

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"
  }
]
```

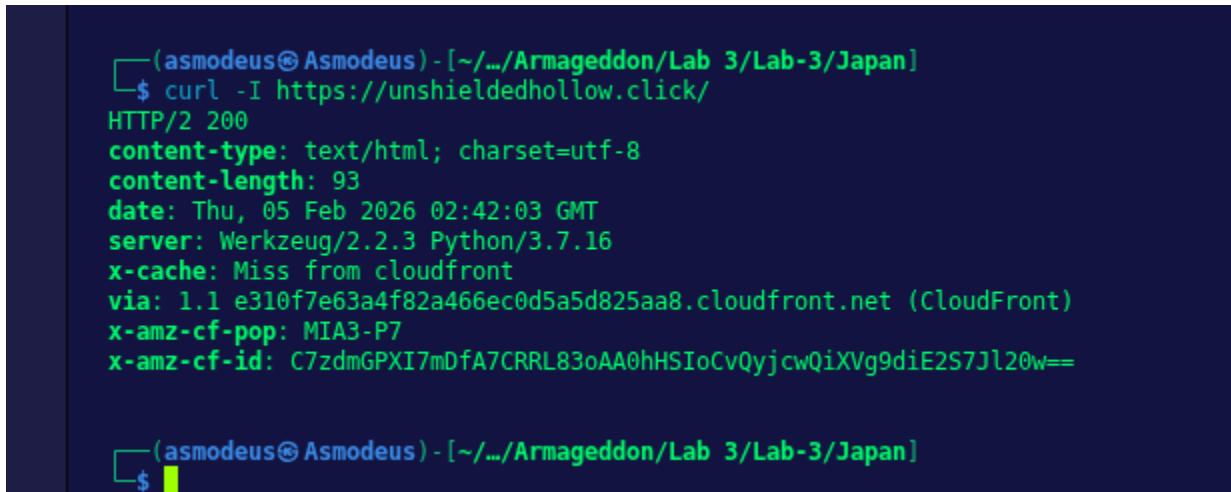
São Paulo: No RDS

```
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]
$
```

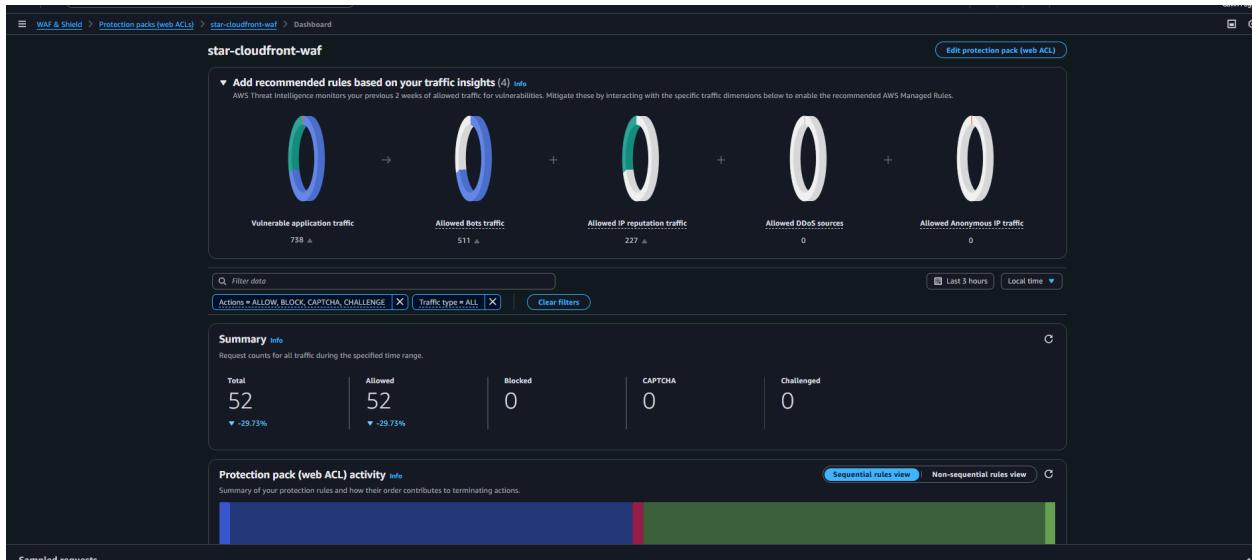
And/or submit CloudFront standard log evidence (Hit/Miss/RefreshHit)

3) WAF proof

Provide:

WAF log snippet or Insights summary

WAF logging destination options are documented



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					
Select a lookup attribute key					
Enter a lookup value					
Event name	Event time	User name	Event source	Resource type	Resource name
GetOriginRequestPol...	February 04, 2026, 17:18:26 (UT...	AWSVLU	cloudfront.amazonaws.com	-	-
ListRolePolicies	February 04, 2026, 17:18:26 (UT...	AWSVLU	iam.amazonaws.com	-	-
ListTagsForResource	February 04, 2026, 17:18:26 (UT...	AWSVLU	wafv2.amazonaws.com	-	-
ListTagsForCertificate	February 04, 2026, 17:18:26 (UT...	AWSVLU	acm.amazonaws.com	-	-
DescribeCertificate	February 04, 2026, 17:18:26 (UT...	AWSVLU	acm.amazonaws.com	-	-
ListAttachedRolePol...	February 04, 2026, 17:18:26 (UT...	AWSVLU	iam.amazonaws.com	-	-
GetWebACL	February 04, 2026, 17:18:26 (UT...	AWSVLU	wafv2.amazonaws.com	-	-
GetHostedZone	February 04, 2026, 17:18:26 (UT...	AWSVLU	route53.amazonaws.co...	AWS::Route53::Hosted...	Z03792151WUZ7JF79VV6Q
GetRole	February 04, 2026, 17:18:26 (UT...	AWSVLU	iam.amazonaws.com	-	-
GetCachePolicy	February 04, 2026, 17:18:26 (UT...	AWSVLU	cloudfront.amazonaws.com	-	-
GetCallerIdentity	February 04, 2026, 17:18:25 (UT...	AWSVLU	sts.amazonaws.com	-	-
ListManagedNotificat...	February 04, 2026, 17:17:55 (UT...	AWSVLU	notifications.amazon...	-	-
ListManagedNotificat...	February 04, 2026, 17:17:54 (UT...	AWSVLU	notifications.amazon...	-	-
ListManagedNotificat...	February 04, 2026, 17:17:14 (UT...	AWSVLU	notifications.amazon...	-	-
ListManagedNotificat...	February 04, 2026, 17:16:54 (UT...	AWSVLU	notifications.amazon...	-	-
ListManagedNotificat...	February 04, 2026, 17:16:50 (UT...	AWSVLU	notifications.amazon...	-	-
ListManagedNotificat...	February 04, 2026, 17:15:54 (UT...	AWSVLU	notifications.amazon...	-	-

5) Network corridor proof (TGW)

Students prove:

TGW attachments exist in both regions
routes point cross-region CIDRs to TGW

Tokyo

Attachments									
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			
<input checked="" type="checkbox"/> shinjuku-to-liberdade...	tgw-attach-0b86f8b6da54a18d3	tgw-Of53918ba065e8b2b	Available	Peering	tgw-0d54e3fd5455b568	tgw-rtb-06baa2f2bb9f9e8de			
<input type="checkbox"/> shinjuku-attach-toky...	tgw-attach-02c4692614c6f2fb6	tgw-Of53918ba065e8b2b	Available	VPC	vpc-0460407572b995c99	tgw-rtb-06baa2f2bb9f9e8de			

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

Details | Flow logs | Tags

Details			
Transit gateway attachment ID tgw-attach-0b86f8b6da54a18d3	State Available	Resource type Peering	Association route table ID tgw-rtb-06baa2f2bb9f9e8de
Requester ID tgw-Of53918ba065e8b2b	Requester region Tokyo (ap-northeast-1)	Requester owner ID 814910273374	Association state Associated
Acceptor ID tgw-0d54e3fd5455b568	Acceptor region São Paulo (sa-east-1)	Acceptor owner ID 814910273374	Dynamic routing Disabled

Sao-Paulo

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
Requester ID tgw-0f53918ba065e8b2b	Requester region Tokyo (ap-northeast-1)	Requester owner ID 814910273374
Acceptor ID tgw-0d54e3fc5455b568	Acceptor region São Paulo (sa-east-1)	Accepter owner ID 814910273374
		Association route table ID tgw-rtb-0906255e29dc2f011
		Association state Associated

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

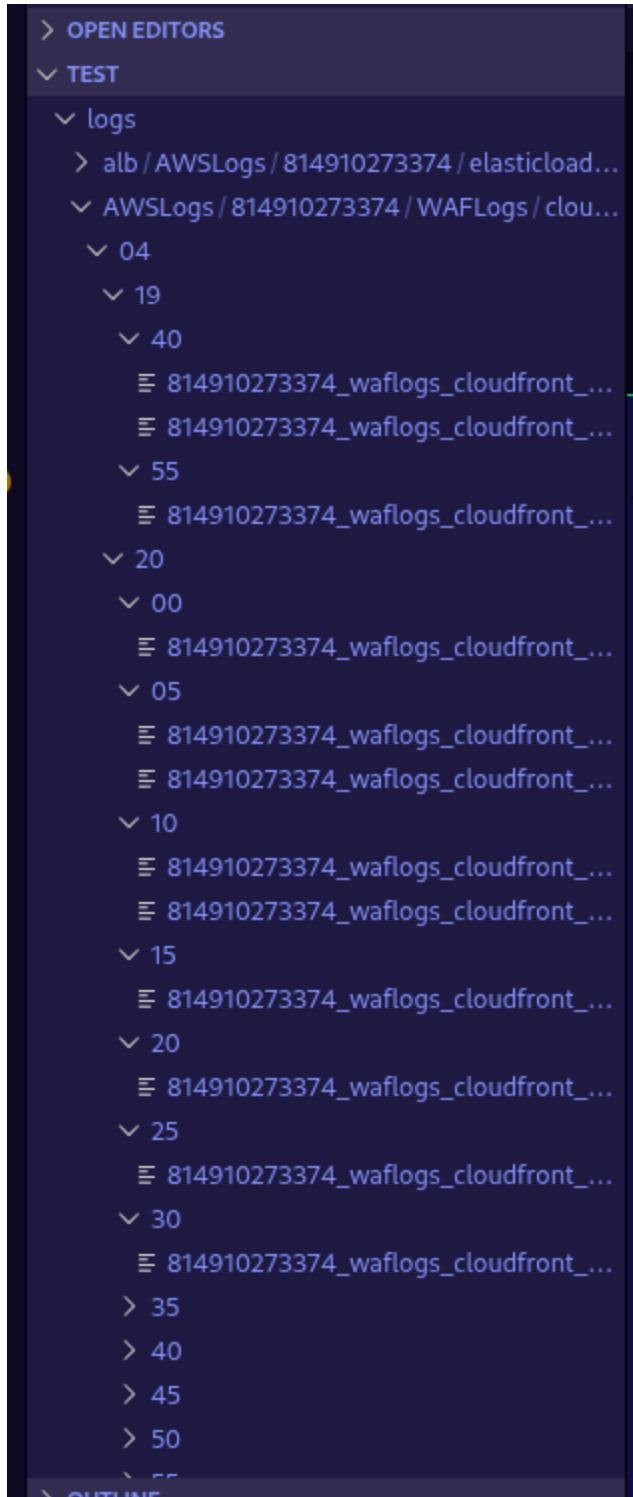
```
aws s3 ls s3://Class_Lab3/
```

If logs are under a folder/prefix:

```
aws s3 ls s3://aws-alb-logs-sa-east-1-star1-814910273374/ --recursive | tail -n 20
```

```
[asnodeus@Asnodeus:~/Arnageddon/Lab 3/Log]# aws s3 ls s3://aws-alb-logs-sa-east-1-star1-814910273374/ --recursive | tail -n 20
2026-02-04 16:00:04    1133 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374.elasticloadbalancing.sa-east-1.app.LoadExternal.10e5401db27c5ea4_20260204T21082_54.232.74.66.1vscsfr3.log.gz
2026-02-04 16:00:04    571 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/app.LoadExternal.10e5401db27c5ea4_20260204T21082_54.233.248.94.69001021.log.gz
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.10e5401db27c5ea4_20260204T21152_54.233.248.94.3zv0mcf.log.gz
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.10e5401db27c5ea4_20260204T21302_54.232.74.66.569ccfb.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.10e5401db27c5ea4_20260204T21352_54.232.74.66.1ohjyhra.log.gz
2026-02-04 16:35:08   1318 alb/AWSLogs/814910273374/elasticloadbalancing.sa-east-1/app.LoadExternal.10e5401db27c5ea4_20260204T21352_54.233.248.94.4644ad97.log.gz
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.10e5401db27c5ea4_20260204T21552_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.10e5401db27c5ea4_20260204T21552_54.233.248.94.g7e0dlos.log.gz
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.10e5401db27c5ea4_20260204T22162_54.232.74.66.3kjulvu.log.gz
2026-02-04 17:10:08    818 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374.elasticloadbalancing.sa-east-1.app.LoadExternal.10e5401db27c5ea4_20260204T22162_54.233.248.94.ev4fw28c.log.gz
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.10e5401db27c5ea4_20260204T22152_54.232.74.66.260gdkmk.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.10e5401db27c5ea4_20260204T22152_54.233.248.94.4jh295r.log.gz
z
2026-02-04 17:35:09    992 alb/AWSLogs/814910273374/elasticloadbalancing/sa-east-1/2026/02/04/814910273374.elasticloadbalancing.sa-east-1.app.LoadExternal.10e5401db27c5ea4_20260204T22352_54.233.248.94.4whteaom.log.gz
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.10e5401db27c5ea4_20260204T22402_54.232.74.66.2vichn0u.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.10e5401db27c5ea4_20260204T22302_54.233.248.94.hbm1e3r.log.gz
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.10e5401db27c5ea4_20260204T23402_54.232.74.66.3ttttnji.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.10e5401db27c5ea4_20260204T23402_54.233.248.94.2c74ctbd.log.gz
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.10e5401db27c5ea4_20260204T23452_54.232.74.66.3x88tp.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.10e5401db27c5ea4_20260204T23452_54.233.248.94.5zx37g7.log.gz
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.10e5401db27c5ea4_20260204T23552_54.233.248.94.6xznnxyv.log.gz
z
```

Download one file manually (sanity check):



```
aws s3 cp s3://aws-waf-logs-sa-east-1-star1-814910273374/ logs.gz .
```

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

```
terminal Help ← → Q Test ⌂ ⌂ ⌂ ... malgus_residency_proof.py ✘ malgus_tgw_corridor_proof.py ✘ malgus_cloudfront_log_explainer.py ... malgus_residency_proof.py
1 #!/usr/bin/env python3
2 import boto3, json
3
4 # Reason why Darth Malgus would be pleased with this script.
5 # Malgus wants proof, not opinions: "Show me the database lives ONLY in Tokyo."
6 # Reason why this script is relevant to your career.
7 # Auditors demand evidence bundles. Automating compliance proofs is real-world SRE/SEC work.
8 # How you would talk about this script at an interview.
9 # "I automated data residency verification by checking RDS inventory across regions and exporting an audit artifact."
10
11 def list_rds(region):
12     rds = boto3.client("rds", region_name=region)
13     resp = rds.describe_db_instances()
14     out = []
15     for d in resp.get("DBInstances", []):
16         out.append({
17             "region": region,
18             "id": d["DBInstanceIdentifier"],
19             "az": d.get("AvailabilityZone"),
20             "endpoint": d.get("Endpoint", {}).get("Address")
21         })
22     return out
23
24 def main():
25     tokyo = list_rds("ap-northeast-1")
26     sp = list_rds("sa-east-1")
27
28     print(json.dumps({
29         "tokyo_rds": tokyo,
30         "saopaulo_rds": sp,
31         "assertion": "PASS"
32     }))
33
34 if __name__ == "__main__":
35     main()
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS AZURE

zsh + - ×

```
(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.clo4kyoarkz.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",
          "DefaultRouteableAssociation": "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-6d54e3fd5455b568",
        "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_peere1"
          }
        ]
      }
    ]
  }
}
```

```
        "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-0d54e3fc5455b568",
                "TransitGatewayArn": "arn:aws:ec2:sa-east-1:814910273374:transit-gateway/tgw-0d54e3fc5455b568",
                "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_attachments.py
{
    "TransitGatewayAttachments": [
        {
            "TransitGatewayAttachmentId": "tgw-attach-0b86f8b6da54a18d3",
            "TransitGatewayId": "tgw-0d54e3fcfd5455b568",
            "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-0d54e3fcfd5455b568",
            "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]
o $ █

```

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

Script 4 — malgus_waf_summary.py

Summarizes WAF logs (Allow vs Block) from CloudWatch Logs destination.

WAF logging destinations: CloudWatch Logs, S3, Firehose.

Not Created Yet

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.

Not Working

```
o [~]-(asmodeus@Asmodeus)-[~/Armageddon/Lab 3/Lab-3/Test]
$ python malgus_cloudfront_log_explainer.py
Traceback (most recent call last):
  File "/home/asmodeus/Downloads/Armageddon/Lab 3/Lab-3/Test/malgus_cloudfront_log_explainer.py", line 35, in run
    p = subprocess.run(cmd, check=True, capture_output=True, text=True)
  File "/usr/lib/python3.13/subprocess.py", line 577, in run
    raise CalledProcessError(retcode, process.args,
                           output=stdout, stderr=stderr)
subprocess.CalledProcessError: Command '['aws', 's3', 'ls', 's3://Class_Lab3', '--recursive']' returned non-zero exit status 254.

During handling of the above exception, another exception occurred:

Traceback (most recent call last):
  File "/home/asmodeus/Downloads/Armageddon/Lab 3/Lab-3/Test/malgus_cloudfront_log_explainer.py", line 222, in <module>
    raise SystemExit(main())
           ^^^^^^
  File "/home/asmodeus/Downloads/Armageddon/Lab 3/Lab-3/Test/malgus_cloudfront_log_explainer.py", line 174, in main
    keys = aws.s3.ls_recursive(args.bucket, args.prefix)
  File "/home/asmodeus/Downloads/Armageddon/Lab 3/Lab-3/Test/malgus_cloudfront_log_explainer.py", line 48, in aws.s3_ls_recursive
    out = run(["aws", "s3", "ls", uri, "--recursive"])
  File "/home/asmodeus/Downloads/Armageddon/Lab 3/Lab-3/Test/malgus_cloudfront_log_explainer.py", line 41, in run
    raise RuntimeError(f"Command failed: {cmd}\n{msg}")
RuntimeError: Command failed: aws s3 ls s3://Class_Lab3 --recursive
An error occurred (NoSuchBucket) when calling the ListObjectsV2 operation: The specified bucket does not exist

o [~]-(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
```

```

malgus_residency_proof.py malgus_tgw_corridor_proof.py malgus_cloudfront_log_explainer.py
25 import sys
26 import tempfile
27 from collections import Counter
28 from typing import Dict, List, Optional
29
30 TARGETS = {"Hit", "Miss", "RefreshHit"}
31
32 def run(cmd: List[str]) -> str:
33     """Run a command and return stdnout+ raise with clear error if it fails."""
34
35     (asmodeus@Asmodeus)-[~/Armageddon/Lab 3/Lab-3/Test]
36     $ python malgus_cloudfront_log_explainer.py logs/
37     usage: malgus_cloudfront_log_explainer.py [-h] [--bucket BUCKET] [--prefix PREFIX] [--latest LATEST] [--keep]
38     malgus_cloudfront_log_explainer.py: error: unrecognized arguments: logs/
39
40     (asmodeus@Asmodeus)-[~/Armageddon/Lab 3/Lab-3/Test]
41     $ python malgus_cloudfront_log_explainer.py \
42     --bucket aws-waf-logs-sa-east-1-star1-814910273374 \
43     --prefix AWSLogs/814910273374/
44
45     Found 54 objects. Analyzing latest 3:
46     - 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-
47     waf_20260205T0630Z_cbe269c8.log.gz
48     - 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-
49     waf_20260205T0635Z_241e8144.log.gz
50     - 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-
51     waf_20260205T0650Z_e6c3cff4.log.gz
52
53     === CloudFront Cache Outcome Report (Standard Logs) ===
54     Core total (Hit/Miss/RefreshHit): 0
55     All counted lines/notes: 14
56
57     Core outcomes:
58         Hit      0  (0.0% of core)
59         Miss    0  (0.0% of core)
60         RefreshHit 0  (0.0% of core)
61
62     Other outcomes / parsing notes (top 20):
63     Other:(missing_fields_header) 14
64
65     Interpretation (ops):
66     • High Hit% usually means lower latency & lower origin load.
67     • High Miss% suggests caching policy mismatch, uncachable headers,
68       query-string/cookie variance, or origin Cache-Control behavior.
69     • RefreshHit means CloudFront revalidated with origin and served cached content (often good).
70
71
72     (asmodeus@Asmodeus)-[~/Armageddon/Lab 3/Lab-3/Test]
73     $ [REDACTED]

```

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.jsonl
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, uncachable 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_feld4625.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_24le8144.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, uncachable headers, query-string/cookie variance, or origin Cache-Control behavior.
• RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=====
```

```
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_ca14bf56.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_9d97c9c6.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/2026/02/05/25/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T0525Z_7cf75ada.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_feld4625.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_24le8144.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, uncachable headers, query-string/cookie variance, or origin Cache-Control behavior.
• RefreshHit means CloudFront revalidated with origin and served cached content (often good).
=====
```

```
python3 malgus_cloudfront_log_explainer.py --prefix cloudfont-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-starl-814910273374 \
--prefix AWSLogs/814910273374/

Found 54 objects. Analyzing latest 5:
- s3://aws-waf-logs-sa-east-1-starl-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/10/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T06102_fed4d625.log.gz
- s3://aws-waf-logs-sa-east-1-starl-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/20/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T06202_109a42da.log.gz
- s3://aws-waf-logs-sa-east-1-starl-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/30/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T06302_cbe269c8.log.gz
- s3://aws-waf-logs-sa-east-1-starl-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/35/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T06352_241e8144.log.gz
- s3://aws-waf-logs-sa-east-1-starl-814910273374/AWSLogs/814910273374/WAFLogs/cloudfront/star-cloudfront-waf/2026/02/05/06/50/814910273374_waflogs_cloudfront_star-cloudfront-waf_20260205T06502_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 cloudfont.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.