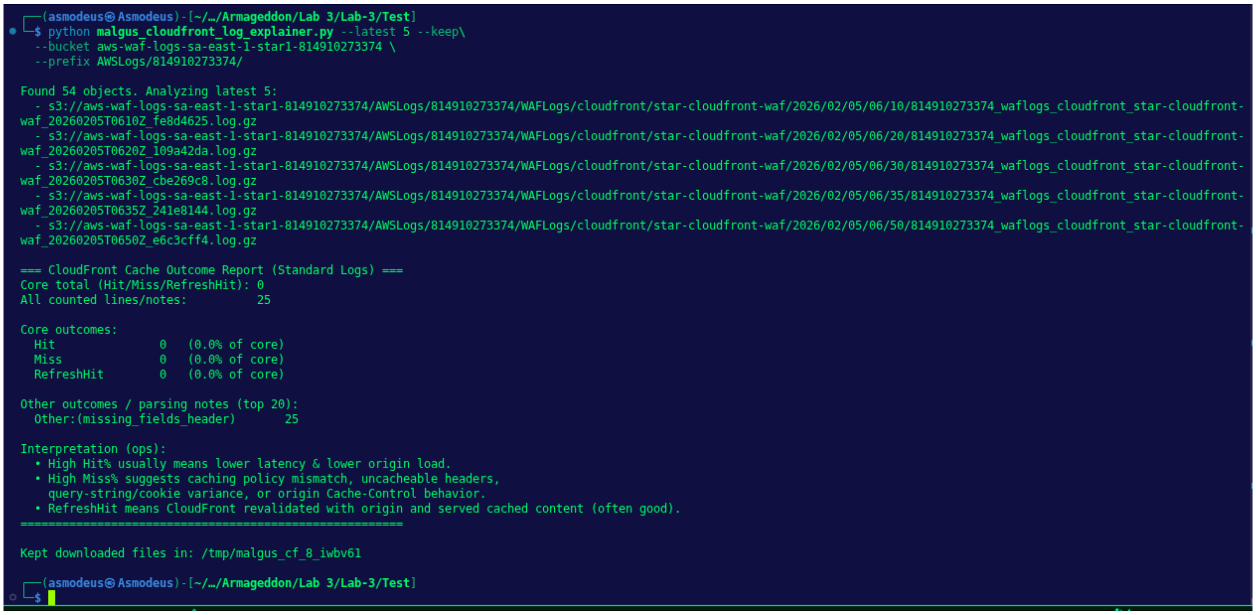
```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py --latest 5\
  --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
  --prefix AWSLogs/814910273374/

Found 54 objects. Analyzing latest 5:
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/10/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0610Z_fe8d4625.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/20/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0620Z_109a42da.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0630Z_cbe269c8.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0635Z_241e8144.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0650Z_e6c3cff4.log.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 0
All counted lines/notes:          25

Core outcomes:
  Hit              0   (0.0% of core)
  Miss             0   (0.0% of core)
  RefreshHit       0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:(missing_fields_header)       25

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=======================================================


┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

python3 malgus_cloudfront_log_explainer.py --prefix cloudfront-logs/ --latest 10



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py --latest 10\
  --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
  --prefix AWSLogs/814910273374/

Found 54 objects. Analyzing latest 10:
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/04/05/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0405Z_ca148f56.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/04/20/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0420Z_0d97c9c6.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/04/25/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0425Z_7de9a87a.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/04/55/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0455Z_74652635.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/05/25/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0525Z_7c7f5ada.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/10/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0610Z_fe8d4625.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/20/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0620Z_109a42da.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0630Z_cbe269c8.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0635Z_241e8144.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0650Z_e6c3cff4.log.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 0
All counted lines/notes:          52

Core outcomes:
  Hit              0   (0.0% of core)
  Miss             0   (0.0% of core)
  RefreshHit       0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:(missing_fields_header)       52

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=======================================================


┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

```
python3 malgus_cloudfront_log_explainer.py --prefix cloudfront-logs/ --latest 5 --keep
```



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py --latest 5 --keep\
  --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
  --prefix AWSLogs/814910273374/

Found 54 objects. Analyzing latest 5:
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/10/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0610Z_fe8d4625.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/20/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0620Z_109a42da.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0630Z_cbe269c8.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0635Z_241e8144.log.gz
  - s3://aws-waf-logs-sa-east-1-star1-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-
waf_20260205T0650Z_e6c3cff4.log.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 0
All counted lines/notes:          25

Core outcomes:
  Hit               0   (0.0% of core)
  Miss              0   (0.0% of core)
  RefreshHit        0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:(missing_fields_header)       25

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=======================================================

Kept downloaded files in: /tmp/malgus_cf_8_iwbv61

┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```

From stdin (nice for pipelines)

```
zcat cloudfront.log.gz | python3 malgus_cloudfront_log_explainer.py --mode standard -
```

Where “Hit / Miss / RefreshHit” come from (student-facing truth)

In standard CloudFront logs, you usually read the field:

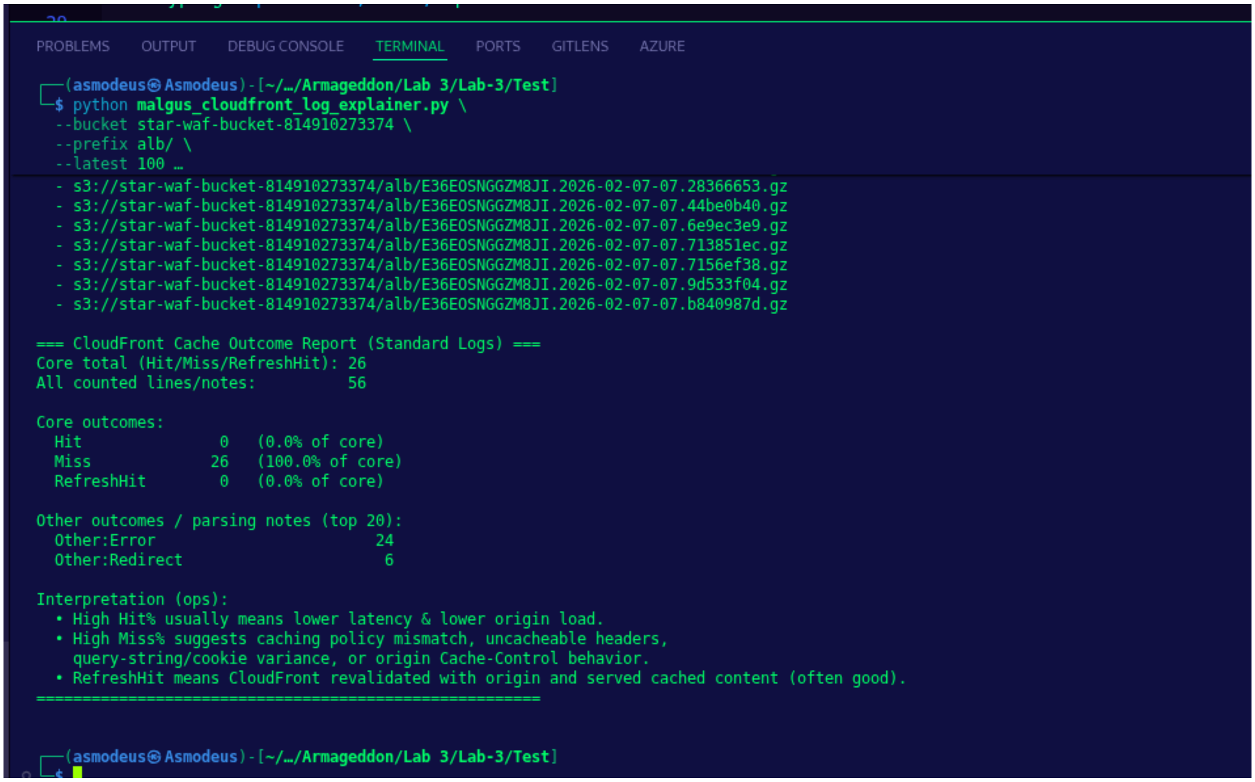
x-edge-result-type (primary)

sometimes also x-edge-response-result-type

Values commonly include: Hit, Miss, RefreshHit, plus other states like Error, LimitExceeded, etc.

That’s why the script reports “Other:*” — so students don’t blindly ignore unusual outcomes.

```
python malgus_cloudfront_log_explainer.py \
  --bucket <bucket-name> \
  --prefix <prefix>/ \
  --latest 3
```



```
┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ python malgus_cloudfront_log_explainer.py \
  --bucket star-waf-bucket-814910273374 \
  --prefix alb/ \
  --latest 100 …
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.28366653.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.44be0b40.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.6e9ec3e9.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.713851ec.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.7156ef38.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.9d533f04.gz
  - s3://star-waf-bucket-814910273374/alb/E36EOSNGGZM8JI.2026-02-07-07.b840987d.gz

=== CloudFront Cache Outcome Report (Standard Logs) ===
Core total (Hit/Miss/RefreshHit): 26
All counted lines/notes:          56

Core outcomes:
  Hit               0   (0.0% of core)
  Miss             26   (100.0% of core)
  RefreshHit        0   (0.0% of core)

Other outcomes / parsing notes (top 20):
  Other:Error                     24
  Other:Redirect                   6

Interpretation (ops):
  • High Hit% usually means lower latency & lower origin load.
  • High Miss% suggests caching policy mismatch, uncacheable headers,
    query-string/cookie variance, or origin Cache-Control behavior.
  • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=======================================================


┌──(asmodeus㉿Asmodeus)-[~/…/Armageddon/Lab 3/Lab-3/Test]
└─$ 
```